UPPSALA
UNIVERSITET

IT 16 035
Examensarbete 30 hp
Juni 2016

Poietic design

Heuristics and applications

Johanna Löfvenberg

Institutionen för informationsteknologi
Department of Information Technology

UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
http://www.teknat.uu.se/student

Abstract

Poietic design: Heuristics and applications

Johanna Löfvenberg

Good design is often derived from user-centred design, and systems have to be too adapted for their users. This delimits the system and the user alike. With poietic design, the system is designed to be less limited and the user is encouraged to get a deeper, more intuitive understanding of how it works. The properties of the proposed design philosophy poietic design are based on theoretical research and examples of artefacts that excel in communicating how they work. The properties, or heuristics, are used to analyse a digital system and a display, and to propose poietic redesigns. An experiment is conducted where the performance of the poietically designed display is compared to an ecological interface design (EID) display. The EID display was optimised for one of the tasks executed while the poietic display was designed to perform approximately as good for both tasks. The results show that the poietic design performed better than the EID display in the task that the EID design was not specifically designed for. This implies that poietic design has good potential to be used as a way to design information visualisation displays.

Handledare: Mikael Laaksoharju
Ämnesgranskare: Anders Jansson
Examinator: Mats Lind
IT 16 035
Tryckt av: Reprocentralen ITC

Summary in Swedish

God design utmärks ofta av att användare intuitivt förstår hur en artefakt ska användas eller kan förutsäga hur den fungerar. Det idag dominerande sättet att tänka kring design handlar dock om att förenkla för användare att genomföra komplicerade uppgifter genom att dölja all komplexitet. Detta sätt att tänka fokuserar alltså inte på hur bra artefaktens utformning hjälper användaren att förstå hur denna fungerar och vilka effekter den har. Istället utgår det enbart från användaren, dennas egenskaper, och en förutbestämd uppfattning om dennas behov. Detta begränsar både system som användare.

Detta gäller till viss del även för 'ekologisk gränssnittsdesign' (ecological interface design, EID), en designfilosofi med fokus på att hjälpa expertanvändare att ta beslut i komplexa miljöer. EID använder med andra ord människan som måttstock; den utgår från våra (människors) egenskaper. Man designar artefakter utifrån dessa men tar hänsyn till möjligheter och hinder i miljön.

Det vore dock intressant att prova om det går att tänka på andra sätt kring design. En designfilosofi som fått namnet 'poietisk design' utgår från andra hållet; istället för att utgå från människans egenskaper värderar man istället hur artefakters egenskaper ska kunna göras förnimbara för människor. Meningen med poietisk design är att synliggöra en artefakts egenskaper för att kunna tillåta användare att utveckla nya intuitioner om dess funktion och de möjliga effekter den har på omvärlden, utan att värdera effekterna. En elsladd som lyser när den används och lyser starkare ju mer ström som dras är ett exempel på detta; man kan efter en stund göra en hyfsad uppskattning om hur mycket ström som används, men om detta är negativt eller positivt får användaren själv bedöma. Kort sagt handlar det om en slags informationsvisualiseringar. Liksom designfilosofin 'calm technology' får användaren en uppfattning av en process status på ett diskret sätt.

I detta arbete undersöks poietisk design och vad exakt som definierar designfilosofin och skiljer den från andra. Designfilosofins egenskaper som föreslås är baserade på teoretisk forskning och på exempel på artefakter som kommunicerar väl hur de fungerar. Egenskaperna, eller heuristikerna, används sedan för att analysera två informationsvisualiseringar, och till att föreslå alternativa poietiska designförslag till dessa. Slutligen utförs ett experiment där en poietiskt designad display jämförs med en display som är designad utifrån EID-principer. EID-displayen var optimerad för den ena av de två uppgifterna som utfördes medan den poietiska displayen var designad att fungera ungefär lika bra för båda uppgifterna. Resultaten visar att den poietiska displayen presterade bättre än EID-displayen i uppgiften som EID-displayen inte var specifikt designad för. Detta indikerar att poietisk design har god potential till att användas för att designa informationsvisualiseringar.

Dedications

I want to thank the people who have helped me to write this thesis. To Mikael Laaksoharju for all discussions about poietic design, for helping me with the experiment, and for always taking time to explain and discuss everything I have wondered about poietic design. To Anders Jansson for taking time to read and give feedback on the thesis. To Mats Lind for helping me with the experiment and with the statistical analysis of its results. To everyone who participated in the experiment. And finally, to my friends who read and gave feedback on my thesis.

Johanna Löfvenberg

Uppsala, June 2016

List of contents

List of figures

List of tables

1 Introduction

Over the years, good user interfaces have become progressively more important. A well-designed artefact or system is safer, more informative, and more enjoyable to use than something that is designed badly. It is not an easy task to design good user interfaces, though; users' subjective opinions and goals, along with design theories that make use of different perspectives create great challenges for designers.

Today, good design is often considered to be designed from a user-centred perspective. The technology should be adapted to its users, not the other way around. Computers should be 'user-friendly'; easy to use, intuitive, and not too complex. Complexity should be hidden away, as it is seen as the counterpart to the desired simplicity (Norman, 2010).

Unfortunately, such design perspectives can lead to delimiting user interfaces in an attempt to make them less confusing to the users. This kind of thinking paves the way for less powerful systems, where the user cannot utilise the full potential of the technology. For example, think of metaphors in computers: desktop, recycle bin, the floppy disk icon, and so on. While the metaphors helped early computer users to create mental models over how the computer's functions work, they might be more delimiting of the computers' potential uses today. As we are too used to the metaphors now to imagine other ways of doing things it is also difficult to stop using them. Furthermore, hiding functionalities makes for a clean interface, but it is also makes it more difficult for the users to find them (Norman, 2010, 2013).

In this thesis, a new design philosophy called 'poietic design' is discussed. Poietic design attempts to solve the issue about delimited systems and hidden complexity. It is true that the user does not need to know every detail of how an artefact works, but they should be able to form a useable mental model of it from its design. As will be discussed later, poietic design is largely about giving the user an intuitive understanding of the system.

What can we gain from poietic design? Hopefully, users will get a deeper understanding of the technology they use, which would lead to a better foundation for decision-making. This would support autonomous decisions. The artefact could also become less delimited, meaning that it opens up more possible actions and thus making it more powerful.

As poietic design is a rather novel concept, research is therefore lacking. The purpose of this thesis is to present heuristics based on this new way to look at design, and to apply said heuristics to real-life examples.

1.1 Research questions

Based on the discussion above, my research questions are the following:

1. What attributes define poietic design?
2. What distinguishes poietic design from other design philosophies?
3. Is poietic design useful for designing user interfaces?

In order to answer the first question, heuristics for poietic design needed to be defined. To produce the heuristics, background research has been conducted about a wide range of topics that were deemed to be related to poietic design in different ways. As no research has been done about poietic design before, the topics that were believed to be related to the design philosophy in question were based on discussions with the founder of poietic design, Mikael Laaksoharju. Some examples of artefacts that communicate well how they work have also been looked into in order to get a deeper understanding of which attributes are poietic and why.

To answer the second research question, a comparison between poietic design and the related topics has been performed. Besides helping to answer what attributes make poietic design unique, this also helped to define poietic design further.

To answer the third research question, two cases have been performed. Case 1 is an attempt to redesign an existing software into a more poietic version of it. Case 2 is an experiment in which a poietically designed display is tested and compared to a display that is designed after a different design philosophy. The purpose of both cases is to give an example of a poietically designed display and to show that poietic design not only can be used in designing user interfaces, but also to improve them. The purpose of the cases aligns with that of the whole thesis, which is to introduce poietic design as a plausible design philosophy for designing user interfaces.

1.2 Limitations

The design heuristics are not evaluated in any user studies other than that of Case 2, where the performance of a poietic display is tested. Thus, the heuristics will only be analysed theoretically with the exception for in Case 2. The reason for this is that it takes time for users to get used to new ways of interaction, which would require a longitudinal study. This unfortunately lies beyond the scope of this thesis.

1.3 Disposition

A short background of the motivations of poietic design is found below. After that comes the Theory section where all topics that were deemed to be related to poietic design are described. These topics are further compared to poietic design in a discussion in the Poietic design section, where the theoretical background of poietic design is also found.

Following that, a short discussion about artefacts that communicate well how they work is found. Then, the properties of poietic design are introduced and described in detail.

The two cases mentioned above are thoroughly described in the Cases section. In both cases, a design is described and a poietic redesign is introduced. In Case 2, the method, the results, and a discussion are found as well, since the poietic redesign was tested.

The thesis is finally concluded with a general discussion.

2 Background

When computers started to become available to the general public, people had to learn how to use them. Humans had to adapt to the computers. As the years went on, the cost of computers went down and they started to become more widespread among the public. With more consumers, the higher demands they could make on the companies that produced the computers and as such, the computer interfaces had to become easier to use and understand for the laymen. Computers had to be adapted to be used by humans. One could express this change in focus as a paradigm shift in design philosophy. The first paradigm or approach can be described as normative. Like instructions, normative approaches describe the one way to perform a task; they describe how a system and its users should behave, but this does not always reflect the reality (Vicente, 1999). The second paradigm or approach can be described as descriptive. In contrast to the normative approaches, descriptive approaches describe how the users actually behave; they focus on current work practices, but tend to ignore hitherto unexplored possibilities (Vicente).

The science of designing user interfaces is called interaction design. Human-computer interaction (HCI) is an associated science, but with more focus on how computers influence humans (Dix, 2009). Many different design philosophies and theories exist within HCI, presumably because it is a multidisciplinary science. HCI is both an academic discipline as well as a design discipline, while it also draws form other disciplines like psychology (Dix).

Several design philosophies exist due to the fact that there is no objectively correct way of designing interfaces, there is no easy answer to how they should behave and what they should look like. One can say that display and interface design is a subtle science and an exact art (Bennett & Flach, 2011). There are no objective truths in interface design as there are in the fields of mathematics and physics. There is no algorithm of interface design to make the interface perfect, hence the term ‘subtle science’ (Bennett & Flach). However, interface design is not purely an art form either. The design of a display or an interface is a creative act, but as opposed to general art, an interface or a display needs to convey specific messages, hence the term ‘exact art’ (Bennett & Flach).

Good design is often distinguished by the fact that users should intuitively understand how an artefact is used, or it can be predicted how it works. However, the current dominating way of thinking regarding design is about simplifying an artefact’s or a system’s way for users to perform complicated tasks by hiding its complexity (Cooper, Reimann, & Cronin, 2007). This way of thinking does not focus on how well the layout of the artefact helps the user to understand how it works and the effects of their actions. Instead, it is based solely from the user, their qualities, and a presupposed idea of their needs. This creates uninspired, weak systems where the possibilities are restricted. Furthermore, this way of designing does not support autonomy for users. In summary, computers today become too adapted for humans, so much that their users cannot access the computers’ full potential.

It would be interesting to investigate if we can think of design in other ways. A design philosophy under the name ‘poietic design’ comes from another direction; instead of basing artefacts on human characteristics, one values how the characteristics of artefacts can be made perceptible for humans. The meaning with poietic design is to make characteristics of artefacts visible in order to allow users to develop intuitions about its functions and the possible effects it has on the world, without evaluating said effects for them. In summary, poietic design deals with information visualisations.

This line of thinking belongs to the so-called third generation of design philosophy, a formative approach. “Normative approaches focus on legislating work. Descriptive approaches focus on portraying work. In contrast, formative approaches focus on identifying requirements both technological and organizational that need to be satisfied if a device is going to support work effectively” (Vicente, 1999, p. 110).

These three approaches can be applied to many areas of research, including HCI (Vicente, 1999). Poietic design belongs to the formative approach, that is, the third generation of design philosophy. The second generation of the user-focussed descriptive approach is still considered to be good design, however. While the lessons and knowledge from the previous generations should not be forgotten, they are not enough for complex socio-technical systems (Vicente).

As users, we are ready for this change. Many technologies go from complex to simple, to more complex again – but in improved ways (Norman, 2010). For example, cell phones were difficult to use in the beginning and were thus made simpler (it is not easy for designers to know what will work in the beginning). After some time, most people grew accustomed to cell phones, and started to want more functionality, more complexity. Computers are probably way overdue this step. They have gone from being incredibly complex and only usable by expert users to being used by almost everyone. Computers also get more and more prevalent, more ubiquitous (Weiser, 1991). People grow up with computers all around them and learn to use them earlier than ever. When people in general are more used to computers it is time to lay more focus on the computer itself again, rather than having it as a platform to build other tools, which is how the computer is often used like today (Mikael Laaksoharju, personal communication). Metaphors that helped early computer users to understand computer programs limits the computers' use more than they open up possibilities.

3 Theory

In order to understand the concept of poietic design, a broad background research has been conducted. Some, but certainly not all, of the concepts that are related to poietic design in any way are described and discussed below.

3.1 Ecological interface design

Ecological interface design (EID) was introduced by Rasmussen and Vicente (1989) to characterise a use-centred approach to display design (Flach, Tanabe, Monta, Vicente, & Rasmussen, 1998) as a way to help operators cope with system complexity (Borst, Flach, & Ellerbroek, 2015). The framework has roots from Gibson's work on visual perception. Gibson argued that animals are inseparable from the environment, and one does not see only with one's eyes, but with one's head and body as well; observation implies movement (Gibson, 1979). Thus, EID focuses on the work domain, i.e. the 'ecology' instead of the user or the technology (Borst et al.). Bennett and Flach (2011) called this a triadic approach, as opposed to the more conventional dyadic approach which only considers the user and the technology, not the work context. Other triadic approaches besides EID are distributed cognition and situated cognition (Bennett & Flach).

The term 'ecological' was chosen in recognition of Brunswik's and Gibson's work, who both emphasised the significant role of stimulus or ecology in shaping human performance (Borst et al., 2015). The term was chosen to emphasise the focus on constraints arising from the work ecology (Flach et al., 1998). However, as opposed from the common misconception, this does not mean that EID displays are simple, natural, and intuitive (Borst et al.). EID is primarily intended for domain experts, and with increased levels of automation it becomes important to show more information, not less (Borst et al.).

> *The goal of ecological representations is not to trivialize complex work such that an ecological interface will be natural and easy to use for someone with little or no experience, but rather to organize the information in ways that leverage the deep structure of the work domain* (Borst et al., 2015).

According to Rasmussen's model of cognitive control, there are three modes of processing: skill-based, rule-based, and knowledge-based (Bennett & Flach, 2011; Rasmussen, 1983). The lowest level is skill-based or direct processing or behaviour, that occurs in situations where the observer can coordinate directly with the ecology (Bennett & Flach). It is skill-based because it requires practice through doing, and because once the coordination is mastered, the result is very smooth and fluid interactions between observer and environment, and require very little mental effort (Bennett & Flach). The next level is rule-based processing or behaviour, where there is a consistent relation between the representation and the environment or between the representation and the appropriate response can be described by a simple rule (Bennett & Flach). Rule-based behaviours are usually associated with heuristics, that is, 'shortcuts' in the decision-making process, heteronomous thinking (see below). The highest level of processing is knowledge-based processing or behaviour, which occurs whenever a person is presented with an unfamiliar situation and has to learn the relation between the form of the representation and the meaning of it (Bennett & Flach). This behaviour requires deep levels of processing and high mental effort (Bennett & Flach).

The goal of EID is to have an interface that will not force cognitive control to a level higher than required by the demands of the task, and still provide support for each level.

> *The ultimate goal of ecological interface design is to provide effective decision-making and problem-solving support […] informed decisions about interface design can only be made within context of both the domain […] and the cognitive agents […] responsible for doing the work* (Bennett & Flach, 2011, p. 111).

3.2 Affordances and Signifiers

Affordances as a concept was first used by Gibson, who described them as offerings of nature, possibilities or opportunities (Gibson, 1979). To Gibson, affordances are a relationship, actionable properties between the world and an actor (Norman, 2004, 2010). We are capable of assessing objects according to their perceptible properties (Taylor, n.d.). Some affordances are not obvious, and many have yet to be realised, but there is a natural and direct relationship between the perceptible qualities of tangible things and what we can do with them (Taylor). The affordance is always there to be perceived (Gibson). This view matches Gibson's view on perception, which is very direct, natural, and ecological.

However, the term has also been used by Norman (Norman, 2004, 2013; Taylor, n.d.). Year 1988, he defined affordances as "the perceived and actual properties of the thing, primarily those fundamental properties that determine just how the thing could possibly be used […] Affordances provide strong clues to the operations of things" (Norman, 2013, p. 9). The concept caught on, but not always with true understanding (Norman, 2004). Norman admits that he should have used the term 'perceived affordance', as he cares more about what the user perceives than what is actually true (Norman, 2004). Designers should be concerned with the perceived affordances, since if the user cannot perceive the affordances, it might as well not exist (Norman, 2010). In product design, where one deals with physical objects, there can be both real and perceived affordances (Norman, 2004). In human-computer interaction, we cannot rely on the natural relationship between things and man (Taylor), the designer has only control over perceived affordances (Norman, 2004).

> *Because I can click anytime I want, it is wrong to argue whether a graphical object on the screen 'affords clicking'. It does. The real question is about the perceived affordance: Does the user perceive that clicking on that location is a meaningful, useful action to perform?* (Norman, 2004).

Yet later, Norman suggested that designers should be concerned about 'signifiers' rather than affordances (Kaptelinin, 2014). If affordances define what actions are possible, signifiers specify how people discover them (Norman, 2013). "I strongly urge the design community to distinguish between affordances and signifiers. In most cases, the word affordance should go away, for invariably the designer cares only about what can be perceived, which means the signifiers." (D. Norman, 2010, p. 229).

In summary, there is a distinction between Gibson's 'real' affordances and Norman's 'perceived' affordances. In this thesis, Gibson's meaning of the term 'affordances' will be referred to as affordances, and Norman's interpretation will be referred to as 'signifiers'.

3.3 Calm technology and Ubiquitous computing

The idea of technology that moves from the periphery of our attention to the centre of it, and then back again, is called 'calm technology' (Weiser & Brown, 1996). Weiser and Brown claim that designs that both calm and inform the user are not often seen together. Indeed, one just has to look at today's computers, tablets, and smartphones to see that they constantly demand and compete for our attention. With technology becoming more ubiquitously present, it will not be possible nor desired for all technology to be in the centre of the users' attention (Bakker, van den Hoven, Eggen, & Overbeeke, 2012).

However, some technology such as comfortable shoes or writing with a pen leads to calm and comfort, and Weiser and Brown believe that such technology engages our attention in a different way (Weiser & Brown, 1996). "Calm technology engages both the *center* and the *periphery* of our attention, and in fact moves back and forth between the two" (Weiser & Brown).

This attribute of calm technology, to move between the centre and the periphery of attention, is calming because of two reasons. The first reason is that by placing things in the periphery we can attend to more things than if we had everything at the centre of our attention; the periphery is informing without overburdening (Brown, 2012; Weiser & Brown, 1996). The second reason is that by recentering something that has been in the periphery we take control over it, this is calming through increased awareness and power (Brown; Weiser & Brown).

The most famous example of calm technology is probably ‘the dangling string’, created by the artist Natalie Jeremijenko (Weiser & Brown, 1996). ‘The dangling string’ is basically a string that twitches in proportion to how busy the network is, complemented by a characteristic noise (Weiser & Brown). Thus, the string provides information about the network traffic in the periphery of the viewer’s attention. Other examples of artefacts created with peripheral interactions in mind are the ‘CawClock’, a clock aimed to provide peripheral awareness of time (Bakker et al., 2012); ‘NoteLet’, a bracelet made for photographing moments the user wants to take notes of (Bakker et al.); and ‘Whack gestures’, by which users can interact by striking a device with the open palm or the heel of the hand, without having to grasp or glance at the device (Hudson, Harrison, Harrison, & LaMarca, 2010).

The notion of technology in the periphery, that is, calm technology, is related to the notion of affordances (Weiser & Brown, 1997). For Weiser and Brown, however, the term ‘affordance’ does not reach far enough into the periphery where a design must be attuned but not attended to (Weiser & Brown). Presumably, what Weiser and Brown mean with affordances is what in this report is described as signifiers.

Calm technology is often associated with ubiquitous computing. The term ‘ubiquitous computing’ was coined by Mark Weiser (1991) and it means that very small computers are more or less embedded into objects all around us (Friedewald & Raabe, 2011). Things tend to fade into the background when we learn something sufficiently well (Weiser). For example, writing is found everywhere around us, yet we are not overwhelmed by it. We hardly even notice it. Weiser means that computers will fade into the background, just like writing, as they “weave into the fabric of everyday life until they are indistinguishable from it”. Ubiquitous computing is probably associated with calm technology because the concepts are similar and because they originate from the same place (Xerox PARC) and from the same people. One could say that calm technology is an environmentally oriented area of ubiquitous computing, where the focus lies on unobtrusiveness and invisibility of the interface.

Weiser (1991) believed that ubiquitous computing would be the dominant mode of computer access by now. However, while ubiquitous computing is everywhere now, calm technology has been all but abandoned because it is harder to design and implement than traditional attention-grabbing technology (Brown, 2012). Weiser and Brown believe that it is worth to pursue despite the challenges:

> *The most potentially interesting, challenging, and profound change implied by the ubiquitous computing era is focus on* calm. *If computers are everywhere they better stay out of the way, and that means designing them so that the people being shared by the computers remain serene and in control* (Weiser & Brown, 1997).

3.4 Decision-making and autonomy

What decides that one has made a good decision? That depends on the situation. Broadly speaking, humans seem to use one out of two different cognitive processes when making decisions, so-called System 1 or System 2 thinking (Laaksoharju, 2014). System 1 thinking is rapid, automatic, instinctive, and relieve us from high mental effort (Sunstein, 2005), and is therefore a cognitively economic strategy (Laaksoharju). System 2 or rule-based thinking is slower, reflective, calculative, logical, and uses high mental effort (Laaksoharju; Sunstein).

One can also discuss decision-making in the more philosophical terms of ‘agency’. Agency extends from autonomy to heteronomy (Kagitcibasi, 2005). Autonomous morality means subject to one’s own rule, being self-governed, and heteronomous morality means subject to another’s rule, being governed from the outside (Kagitcibasi). Laaksoharju (2014, p. 66) defines 'heteronomy' as "the instinctive, habitual and authority bound attitude that leads us to believe that we are not really making decisions", and 'autonomy' as "a deliberate, purpose-seeking attitude, which requires systematic reflection about advantages and disadvantages in different courses of actions. In terms of decision-making, the word 'heteronomous' will be used for fast and low-cost mental effort decisions, and 'autonomous' for well thought-out and high-cost mental effort decisions.

It seems like we generally use System 1 thinking when making heteronomous decisions, and System 2 thinking when making autonomous decisions. Indeed, an ethical or autonomous choice needs System 2 thinking (Laaksoharju, 2014). This connection between these concepts has seldom been described with the exception of Laaksoharju, probably because they belong to different disciplines. Sunstein (2005) discusses System 1 and 2 thinking in morality, and according to him, autonomous decision-making seems to be more or less equal to System 2 thinking. However, autonomy needs purpose-seeking independence as well, as System 1 and System 2 thinking processes do not account for all aspects of decision-making (Laaksoharju). Thus, in order to make autonomous decisions one also needs to have focus on the goal, what one tries to achieve (Mikael Laaksoharju, personal communication).

Heuristics belong to the System 1 processes (Laaksoharju, 2014). Research has shown that people make use of heuristics in real-world decision-making (Klein, 2008; Sunstein, 2005; Todd & Gigerenzer, 2000). People do not generate alternative options and compare them to each other, and even when people do compare options, they rarely use systematic evaluation techniques (Klein). This is because people have limited time, knowledge, and computational power (Todd & Gigerenzer). There are several models of heuristic decision-making, for example Klein's Recognition-Primed Decision model (Klein), Todd and Gigerenzer's fast and frugal heuristics, and Naturalistic Decision-Making theory (Epstein, 2013; Klein).

All of the heuristics mentioned above describe how people can make good and accurate decisions without having to compare options. Indeed, according to Rasmussen's model of cognitive control (Rasmussen, 1983), which distinguished skill-based, rule-based, and knowledge-based behaviour operating within the context of a decision-ladder that permits shortcuts in decision-making (Bennett & Flach, 2011; Klein, 2008; Vicente, 1999), experts develop associations that allow them to recognise solutions by using mental shortcuts. "Due to their extensive experience, experts have developed many associations that provide them with opportunities to bypass the intensive counting-out processes associated with the higher processes in the decision-ladder" (Bennett & Flach, 2011, p. 76). Heuristics are 'shortcuts' in decision-making. They are fast and mentally cheap. Thus, they are an example of System 1 thinking and heteronomy.

On the other end of the spectrum, we have autonomy. As opposed to heuristic shortcuts, using autonomous thinking would mean to take no shortcuts in Rasmussen's decision-ladder. According to Laaksoharju (2014), autonomy means moral maturity. In order to make ethical decisions, it is quite logical that one has to be analytical and carefully weigh options against each other. Sunstein (2005) means that moral assessments should not be based on intuitions, and that some moral judgments are unsound if they are based on moral heuristics. Ethics are usually regarded as something that is good and worth to pursue. Thus, good design should promote autonomy (Laaksoharju).

In summary, System 1 and 2 thinking are cognitive processes and autonomy and heteronomy are about decision-making. Heteronomous thinking is easier than autonomous thinking and usually leads to good decisions, but there are biases, such as confirmation bias, anchoring bias, and conjunction fallacy. Autonomous thinking is usually too mentally expensive for everyday decisions, and does not necessarily generate better decisions than heteronomous thinking. Both ways of thinking have advantages and disadvantages (Table 1). However, heteronomy and autonomy, and System 1 and 2 thinking, do not have to be mutually exclusive (Laaksoharju, 2014).

Table 1. A comparison between advantages and disadvantages of heteronomy and autonomy.

Heteronomy	**Autonomy**
Fast	Slow
Low mental effort (easy)	High mental effort (difficult)
Accurate	Not necessarily more accurate than heteronomous decisions
Less control in decisions	Control in decisions
Biased	Ethical

3.5 Complexity

People cry for simplicity, yet they do not actually want it (Norman, 2010), and designers sometimes try to make complexity disappear too hard, as if they believe that simplicity is axiomatically good (Janlert & Stolterman, 2010). Janlert and Stolterman discuss common strategies for dealing with complexity:

- Eliminate unnecessary complexity: To identify and eliminate unnecessary complexity without compromising functionality. This situation should not arise in the first place.
- Make it simple by sacrificing (quality of) function: To enforce simplicity, no matter what. Less a strategy and more like a retreat or denial.
- Hide complexity: To put the complexities behind a cover of simplicity (e.g. automation). Less interaction and control for users.
- Confine complexity: To concentrate and confine complexity to well defined and clearly delimited parts (e.g. 'advanced' settings). This sends the message 'Use at your peril' to the user.
- Dilute complexity: To spread and divide complexity over a wider area of interaction. This strategy presupposes that the artefact is or can be distributed in space.

However, people like complexity; complexity means more power, more flexibility, and more control (Norman, 2010). Complexity is also desirable when it can improve security, economy, sustainability, range, situatedness, and subtlety of function (Janlert & Stolterman, 2010). Humans seem to seek and enjoy certain experiences of complexity, which can be understood as richness (Janlert & Stolterman). The downside for users handling and living with complex artefacts are information overload, decision overload, communication overload, and a lack of understanding of their systems (Janlert & Stolterman). Yet there are situations that are complex without being overwhelming.

> *The experience of being in a forest with its overwhelming profusion of different life forms and natural structures is seen as richer than being in the controlled and simplified park. The simpler an environment is, the easier it is to understand and handle, but at the same time it lacks the richness and stimulation that we appreciate and enjoy* (Janlert & Stolterman, 2010).

Modern information technology tends to increase the complexity of artefacts (Janlert & Stolterman, 2010). However, it is a false assumption to believe that there is a trade-off between complexity and simplicity (Norman, 2010). They are not opposites; complexity is a fact of the world, whereas simplicity is in the mind of the user (Norman). Complex things do not have to be confusing, and confusing things do not have to be complex (Norman). What is important is how the artefact is designed: "Good design makes complex things simple to use" (Norman, 2010, p. 233).

As mentioned above, EID is not meant to oversimplify a work domain. EID was introduced by Rasmussen and Vicente as a way to help operators cope with system complexity, but with increased levels of automation it becomes important to show more information, not less (Borst et al., 2015). Representations can oversimplify a problem, which can potentially lead to a naïve view of the work domain and a restricted solution space for solving problems (Bennett & Flach, 2011). EID is directed to expert users, and expert users often require more complexity and less simplicity in order to not be restricted in their work. People generally prefer an intermediate level of complexity, and the desired level of complexity varies with knowledge and experience (Norman, 2010). Expert users are, naturally, quite knowledgeable and experienced, hence their need for complex systems.

One can distribute complexity in different ways, spreading them across the loci of complexity (Janlert & Stolterman, 2010):

- Internal: The complexity of the internal workings of the artefact.
- External: The complexity of the artefact's interface with the outside world and the user.
- Interaction: The complexity of the relation between input and output.
- Mediated: Complexity that is located not in the interior or the exterior of the artefact but in the environment, channelled through the artefact and impacting interaction complexity.

A hammer has low internal and external complexity. A typical VCR has high internal and external complexity. A violin has low internal and external complexity, but high interaction complexity as it is difficult to play even though the artefact in itself is simple. The score plays an important role in the complexity of playing the violin; difficult scores mean high mediated complexity.

3.6 Poiēsis

The word 'poiēsis' has been described as the arising of something from out of itself, a bringing-forth (Heidegger, 1977). Heidegger "used the concept of poiēsis to describe the refinement of material into something that had greater aesthetic and moral value" (Laaksoharju, 2014, p. 40). For example, a skilled craftsman brings forth qualities from a piece of a material, or the bursting of a blossom into bloom. Bringing-forth is a revealing, the 'truth', the correctness of an idea (Heidegger).

Technology is a way of revealing (Heidegger, 1977). It belongs to bringing-forth, to poiēsis (Heidegger). It is also linked with 'epistēmē', which means knowing; to be entirely at home in something, to understand and be an expert in it (Heidegger). However, the modern technology does not encompass this:

> *And yet the revealing that holds sway throughout modern technology does not unfold into a bringing-forth in the sense of poiēsis. The revealing that rules in modern technology is a challenging […] which puts to nature the unreasonable demand that it supply* [sic] *energy that can be extracted and stored as such. But does this not hold true for the old windmill as well? No. Its sails do indeed turn in the wind; they are left entirely to the wind's blowing. But the windmill does not unlock energy from the air currents in order to store it* (Heidegger, 1977).

According to Heidegger, modern technology does not capture the essence of technology. Technology should reveal its true nature, not extract and store energy. Modern technology is a tool, a means to achieve something, but it is not natural. This poses a threat to us:

> *The threat to man does not come in the first instance from the potentially lethal machines and apparatus of technology. The actual threat has already affected man in his essence. The rule of Enframing threatens man with the possibility that it could be denied to him to enter into a more original revealing and hence to experience the call of a more primal truth* (Heidegger, 1977).

With 'Enframing' Heidegger means the way of revealing in the essence of modern technology. Heidegger argues that modern technology is contained within a shell, a shell that hides the true nature of technology and interprets it for us. This hinders us from seeing the truth, to understand technology.

According to Heidegger, modern technology lacks poiēsis (Laaksoharju, 2014). “But perhaps it is still possible to talk about poiēsis when it comes to designers of technology; as crafting the digital material into a meaningful, tasteful representation of its essence?” (Laaksoharju, 2014, p. 40).

4 Poietic design

Poietic design is a design philosophy that originates from HCI researcher Mikael Laaksoharju, loosely inspired by the works of philosopher Martin Heidegger. The purpose of poietic design is to make an artefact's features perceptible in order to allow its users to develop new intuitions about its functions and its possible effects on the world. The artefact should not judge these effects; it is rather up to the user to interpret the data. By this sort of information visualisations, the users will get an understanding more similar to the designer of the artefact (a programmer, for example). The goal is to get an intuitive and perceptive interpretation of information rather than a cognitive interpretation.

As opposed to Heidegger (1977), who focussed on the relationship between technology and nature, Laaksoharju argues that the way of producing artefacts is irrelevant and that the relationship between humans and technology is more important (Mikael Laaksoharju, personal communication). Even 'non-natural' artefacts can possess poietic attributes, one can accomplish to bring forth the same sensitivity of the material's attributes even when it comes to technical and digital artefacts; technology does not have to lack poiēsis (Mikael Laaksoharju, personal communication). According to Mikael Laaksoharju (personal communication), we should be able to have the same relationship with technology as we would have had with nature, were we not living in a world of technology; we can form instincts to the relationship with technology.

Others have described the word 'poietic' as "participating in the creation of the work" and "the 'making' of an object" (Carlsson, 2010), and "productive, formative" (Eldridge, 2012). While these definitions are related to Laaksoharju's definition of poietic design, it is the latter's definition that will be discussed for the remainder of the thesis.

The meaning of poietic design will be further explained below while tying the concept together with the concepts discussed in the Theory section.

4.1 Poietic design and ecological interface design

Flach, Tanabe, Monta, Vicente, and Rasmussen (1998) discuss four approaches to interface design:

- Technology-centred approach: Focus on the capabilities and limitations of technologies. Emphasis on what the new technology can do.
- User-centred approach: Focus on the limitations and capabilities of human operators and the implication of these limitations for how systems should be designed. Emphasis on what humans can do.
- Control-centred approach: Focus on the coupling between humans (controllers) and technologies. Emphasis on the stability of the human-machine control loops.
- Use-centred (ecological) approach: The 'system' is more than the human and the machine; it consists of a work domain as well.

Based on the discussion of generations of design philosophies by Vicente (1999) in the Background section, one can draw parallels between the three generations and these approaches. The technology-centred approach seems to belong with the normative approach in the first generation, the user-centred approach with the descriptive approach in the second generation, and the control-centred and the use-centred approaches with the formative approach in the third generation.

The focus of EID is neither on object or user, but on the whole ecology, the work domain (Borst et al., 2015). That is, EID uses the use-centred or ecological approach. As this approach belongs to the third generation, one can say that EID and poietic design, which also belongs to the third generation, are in the same spirit; both design philosophies aim to empower the user and both are at least partly use-centred.

In EID one designs the environment to make it suited to the human. In poietic design one rather designs the artefact so that it reveals what it can do and what one can do with it. One can thus say that poietic design uses the technology-centred approach. In the technology-centred approach, the interface is generally designed in a way that reflects the technological capabilities (Flach et al., 1998). A concern about this approach is that it can lead to a proliferation of displays that operators must process (Flach et al.). It is also limited in the way that it does not focus on the work domain, thus gives a narrower image of the system than the use-centred approach (Flach et al.). However, poietic design does not have the same concerns, because as opposed to a pure technology-based approach where the user has to adapt to the technology, poietic approach has a more holistic view.

Poietic design also makes use of the user-centred approach. The reality consists of things that are perceptible to a human being, and even more things that are imperceptible. Some of the imperceptible things can become translated into perceptible data that humans might be interested in. This focus on the user is naturally more user-centred rather than technology-centred.

It is a poietic trait to translate imperceptible things into perceptible data, as the goal of poietic design is to give the user a more thorough understanding of what is going on. In order to get this understanding, one needs information of the artefact and of what is going on in the environment. Thus, poietic design is also concerned with the use-centred approach, as it is concerned with the environment, the work ecology.

In summary, poietic design has not just a technology-centred approach, but makes use of a user- and use-centred approach, too. But even though it has a part in all three of these, poietic design most definitively belongs to the third generation of design philosophies, using a formative approach. For even though the technology is in the centre of the design philosophy, it is still adapted for the users. In a system that is purely technology-centred, that is, a system using a solely normative approach, the user is mostly seen almost as an obstacle to the design. Meanwhile, a purely user-centred system, that is, a system using a solely descriptive approach, the technology gets too adapted to the humans, it loses some of its capabilities.

An example of a user-centred approach is our window-based computer systems that most people use today. For casual users they are much easier to use than command prompt systems, but they are also more limited in what they can do. Norman (1981) brings this up with the operating system UNIX. UNIX is a good example of a technology-centred approach. UNIX is an elegant and powerful system and it enables its users to perform actions easily (Norman). Unfortunately, it is very difficult for casual users to learn, and it is easy to make fatal mistakes (Norman). Though Norman describes the faults of UNIX at lengths, he still makes it clear that it is a superior operating system.

Would it not be good to have an operating system that is as powerful as UNIX, but that also helps the user to understand how to use it and why certain actions work and others not? Just like EID, poietic design provides the user with a deeper understanding of the system or artefact. The goal of poietic design is to get the user to think more like a programmer, to understand why certain actions produce certain results. The user should be able to understand what is going on in the system.

4.2 Poietic design and signifiers

Unlike design with good signifiers, poietically designed artefacts do not necessarily need to be obvious in their purpose and what one can use them for immediately. However, just like with signifiers, the user will get a feeling of how something works without having to read manuals. The system or artefact does not need to have good signifiers, but it has to be possible to discover everything that one can do with it.

Note that manuals per se are not strictly a sign of bad design according to poietic design. Engaging and relevant manuals are useful and often needed for complex systems and artefacts. Most people want ‘just-in-time’ learning, meaning they want to learn things when they need it (Norman, 2010). Only overly detailed, purely text-based manuals are considered bad. (Few people probably consider those kinds of manuals good!) Manuals should be reserved for quick and efficient instructional material (Norman).

In poietic design it is considered important to get an intuition of how artefacts or systems work and what certain actions lead to. As such, signifiers are important for giving the user these intuitions.

4.3 Poietic design and calm technology

Calm technology works well intuitively because the information given feels natural and it is easy to interpret its meaning. In the example of the dangling string, it feels natural and obvious that more motion activity (dangling) and noise would mean *more* of something, in this case more network activity.

A fan produces noises as a consequence of it working hard. Smoke and smell of burning are consequences of fire. These consequences might be annoying, but they still provide information. Today it is not unusual for cars to be equipped with noise and shock dampeners. But by removing all feedback from the environment we lose information (Norman, 2009).

As in calm technology, the user gets a feeling of a process' status in a discreet way from poietic design. However, as mentioned above it is not necessary that the user will get the intuitions immediately. Were the user to understand everything immediately, poietic design would be limited to simple systems. Instead, it is more important to give the user opportunities to develop intuitions over time.

Roughly, the goals of poietic design are similar to those of calm technology. In both design philosophies, the artefact or system gives the user an intuitive understanding of its status. The main difference between the design philosophies is that for calm technology, the user should be able to perceive data from the periphery, while for poietic design, the user can perceive the data in the centre of attention, but they might have to develop new intuitions in order to get the intuitive understanding.

4.4 Poietic design and autonomous decision-making

While heuristic decision-making has its own virtues, Sunstein (2005) argues that moral heuristics can lead to errors and confusion. As such, one could argue that without using autonomous thinking, one cannot be considered to truly make ethical decisions. Thus, good design should promote autonomy (Laaksoharju, 2014).

According to Laaksoharju (2014), autonomy requires System 2 reasoning. However, most people do not generally use System 2 thinking, at least not for most everyday decisions (Klein, 2008). The majority of people usually make decisions based on their gut feeling, thus using System 1 reasoning. We do not think systematically over moral questions, but rather follow our intuitions and previous experiences (Bennett & Flach, 2011; Klein). Experts strive for satisficing decisions rather than optimised (Bennett & Flach).

Poietic design supports autonomous thinking. Not by making people use System 2 thinking, but by designing artefacts in a way that improves people's gut feelings. As such, poietic design is not looking for the true meaning of autonomy, but rather make people act intuitively in an autonomous way.

With poietic design, users get an intuitive understanding of an artefact's or a system's status. With a better foundation of understanding, the decision-making process will be easier. By giving a better foundation, poietic design supports autonomous thinking because the users have more information available to them. At the same time, their need for System 2 thinking gets reduced. Thus, poietic design supports autonomous thinking without the need to be analytical.

Intuitions come from inductive inferences. One sees a consequence of an action or an event, and make predictions based on several observations. We create mental models. Mental, or conceptual, models can be described as models people have of themselves, others, the environment, and the things with which they interact (Norman, 2013). They are a person's understanding of how something works

(Laaksoharju, 2014; Norman, 2010). They are important to users for understanding how something works, even though the models do not necessary have to be correct, or even complete; as long as they are sufficient for their purposes they suffice (Laaksoharju). It is when the artefact does not behave accordingly to the user's (erroneous) mental model that trouble occurs. Therefore, the design should help users to form suitable mental models.

Intuitions are usually paired with System 1 thinking. However, one can be autonomous even if one has received information inductively. We make decisions automatically with System 1 thinking (for example the 'Asian disease problem', see Sunstein, 2005) and call it intuitive decision-making. But when we say 'intuitively', we usually refer to perception, not information retrieval. This means that intuition is not concerned with how we interpret information that we have already gained, but rather when perceive the information and before we actually interpret it. Intuition is not based on what we know, but rather what we have sensed.

One has to distinguish between cognitive processing and decision-making. System 1 and 2 thinking are concerned with the former and heteronomy and autonomy the latter.

As previously stated, System 2 thinking and autonomy are slower and more mentally taxing than their opposites. Can one be considered to behave autonomously if one does not use both System 2 thinking and autonomy? Perhaps both are not needed at all times. If one has to be analytical and systematic when retrieving information, one's cognitive resources get taxed. If one was to perceive information intuitively, the cognitive resources would be less strained, and more energy could be spent on using System 2 thinking for decision-making. This enables autonomy.

4.5 Poietic design and complexity

Like EID, poietic design is about showing the user more complexity of the artefact or system in order for them to get a deeper understanding of it and to empower them into being able to perform more actions. This might make the system look more complex, even though it is in fact not; the complexity is just re-distributed. By increasing the external complexity, the interaction complexity decreases. Just by looking at the poietically designed artefact it can be perceived as more complex, but when the user starts to understand what it means they will interact with it more easily. The point is that everything that can be interacted with needs to be visible. Norman describes the difference between physical and digital systems:

> *We usually can figure out what to do with physical systems because everything is visible so the alternatives are clear. electronic, computer-based systems need to do the same, to present sufficient information about how they work so that when something goes wrong, there is perceptible evidence of the problem and the possible alternative actions* (Norman, 2010, pp. 226-227).

It is of course possible that the interaction complexity increases when designing poietically. However, if done right, complexity should not pose a problem. Firstly, people enjoy a certain level of complexity (Norman, 2010), and we seldom want fewer alternatives for our technologies. Secondly, a goal for poietic design is to enable the user to behave more like an expert user, and expert users require a higher level of complexity. It should not be too simple, as it limits the users' actions. Thirdly, as previously stated, complexity and simplicity are not mutually exclusive (Norman).

It is conceivable that sometimes complexity is not wished for, however. Cooper et al. (2007) argue that user interfaces should be based on the user's mental model rather than the implementation model:

> *If we create represented models that are simpler than the actual implementation model, we help the user achieve a better understanding […] One of the most significant ways in which computers can assist human beings is by putting a simple face on complex processes and situations* (Cooper et al., 2007, p. 30).

This is the user-centred kind of thinking that poietic design attempts to challenge. While artefacts and systems that are only intended to be used once or twice probably should not be too complex, one should not dumb down the user and delimit them. By basing the interface on the users' mental models, their already existing misconceptions of the system will only be reinforced (Vicente, 1999). While it is important to not confuse the users with too much complexity, they should not be misled by incorrect simplicity either. "I do not claim that interface details have to be mirror images of the respective system functions but they should correspond distinctly to one. The interface should be an abstraction, not a distortion of the system model" (Laaksoharju, 2014, p. 57).

5 Examples of artefacts that communicate how they work

Interactive Institute Swedish ICT has invented ‘Power Aware Cord’ (sold under the name ‘The Pac’), an electrical cord that shows how much energy is used by an artefact in real time (Patmalnieks, 2016). It makes electricity visible by representing it with a ‘light flow’ (“The Pac,” n.d.), see Figure 1. The higher the power usage, the faster the light flows. This is an excellent example of poietic design; the user can after a while get an intuition of how much electricity that is spent, and the user has to decide by themselves whether it is positive or negative. The Pac does start to blink if it feels overloaded. This is not a poietic design trait, as it starts to tell the user when it deems that too much electricity has been spent instead of letting the user decide that for themselves. The feature makes sense for the Pac’s design since it is designed to promote sustainable behaviours with help of persuasive design.

Figure 1. The PAC. Image retrieved from Veckans Affärer (Jeppsson, 2015).

In his thesis, Gustafsson (2010) investigates how to promote sustainable behaviours in fun and enjoyable ways with the help of persuasive, ubiquitous, and playful energy feedback solutions. While his focus primarily lies on persuasive technology design, some of his goals align with those of poietic design. Two of the research questions he investigated were “Reveal the transparent and hidden use of energy” and “Intuitively and esthetically visualize and conceptualize energy usage” (Gustafsson, 2010, p. 61). Gustafsson points out the difficulty for people to understand concepts such as watts and kilowatt-hours, and how designers can visualise these imperceptible concepts:

> *Enable quicker routes from a presentation to an intuitive and emotional understanding of a quantity, provide a language for mutual understanding among, for example, inhabitants of a home, make energy more concrete and its use more graspable, and create associations to energy use* (Gustafsson, 2010, p. 64).

It seems like Gustafsson realises the importance of immediate feedback and giving the user an intuitive understanding of data representations. He also seeks to reveal imperceptible properties to humans by making them perceptible, as well as making the properties understandable to the users.

Another example of an artefact that clearly communicates how it works is a mill, like a windmill or a watermill. Its purpose might not be obvious at first sight, but we get an intuition of how it works by looking at it. The mill's arms' or wheel's rotation gives it kinetic energy that powers the mill's less obvious machinery. We even get information of how hard the wind is blowing or how much water is flowing from the speed of the mill's rotation.

A third, very simple example of a communicative artefact is a knife. It gives the user immediate feedback when used, and we can observe its effects of its use to see whether the knife is sharp or dull.

None of these examples might be a perfect example of poietic design. However, they do provide concrete examples of some poietic attributes which we can learn from. These traits can furthermore be used to further develop poietic design.

6 Properties of poietic design

Through discussions and iterations with Mikael Laaksoharju, a list of poietic properties has been found and refined during the progress of this thesis. A poietically designed artefact or system must possess these properties. They are as shown below:

- Affords a developing intuitive understanding of its complexity: The user does not necessarily need to understand the artefact's full complexity in the beginning, but over time they should be able to understand it on an intuitive level. The artefact should not appear to be more or less complex than it actually is.
- Informative: The artefact should be informative, that is, it should communicate information in a way that suits human perception. The information should be perceptible to the user and give them an estimation of it. This is not to say that the information perceptible should be inexact; it just does not have to be shown to the last detail. The user seldom needs a "high resolution" visualisation of the information, but it should be possible to get if needed.
- Transparent and honest representation of functionality: Everything that is possible to do should be discoverable. The user should be able to understand the reason(s) for overlapping functionality when applicable.
- Not judgmental or preaching: The artefact does not tell the user whether their actions are good or bad; it lets the user decide that for themselves.
- Gives immediate feedback: The user should receive immediate feedback on their actions.
- Consistent: The artefact's design should be consistent in order to help the user to understand the effects of their actions in order to allow the development of intuitions. Consistency will help the user to form a useable mental model after a period of using the artefact. It will also support autonomous decision-making by reducing the need for using System 2 thinking.
- Perceptible properties: All of the artefact's properties should be perceptible to the user. Not everything has to be immediately visible, but everything should be able to be found after some exploration.
- Affords development of intuitions about the effects of actions: The user should be able to form intuitions about the effects of actions after a period of using the artefact. Even if the user might not fully understand what is going on in the system, they should have a feeling of what is happening.

These properties will be referred to as the heuristics of poietic design.

7 Cases

As previously mentioned, the purpose of the cases is to give an example of the application of poietic design. It will be illustrated that poietic design not only can be used in designing user interfaces, but also to improve them.

In Case 1, a software is described and analysed based on the poietic heuristics listed above. From the conclusions of the evaluation, a solution designed using the heuristics of poietic design is proposed. Note that neither of the designs are tested in this thesis. The reason for this is primarily because of time constraints; the main focus of the thesis is Case 2. Instead, the main purpose of Case 1 is to give an idea of poietic design in practice, that is, how a poietically designed interface could look like.

In Case 2, a display designed based on EID principles is described and analysed based on the poietic heuristics. The experiment in which the display had a part in is also discussed. Following that, a poietic display is proposed and tested in a replication of the experiment where the EID display was originally tested. The EID display and the poietic display are compared and the results are discussed.

7.1 Case 1: Bulk Rename Utility

Observe the figure below (Figure 2). The figure depicts a file renaming software interface named Bulk Rename Utility ("Bulk Rename Utility," n.d.). The purpose of the software is to enable people to change names on several files at the same time, instead of having to change the names one by one, as is custom for window interfaces. While it has got overall positive reviews by users ("Download.com," n.d.), there is no denying that the interface could look rather scary and overwhelming to novice users.

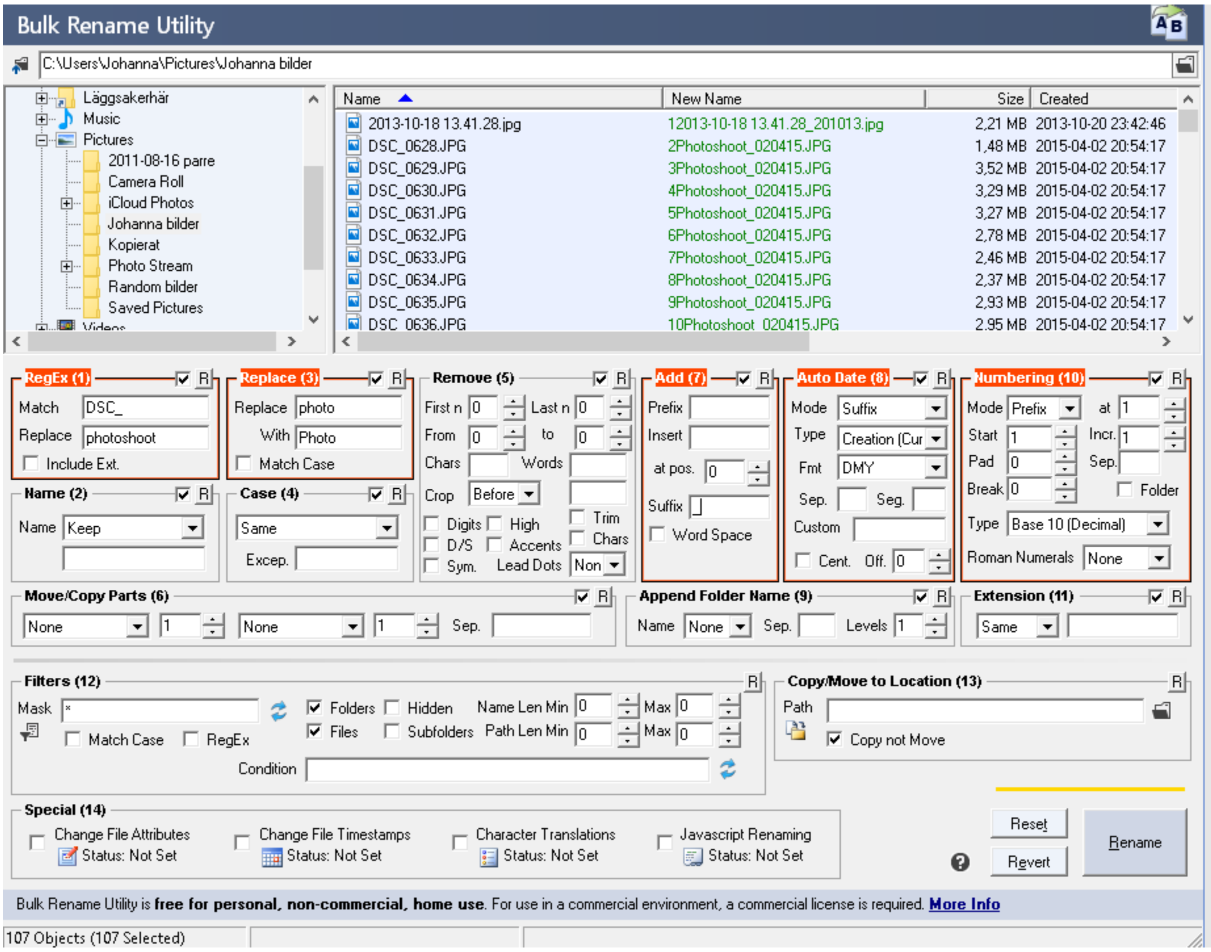


Figure 2. A screenshot of the software Bulk Rename Utility.

On the good side, Bulk Rename Utility empowers and enables the user and is not afraid of complexity. On the other hand, the solution is neither elegant, intuitive, or reflects its level of complexity accurately.

The goals of poietic design align with the ones of this kind of software, one that gives more control and freedom to the users. Nevertheless, Bulk Rename Utility lacks some desirable traits. Basically, we want the software to offer the same possibilities it already does – but in a more poietic way.

What poietic traits does the software already possess? Let us go through the list of previously suggested poietic properties:

- Affords a developing intuitive understanding of its complexity: The user probably does not get a developing intuitive understanding of the software, as all of the name changer menu alternatives are visible at all times. As such, while all possible actions are visible, the user does not develop an understanding of the software's complexity. With time and attempts, the user will learn what changes the different fields do, but not exactly in an intuitive way.
- Informative: The software is informative in the way that it communicates the changes made to the name in a clear and visible way.
- Transparent and honest representation of functionality: The software has indeed a transparent and honest representation of its functionality. It lets the user control exactly where a change in a file name will take place by typing in character numbers. There is no margin of error whatsoever. However, for an inexperienced user there is no telling of why some fields do the same changes as others, and why one cannot do some changes. For example, for a layman user, the 'RegEx' and the 'Replace' alternatives seem to do the same thing, there is an overlap in functionality and nothing in the software tells the user what they have in common. The 'RegEx' or 'Regular Expression' function is more powerful and malleable than the 'Replace' function, since 'RegEx' can match and replace characters using different sets of rules. The expert user knows that 'Replace' is a so-called comfort function for 'RegEx'. The 'Replace' function is just one of the 'RegEx' functionality. However, this is not shown or explained by Bulk Rename Utility. As such, the novice user's mental model will not work here.
- Not judgmental or preaching: The software is neither judgmental nor preaching since it does not tell the user what to do.
- Gives immediate feedback: The software does give immediate feedback on changes. Even though the user has to confirm the changes in order for them to happen, the preview of a change of the file names is shown accurately after the user has changed a parameter.
- Consistent: The software is consistent. The empty fields allow the users to write freely, the dropdown menus allow them to choose one item from the menu, the checkboxes can either be marked or not, and so on. Because of this, the design supports a mental model and autonomy for the user.
- Perceptible properties: All of the software's properties are perceptible. The software makes it clear what fields have been tampered with (with an angry red framing), and the file names that will be changed have their new names spelled out with a green text.
- Affords development of intuitions about the effects of actions: With time and some trial and error from the user's side they will probably develop an intuition about a certain action's effect. Although one could argue that the user just learns what fields do what changes.

With a poietic design, we want the same possibilities that the Bulk Rename Utility software grants the user, but in a slightly other way. The user should be able to get an intuition of how to use the software, and be able to learn it. The common user should be able to learn to use the software like an expert user.

7.1.1 A poietic redesign: The name template

A proposal of a poietic redesign was conceived after discussions and reflections about poietic design with Mikael Laaksoharju.

A more poietic solution could be to enable the user to change all file names in a folder (alternatively all marked files) by right-clicking (Figure 3). For example, if the user right-clicks when inside a folder, an alternative for renaming all files (or all marked files) could show. When choosing it, a name template shows up (Figure 4). Another way is to show the name template directly in the folder, for example at the top of the files.

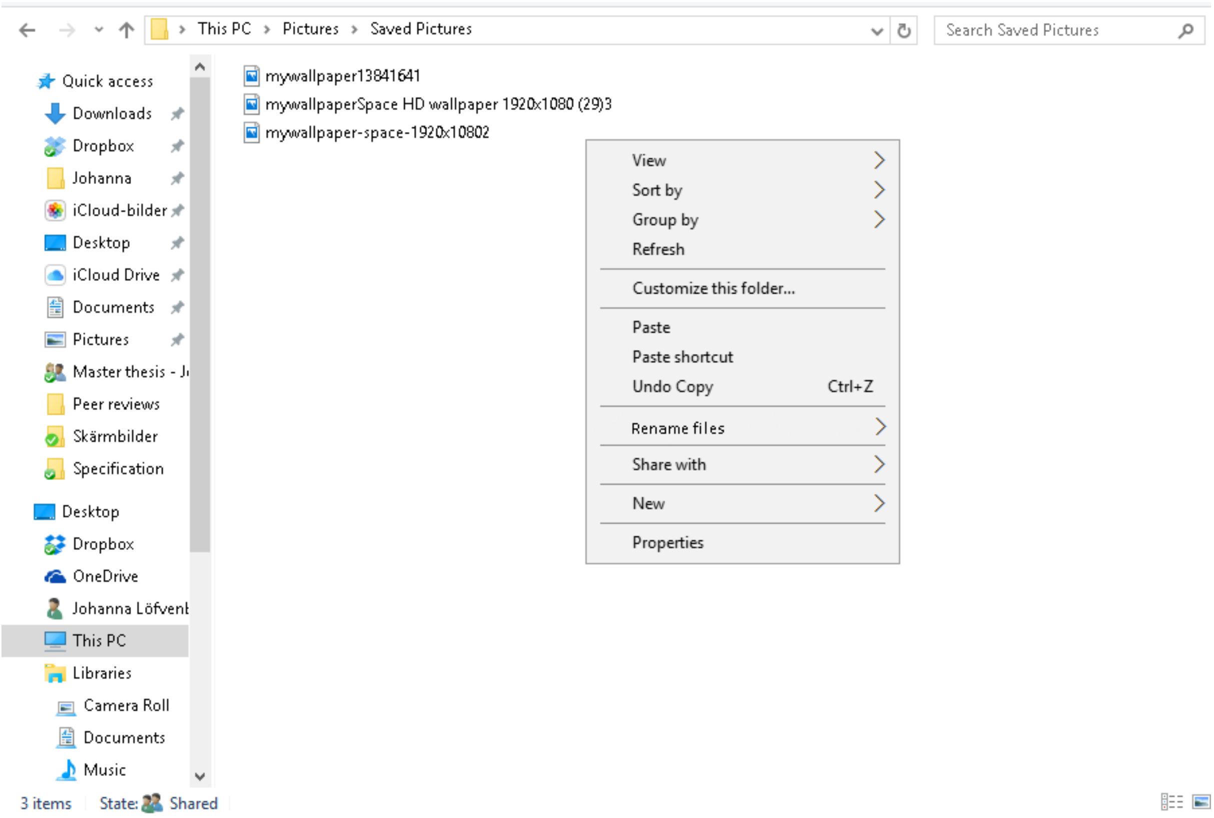


Figure 3. When the user right-clicks in the folder, the alternative 'Rename files' is available. When the user clicks on it, a name template shown in Figure 4 below shows up.

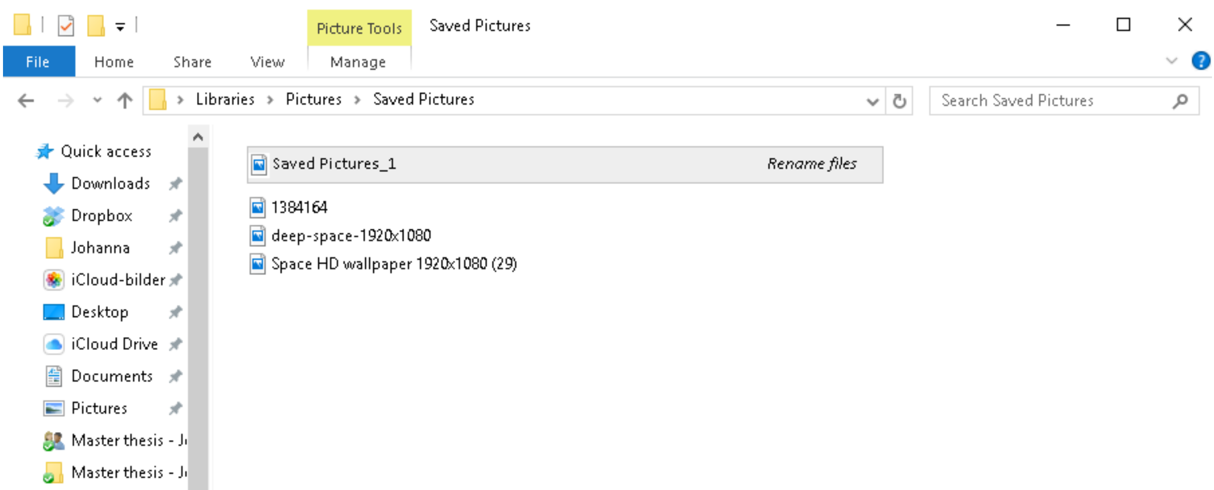


Figure 4. The name template (see text). The user will see this when clicking on the 'Rename files' alternative (Figure 3). Alternatively, it could be shown in the folder at all times.

If the folder is called 'Vacation Paris 2016', the name template could for example spell out 'Vacation Paris 2016_1'. The user can change the name 'Vacation Paris 2016' and add a number before or after the file's name. The files will get a unique number, with the first file getting 1, the next 2, and so on. The folder's name should be the standard template name for the folder's files, because most people will probably want to have the folder's name to describe the contents of the files.

This poietic solution is perhaps not better than Bulk Rename Utility, maybe it would not even work in practice. The redesign is untested, after all. Furthermore, it is not clear from the name template that the number will be unique for every file in the folder. And what if one wishes to name the folders with a number? In the current redesign it would be difficult to distinguish the numbering from the name if the name consisted. That could probably be solved with some sort of dropdown menu (Figure 5), but the issue that the solution is untested remains. It should also be noted that the redesign is not concerned with the 'RegEx' function, as it is a complex function that would need more time and thought to integrate in the design. It is a powerful tool for those who know how to use it, and it is a limitation in the redesign to not have one. However, the point is that with time and more careful designing, the poietic redesign could be just as powerful as Bulk Rename Utility, while being more intuitive to use.

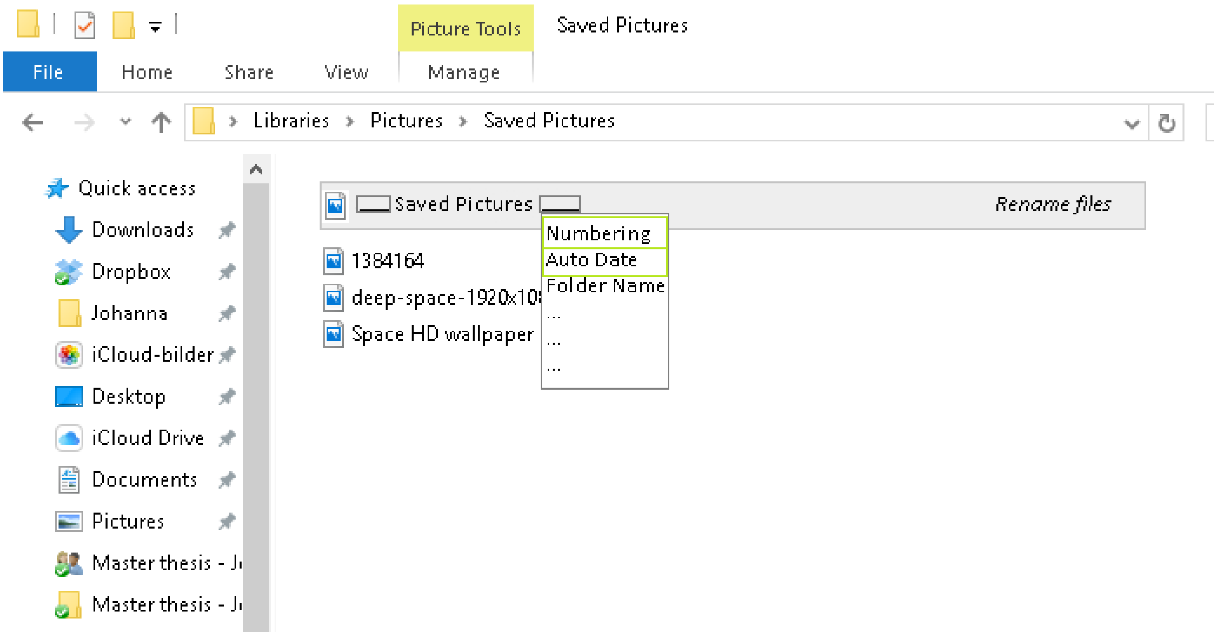


Figure 5. A possible solution to the numbering problem (see text). There is a dropdown menu before and after the changeable name (in the picture, 'Saved Pictures'), where all sorts of name changing functionalities are. The user can select as many options as they wish.

This solution is more poietic than the current software. The differences are shown in the following list:

- Affords a developing intuitive understanding of its complexity: The user has to try out the new solution, but it would probably be easy for the user to develop an understanding of what it can offer.
- Informative: The files' new names are not immediately visible like in Bulk Rename Utility (although it would be trivial to add a 'New name' headline), but the name plate's name is visible during the whole name-changing process. This makes the design informative.
- Transparent and honest representation of functionality: The solution has partially a transparent representation of its functions. The possible actions might not always be immediately visible to the user, but all of them are easily discoverable. As opposed to Bulk Rename Utility, which looks quite complicated at first sight, the solution does not look more complicated than it actually is.
- Not judgmental or preaching: The software is neither judgmental nor preaching.

- Gives immediate feedback: The solution gives immediate feedback on changes. Even though the user has to confirm the changes in order for them to happen, the preview of a change of the file names is shown accurately after the user has changed a parameter.
- Consistent: The solution is consistent. The file name template shown in Figures 4 and 5 shows what the file names will look like. All of the files will be named the same with the exception of the numbering. This supports the user's mental model and autonomy.
- Perceptible properties: While not all of the software's properties are immediately perceptible, they all can be after some exploring.
- Affords development of intuitions about the effects of actions: It would probably be easy for the user to develop intuitions about the effects of their actions, as the possible actions are few. As opposed to Bulk Rename Utility, one works 'directly' with the files' names; one does not have to find the correct field or dropdown menu to make the wanted change, one just has to change directly where in the name one wants the change to be.

In conclusion, the Bulk Rename Utility software does empower and enable the user and has some traits in common with what a poietically designed software would possess, but some more changes in a poietic way could likely improve the user experience. The suggested poietic solution could likely improve the user's intuitive understanding of the software's complexity and that of the effects of their actions.

7.2 Case 2: Configural coordinate display

In a recent paper, Holt, Bennett, and Flach (2015) conducted an experiment to evaluate the performance of four displays. Three of them originated from a study from 1989 by Coury, Boulette, and Smith. The displays were an alpha-numeric table, a bar graph, and a polar graphic. The fourth display was designed with a triadic approach in mind rather than the dyadic approach by Holt et al. themselves. The authors' 'configural coordinate' display was thus designed by using principles of EID. The configural coordinate display was designed to achieve perception of a system state in a more direct fashion (Holt et al.). "A single point is calculated to simultaneously capture differences between each pair of relevant variables" (Holt et al.).

The results showed that the configural coordinate display was the most effective display, especially in terms of latency and accuracy (Holt et al., 2015). This was believed to be due to that the display supported state identification effectively; the users did not have to perform any mental calculations, nor any fine perceptual discriminations (Holt et al.). Thus, the configural coordinate was the most effective display due to the more direct mappings between display, perceptual, and domain constraints (Holt et al.).

A problem with the configural coordinate display that the authors mention themselves is that the user loses information about the environment. "The configural coordinate display hides the value of the four individual variables and the ways which each of them contribute to the overall system state" (Holt et al., 2015). In a complex, dynamic work domain or environment, an understanding of the value of individual variables and how they contribute to higher order properties is often essential (Holt et al.).

The reason why the user loses information about the environment is that, unlike the other three displays, the configural coordinate display presents an already calculated result to the user. In the three original displays, the user has to mentally calculate the difference between two system variables to find out the system state, as they display the original numbers for the system variables. The configural coordinate display only shows the already calculated difference of the system variables in the form of a single point. Hence, the authors' EID configured display was designed in a way that fit the task. As such, one could argue that the configural coordinate display cannot be tested fairly along with the other displays, as they do not display the same kind of data. They are incomparable.

In an attempt to remedy this issue, the authors redesigned the configural coordinate display (Figure 6) so that it also shows the original numbers of the system variables like the other displays, or in other words, to “support performance at both divided and focused attention tasks” (Holt et al., 2015). The redesign has basically been equipped with bar graphs similar to those in the bar graph display (the most effective display next to the configural coordinate display according to the results of Holt et al.). However, while the redesigned display is a less streamlined solution for a specific task, Holt et al. do not test its performance in any way. Furthermore, the redesign builds on the configural coordinate display rather than actual data. Arguably, it is natural that a redesign is built on its original design, but the redesign seems to suffer from almost all of its original design’s limitations in this case.

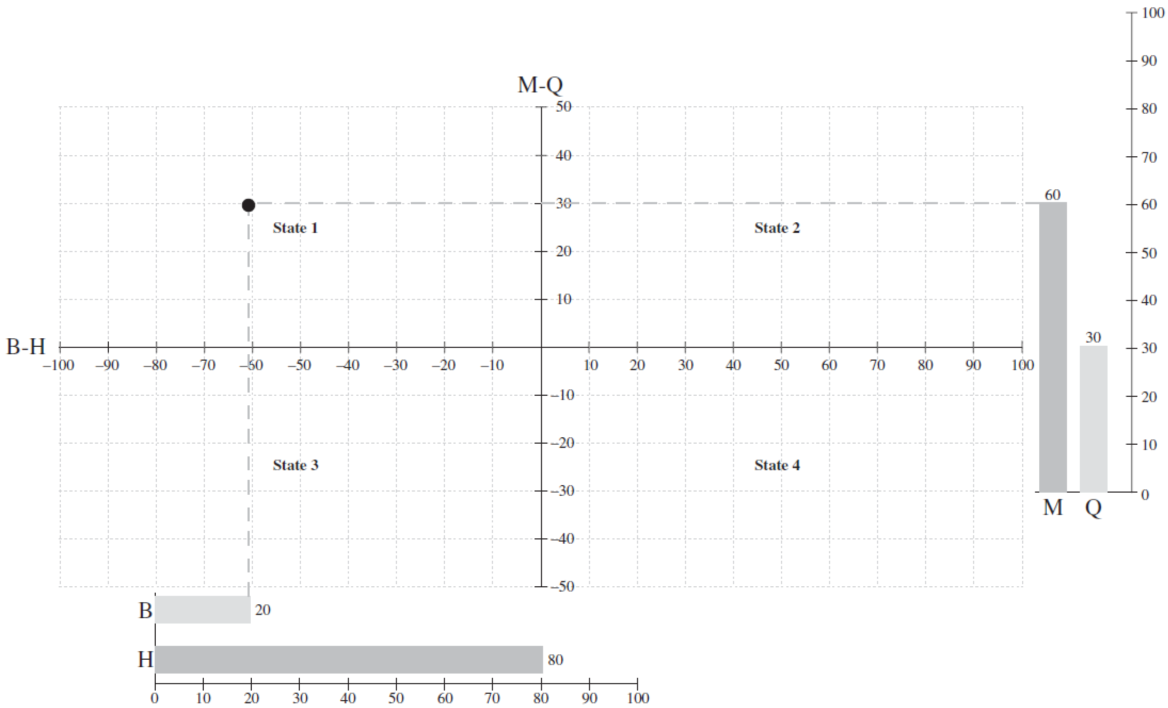


Figure 6. The redesigned configural coordinate display by Holt, Bennett, and Flach (2015). Reprinted by permission of Taylor & Francis Ltd. Copyright Clearance Center Inc., www.tandfonline.com.

The authors conclude the paper by arguing that their results reemphasise the necessity of a triadic approach for display and interface design (Holt et al., 2015). They also underline the importance of taking the work domain in consideration (Holt et al.). While both of these conclusions are very in line with EID goals, their display design is not completely along the same line. From the article it seems like the primary goal of EID is just to give users a direct perception of data. Naturally, one article cannot decide this by itself, but both co-authors Bennett and Flach are big names in the field of EID, so presumably they have much to say regarding EID. It seems like their solution feels slightly at odds with their own definition of EID:

> *The goal of an ecological interface is not to take a complex domain and make it seem naturally simple to a naïve user. Rather, the goal is to allow the user to discover and learn the consistencies and eventually to develop a rich network of associations that will support good situation awareness and fast, frugal decision making* (Bennett & Flach, 2011, pp. 104-105).

An EID display should not be easier to use by reducing the simplicity of the display. In EID one is not afraid to show complexity and more information. The user should understand where the difference is coming from.

In order to further illustrate the differences between EID and poietic design, the configural coordinate display is analysed using the heuristics of poietic design:

- Affords a developing intuitive understanding of its complexity: While the user will soon learn to identify the system states depending on where the dot is situated, it might prove to be a difficult task to develop an intuitive understanding on how the four variables define the state. This could be difficult due to that the user will have to focus their attention on three spots in the display, the dot, the diagonally situated variables, and the horizontally situated variables.
- Informative: As mentioned above, the user might need to focus on three spots, perhaps simultaneously, were the display dynamic. Humans are not good at dividing their attention, something several experiments from cognitive science have shown (e.g. the 'dichotic listening task'). It is possible to focus on several tasks at once as long as the information makes use of different senses and we are used to the tasks (e.g. driving a car). However, it is difficult to look at and focus on more than one thing at a time (e.g. texting and driving).
- Transparent and honest representation of functionality: With the addition of the bar graphs all of the data should be discoverable.
- Not judgmental or preaching: The artefact is neither judgmental nor preaching. Since the data is made-up, no system state is perceived as preferable over the others.
- Gives immediate feedback: Since the user is not able to interact with the visualisation, no feedback is given. This should not be deemed un-poietic; the heuristic is simply not applicable.
- Consistent: The display is consistent with how it displays the data. It should be possible to form a mental model of how changes of the variables' values affect the system state. This supports a useable mental model. However, the dot interprets the data for the user, which does not support autonomy. On the other hand, the 'original data' is also shown in the form of the bar graphs, which supports autonomy.
- Perceptible properties: All of the display's data is visible to the user.
- Affords development of intuitions about the effects of actions: Since there is no way of interacting with the display, this heuristic is not applicable to this case.

7.2.1 A poietic redesign: The box

Consider the citation from Bennett and Flach (2011) on the previous page. The similarities between the goals of EID and poietic design are that both aim to allow the user to discover and learn consistencies, which will make them develop associations. While poietic design is also concerned with good situational awareness and fast, frugal decision-making, they are not goals in themselves. Those qualities are rather seen as consequences of poietic design.

The fundamental difference between EID and poietic design is that EID has an actual goal with its design. In EID, the design tries to presuppose its user's goals, which leads to designs such as the one in Figure 6, where essentially only the one task that was used in the experiment by Holt et al. (2015) can be performed. The purpose of a poietic design would be to enable an intuitive understanding of the data, without having any presupposed knowledge of the user's intentions. In summary, poietic design does not presuppose a goal, but strives after bringing forth the essence of the artefact.

The goal with a poietically redesigned display would be to maximise the interpretability of the underlying data, to translate the data into something we can perceive. As opposed to the EID display, there is no need to analyse the data for the user in advance. Instead the user should be able to learn to interpret the data by themselves.

As previously mentioned, exact numbers are not a key point for poietic design. An intuition of the situation is of more importance. Consequently, the poietically designed display gives the user a feeling of the relationship between the four variables and in what state is displayed (see Set-up below for more details), rather than providing them with the exact numbers of the variables.

Observe Figure 7 below. This box-like design makes use of graphs, just like the redesigned EID display. The graphs originate from the middle of the display, the origo, and a light grey area in the shape of a rectangle surrounds them. For more information on how to interpret data from this design and the others, see Method below. The design was suggested by Mikael Laaksoharju as a poietic redesign of the configural coordinate of Holt et al. (2015).

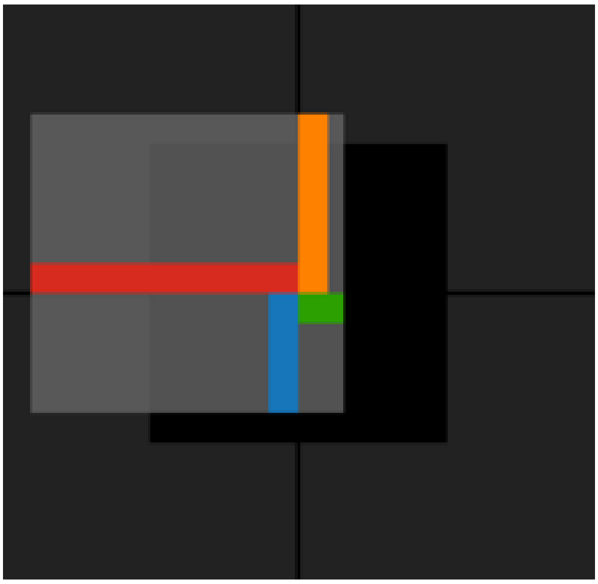


Figure 7. A screenshot of the poietic display that was used during the experiment. The display was designed by Mikael Laaksoharju.

Let us compare the redesign with the heuristics of poietic design:

- Affords a developing intuitive understanding of its complexity: Based on informal pilot tests, it is definitively possible to develop an intuition of what the different parts of the design mean. The four variables are close to each other, and the light grey area surrounding them forms different shapes depending on the height of the variables. This should make it possible to compare and interpret data, and develop an understanding of it.
- Informative: Had the display been dynamic, the user would just have one area to focus their attention on. As opposed to the EID display, all changes of data in the poietic display would just make the light grey area to move around along with the changes of the variables. This works well for humans as we tend to just focus on one thing at a time.
- Transparent and honest representation of functionality: The display shows everything that the user needs to know.
- Not judgmental or preaching: The display is neither judgmental nor preaching, especially since the data does not mean anything as it is made-up.
- Gives immediate feedback: Since the redesign is just an information visualisation and one cannot interact with it, no feedback is given.
- Consistent: The display is consistent in the way it presents the data. It should be possible to form a mental model of how the four variables affect the system state. This supports a useable mental model. The display does not analyse or attempts to analyse the data for the user in advance. This supports autonomy.
- Perceptible properties: All the properties of the data are visible.
- Affords development of intuitions about the effects of actions: Since the redesign is just an information visualisation and one cannot interact with it, this attribute is not applicable to this case.

It could be considered a limitation that only four variables fit in the design, but the same can be said about the EID design, too. Because of this, and due to the experiment is a duplication of the one of Holt et al. (2015), this limitation of design will be overlooked.

7.2.2 Method

An experiment was conducted in order to investigate both whether the EID design really is as good as the results from Holt et al. (2015) say, or if it is as limited as it seems, and if a poietically designed display is less limited and performs better than the EID design.

7.2.2.1 Participants

The test participants were recruited on campus, among acquaintances, and via Facebook with a promise of cinema ticket vouchers as a reward for their voluntary assistance. They were 26 in total, with 11 women and 15 men. Their ages varied between 22-46 years, with a median of 26 years. All of the participants rated their English knowledge as fairly good or better. All participants were aware that the experiment was voluntary and that they could stop at any time. While they were not aware of the details, they knew the general purpose of the experiment.

7.2.2.2 Equipment

The program used in the experiment was written in JavaScript, and made so that three different options could be run (see Set-up below).

The experiment was conducted in a room equipped with 18 computers with accompanying mice and keyboards. All computers used Windows as operating system. The experiment ran on Mozilla Firefox as web browser for all test participants. The outline of the room can be seen in Figure 8. The same room was used on both occasions of the experiment.

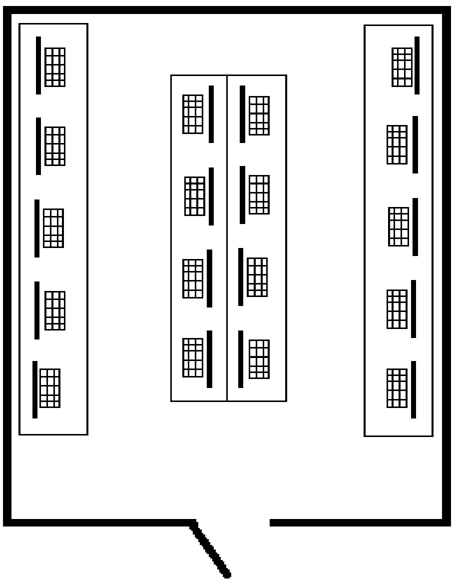


Figure 8. The outline of the room that was used for the experiment.

7.2.2.3 Set-up

The experiment used a mixed design. There were three groups of participants, and two tasks. Each group of participants got tested on a different display design; the poietic redesign (Figure 7), the EID design by Holt et al. (2015) (Figure 9), and a bar graph display that served as a neutral reference (Figure 10). All groups performed both tasks, but just with one kind of display each.

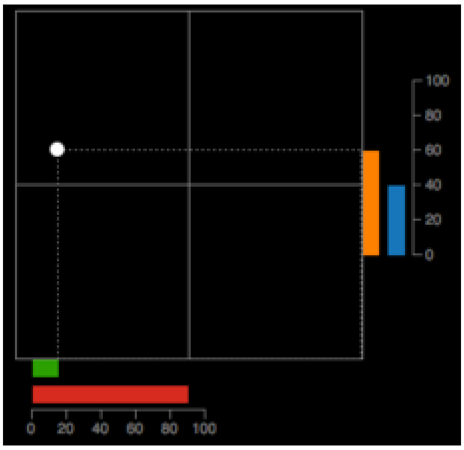


Figure 9. The EID design adapted from Holt et al. (2015).

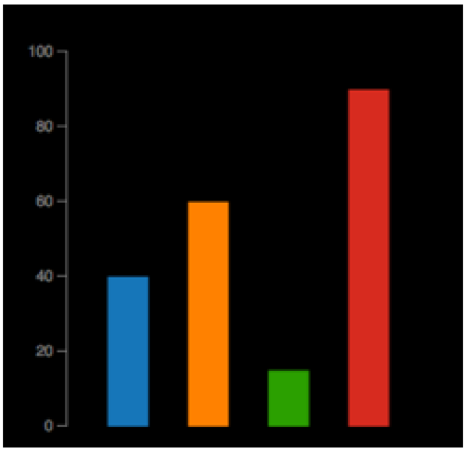


Figure 10. The bar graph.

Task 1 was to identify the correct system state out of four possible. This was to be done as quickly and as accurately as possible. In the poietic display, the system state is decided by finding out which of the four squares the centre of gravity of the light grey area lies (Figure 7). In the EID display, the system state is decided by in which of the four squares the dot is located in (Figure 9). In the bar graph display, the system state is decided by the relative size difference between the first and the second bar as well as the relative size difference between the third and the fourth bar (Figure 11). All three figures above depict system state 1.

Task 2 was also to identify the correct system state out of four possible as quickly and as accurately as possible. However, in this task, the state was determined by the sum of two variables divided by the sum of the other two:

$$(A + B) / (C + D)$$

In the EID display and the bar graph display, the system state is calculated by adding the lengths of the adjacent blue and orange bars and divide that with the combined lengths of the other two adjacent bars, the green and the red ones. For example, it was decided that if (A + B) is slightly shorter than (C +D), the system state is 1. In the poietic display the same rules apply, though one can simply determine the shape of the light grey area in order to determine the system state. For example, if the width of the light grey area is slightly shorter than height, the system state is 1.

The purpose of Task 1 was to replicate the experiment conducted by Holt et al. (2015), as well as to investigate if a poietically designed display performs as well as or better than the EID display. The purpose of Task 2 was to test the displays in a task that fits better to the purpose of poietic design. Task 2 is shaped to be more intuitive and less exact compared to Task 1, meaning the participants do not have to be as aware of the exact values of the bars, just get an intuition of the system states. Task 2 also served the purpose to test whether the EID display truly is a more limited display than the poietic display.

The complete instructions for the experiment that the test participants received are shown below (Figures 11-16).

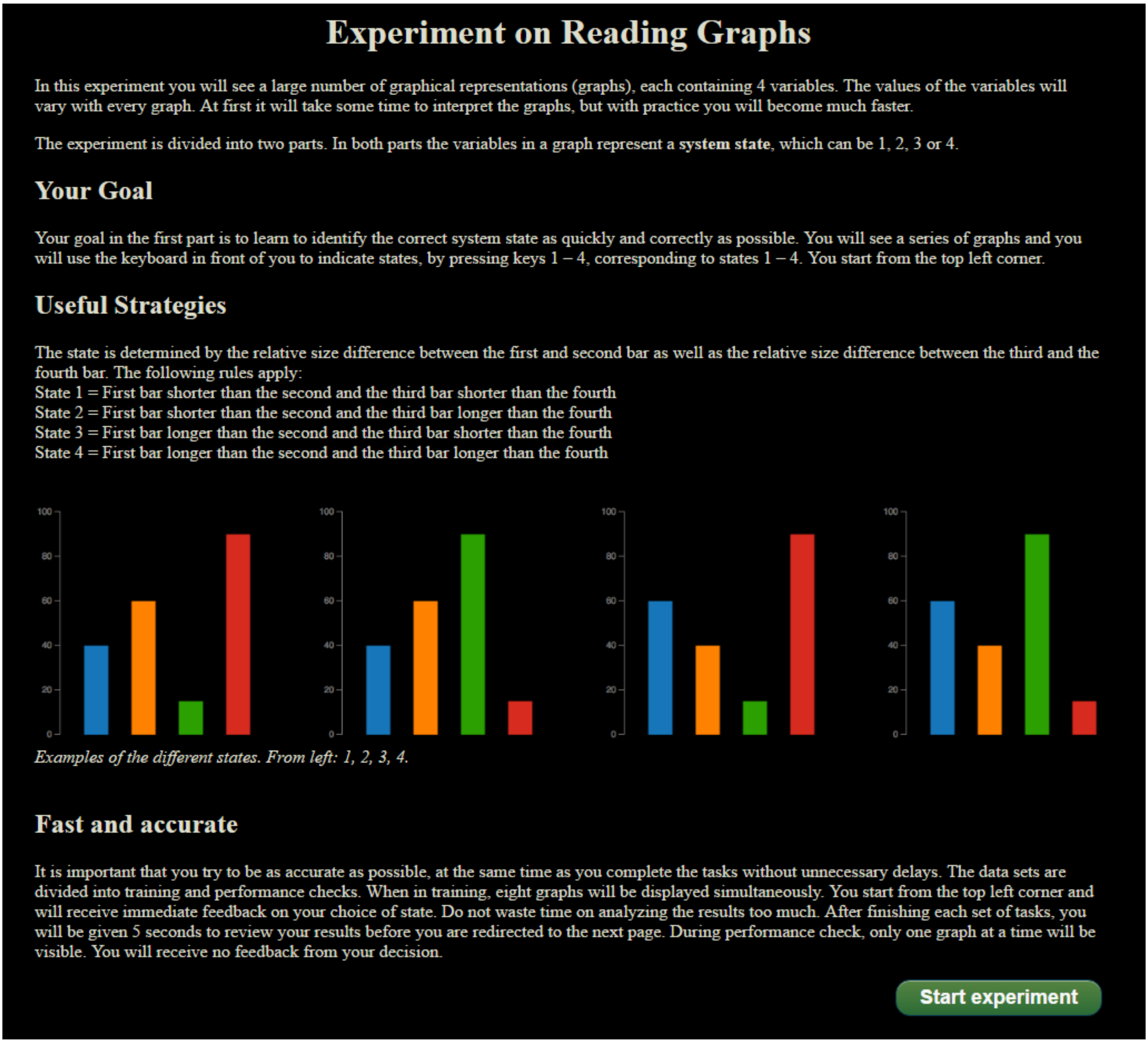
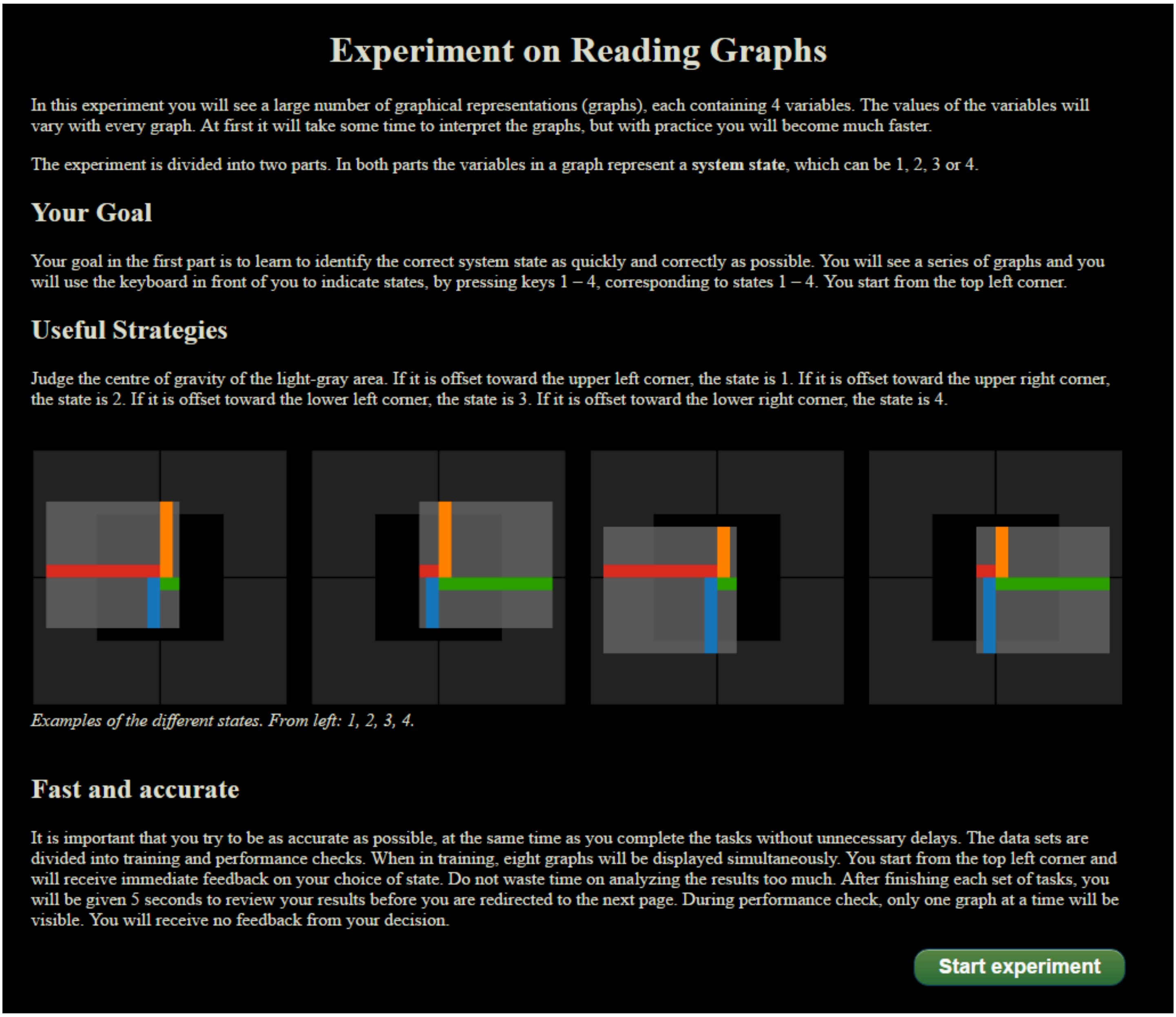


Figure 11. The instructions for the bar graph display, Task 1.


Experiment on Reading Graphs

In this experiment you will see a large number of graphical representations (graphs), each containing 4 variables. The values of the variables will vary with every graph. At first it will take some time to interpret the graphs, but with practice you will become much faster.

The experiment is divided into two parts. In both parts the variables in a graph represent a **system state**, which can be 1, 2, 3 or 4.

Your Goal

Your goal in the first part is to learn to identify the correct system state as quickly and correctly as possible. You will see a series of graphs and you will use the keyboard in front of you to indicate states, by pressing keys 1 – 4, corresponding to states 1 – 4. You start from the top left corner.

Useful Strategies

Judge the centre of gravity of the light-gray area. If it is offset toward the upper left corner, the state is 1. If it is offset toward the upper right corner, the state is 2. If it is offset toward the lower left corner, the state is 3. If it is offset toward the lower right corner, the state is 4.

Examples of the different states. From left: 1, 2, 3, 4.

Fast and accurate

It is important that you try to be as accurate as possible, at the same time as you complete the tasks without unnecessary delays. The data sets are divided into training and performance checks. When in training, eight graphs will be displayed simultaneously. You start from the top left corner and will receive immediate feedback on your choice of state. Do not waste time on analyzing the results too much. After finishing each set of tasks, you will be given 5 seconds to review your results before you are redirected to the next page. During performance check, only one graph at a time will be visible. You will receive no feedback from your decision.

Start experiment

Figure 12. The instructions for the poietic display, Task 1.

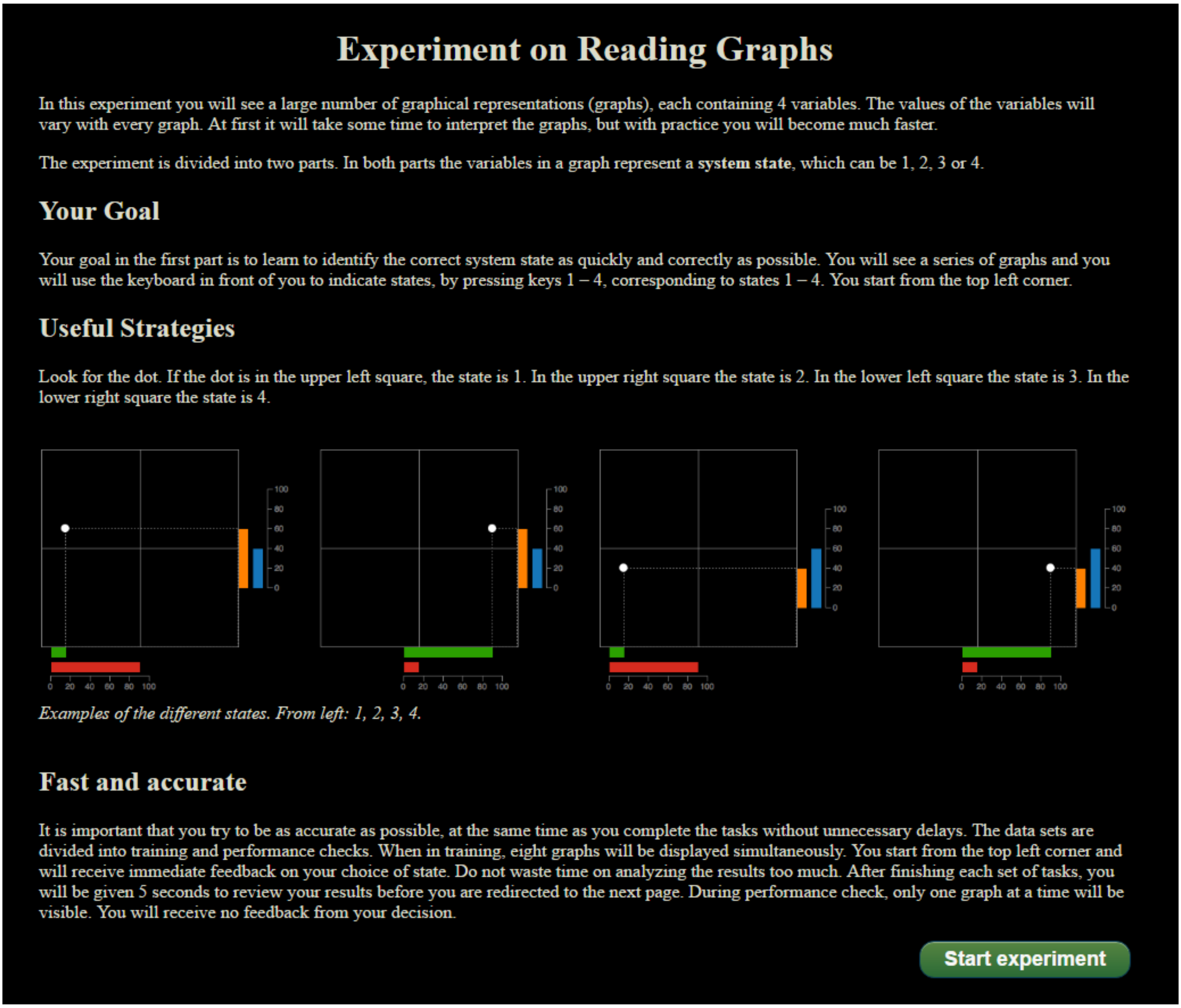


Figure 13. The instructions for the EID display, Task 1.

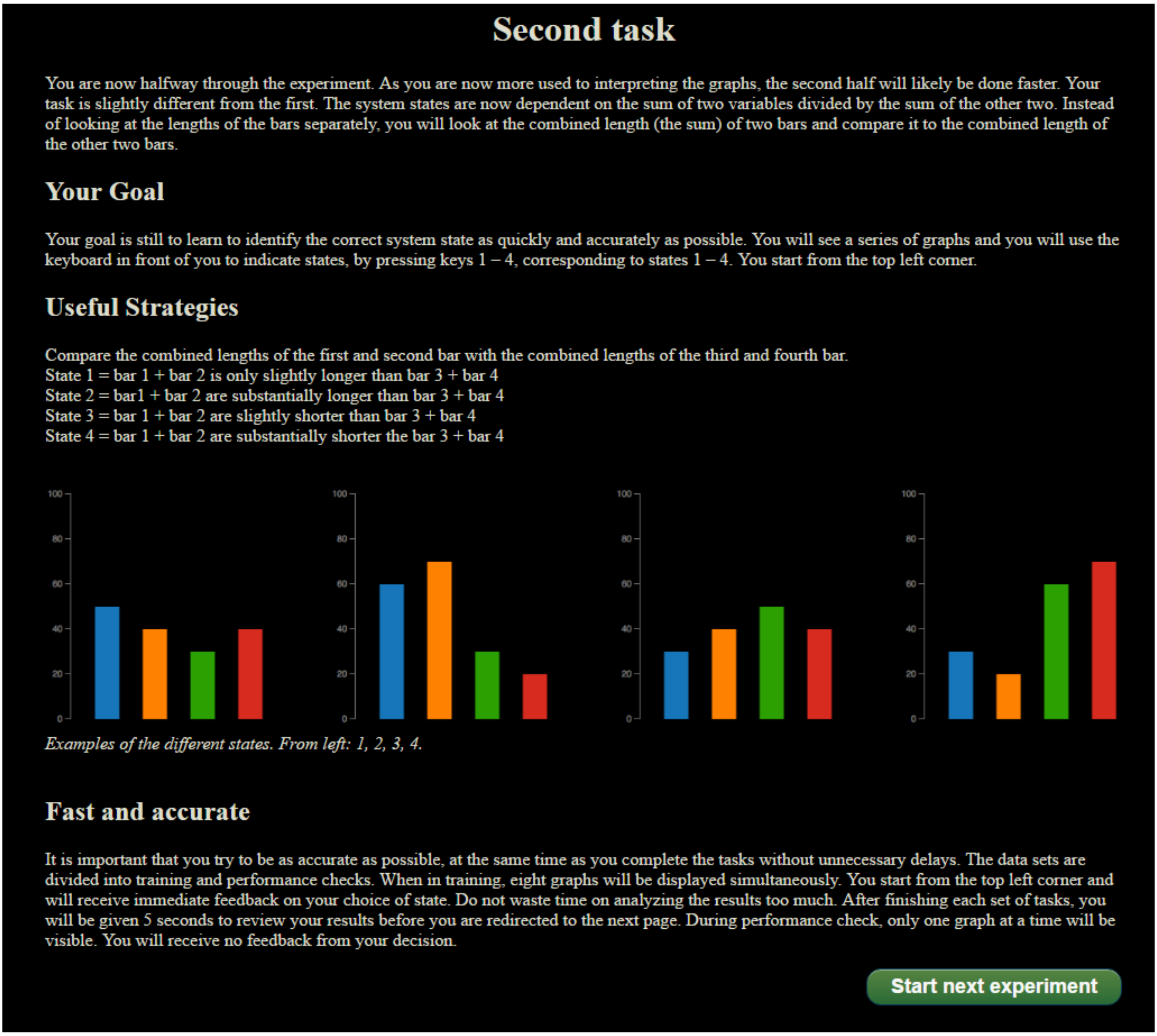
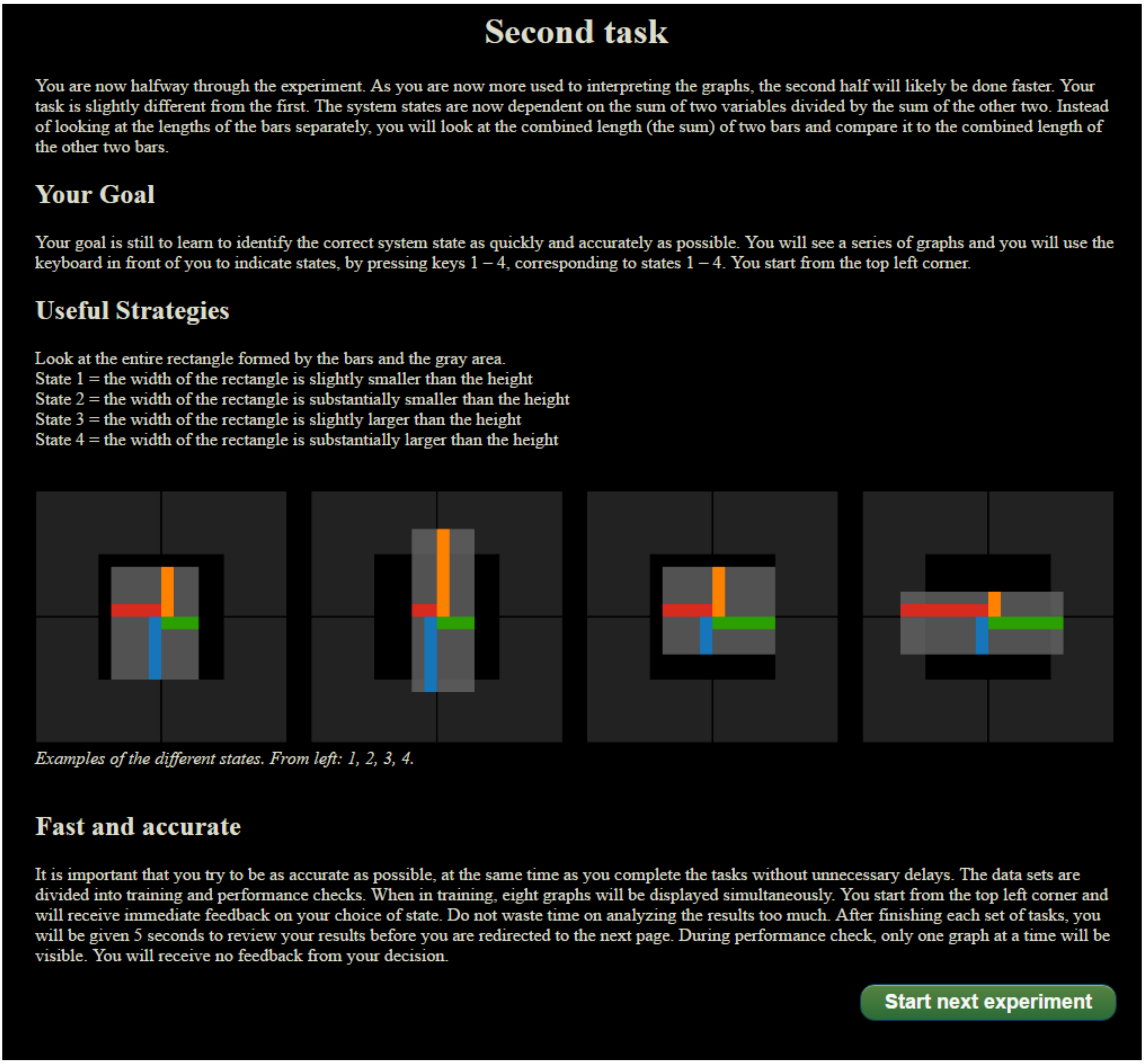


Figure 14. The instructions for the bar graph display, Task 2.


Second task

You are now halfway through the experiment. As you are now more used to interpreting the graphs, the second half will likely be done faster. Your task is slightly different from the first. The system states are now dependent on the sum of two variables divided by the sum of the other two. Instead of looking at the lengths of the bars separately, you will look at the combined length (the sum) of two bars and compare it to the combined length of the other two bars.

Your Goal

Your goal is still to learn to identify the correct system state as quickly and accurately as possible. You will see a series of graphs and you will use the keyboard in front of you to indicate states, by pressing keys 1 – 4, corresponding to states 1 – 4. You start from the top left corner.

Useful Strategies

Look at the entire rectangle formed by the bars and the gray area.
State 1 = the width of the rectangle is slightly smaller than the height
State 2 = the width of the rectangle is substantially smaller than the height
State 3 = the width of the rectangle is slightly larger than the height
State 4 = the width of the rectangle is substantially larger than the height

Examples of the different states. From left: 1, 2, 3, 4.

Fast and accurate

It is important that you try to be as accurate as possible, at the same time as you complete the tasks without unnecessary delays. The data sets are divided into training and performance checks. When in training, eight graphs will be displayed simultaneously. You start from the top left corner and will receive immediate feedback on your choice of state. Do not waste time on analyzing the results too much. After finishing each set of tasks, you will be given 5 seconds to review your results before you are redirected to the next page. During performance check, only one graph at a time will be visible. You will receive no feedback from your decision.

Start next experiment

Figure 15. The instructions for the poietic display, Task 2.

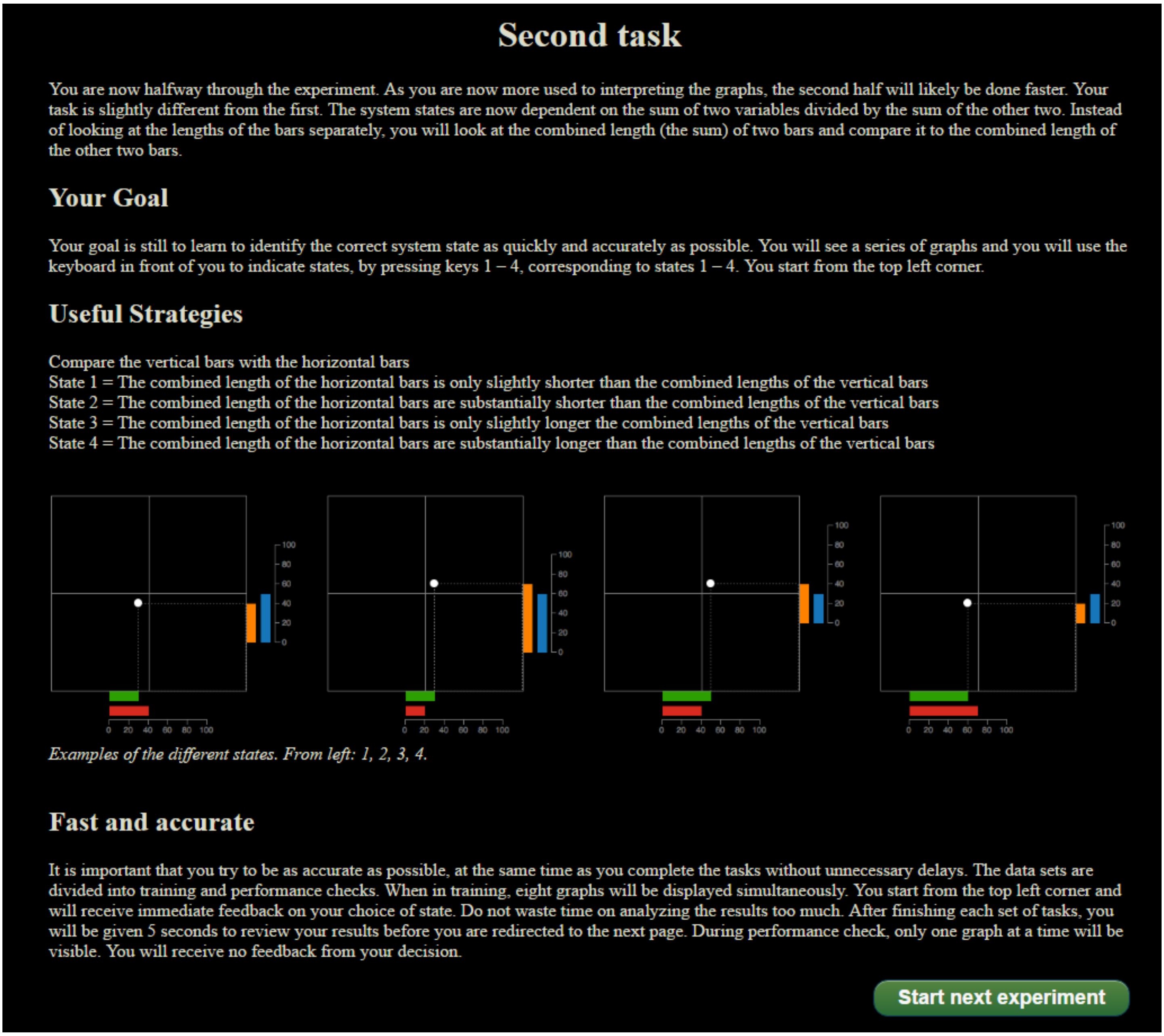


Figure 16. The instructions for the EID display, Task 2.

7.2.2.4 Hypotheses

It was hypothesised that the EID design would perform well in the task it was designed for (Task 1), but not so well for the task that it was not designed for (Task 2). It was also hypothesised that the poietic design would perform better than the bar graph in both tasks and as good as the EID task in Task 1, and better than the EID design in Task 2. The hypotheses can be summarised as follows:

1. EID is faster than Bar in Task 1 at the same level of accuracy.
2. Poietic is faster than Bar in Task 1 at the same level of accuracy.
3. Poietic is as fast as EID in Task 1 at the same level of accuracy.
4. Poietic is faster than EID in Task 2 at the same level of accuracy.
5. Poietic is faster than Bar in Task 2 at the same level of accuracy.
6. Bar is faster than EID in Task 2 at the same level of accuracy.

The hypotheses were to be tested at the 5% significance level, meaning the p-value equals 0.05. All hypotheses with the exception of hypotheses 3 and 9 are directed and would as such be tested unilaterally. Hypotheses 3 and 9 were to be tested bilaterally.

7.2.2.5 Procedure

Due to test participant recruiting problems, two occasions of the experiment took place. Both occasions of the experiment were conducted in a similar way, and in the same room, in the same time of the day.

The participants were randomly assigned into one out of three groups. As mentioned above, one display design was assigned per group. The participants who belonged to the same group and thus tested the same display sat next to each other insofar as it was possible in order to prevent them from seeing the other displays.

In order for the participants to learn to identify the system states, there were training sessions where eight graphs were visible at the same time. The participants got immediate feedback on their choice of system state when they were in a training session. After identifying system states for all graphs, a five second delay followed in order to give the participants some time for reviewing the graphs and their answers. The training sessions were important for the experiment, as poietic design does not require immediate understanding of something, and the participants will learn to get an intuition of the graph only after some practice.

After a training session ended, only one graph was visible at a time, and the participants did not receive feedback on whether their choice of system state was correct or not. This session will be referred to as extended practice. A training session followed by extended practice will be referred to as a block.

The procedure went as following. Task 1: Block 1 (training session, extended practice), Block 2 (training session, extended practice), Task 2: Block 1 (training session, extended practice), Block 2 (training session, extended practice). In total, 384 * 2, altogether 768 graphs were displayed per participant. Repetition occurred, meaning all 768 graphs were not unique.

7.2.3 Results

In order to replicate the study by Holt et al. (2015), both latency and accuracy were tested. There is a trade-off between the two, meaning that test participants either sacrifice some accuracy for shorter latency, or they lengthen their latency for higher accuracy. However, since Holt et al. measured performance in that way, and both latency and accuracy are inarguably important qualities of performance, both were tested in the current experiment as well.

The median values for time to answer and percentage of correct answers were calculated for each test participant (Table 2). The results are visualised in Figure 17 together with the 25 and 75 percentiles. It should be noted that both blocks from the tasks showed similar results. Thus, only the data from the extended practice from the second blocks has been tested and visualised, as the test participants had had a longer learning time in those blocks.

Table 2. Data from the experiment. 'Time' means the median time to choose a state, latency. 'Right' means the percentage of correct answers, accuracy.

	Time, Block 1	Time, Block 2	Right, Block 1	Right, Block 2		Time, Block 1	Time, Block 2	Right, Block 1	Right, Block 2
Poietic, Task 1	1.7	1.4	99.0	100.0	Poietic, Task 2	1.8	1.8	53.1	76.8
	2.0	1.5	94.8	89.6		1.5	1.1	85.4	86.3
	1.0	0.9	80.2	82.3		1.1	0.9	86.5	85.3
	1.5	1.2	99.0	100.0		3.4	2.5	57.9	62.1
	1.6	1.4	94.8	100.0		1.8	1.4	94.8	92.6
	2.3	2.3	92.7	97.9		1.9	1.4	90.6	87.4
	1.5	1.2	95.8	96.9		0.9	0.7	26.0	29.5
	1.5	1.4	99.0	100.0		2.1	1.5	93.8	95.8
	1.7	1.4	94.8	95.8		1.7	1.7	43.8	65.3
Bar, Task 1	1.7	1.3	100.0	100.0	Bar, Task 2	2.7	2.2	82.3	83.2
	1.6	1.5	97.9	94.8		1.0	1.8	39.6	50.0
	2.8	1.8	80.2	84.2		2.3	2.2	69.8	72.6
	1.7	1.4	94.8	99.0		2.5	1.8	76.0	82.1
	3.0	2.0	97.9	100.0		2.8	2.5	81.3	81.1
	1.6	1.4	20.2	27.1		1.4	0.8	27.1	18.9
	1.9	1.6	97.9	99.0		1.4	0.9	75.0	74.7
	2.6	2.5	95.8	95.8		4.4	3.4	85.4	87.4
	4.3	2.2	76.8	95.8		3.8	2.9	79.2	83.0
EID, Task 1	1.6	1.5	99.0	100.0	EID, Task 2	7.2	3.1	65.3	75.8
	1.2	0.9	93.8	91.7		2.4	2.1	15.6	48.4
	1.1	0.9	96.9	93.8		2.9	5.1	18.8	20.2
	0.8	0.8	91.7	97.9		3.0	2.1	73.7	85.3
	1.0	1.0	97.9	100.0		2.7	1.8	26.0	31.6
	1.3	1.2	99.0	97.9		5.6	3.0	79.2	82.1
	1.2	1.0	96.9	96.9		6.1	4.5	83.3	84.0
	1.1	1.1	100.0	99.0		9.2	5.9	85.4	91.6

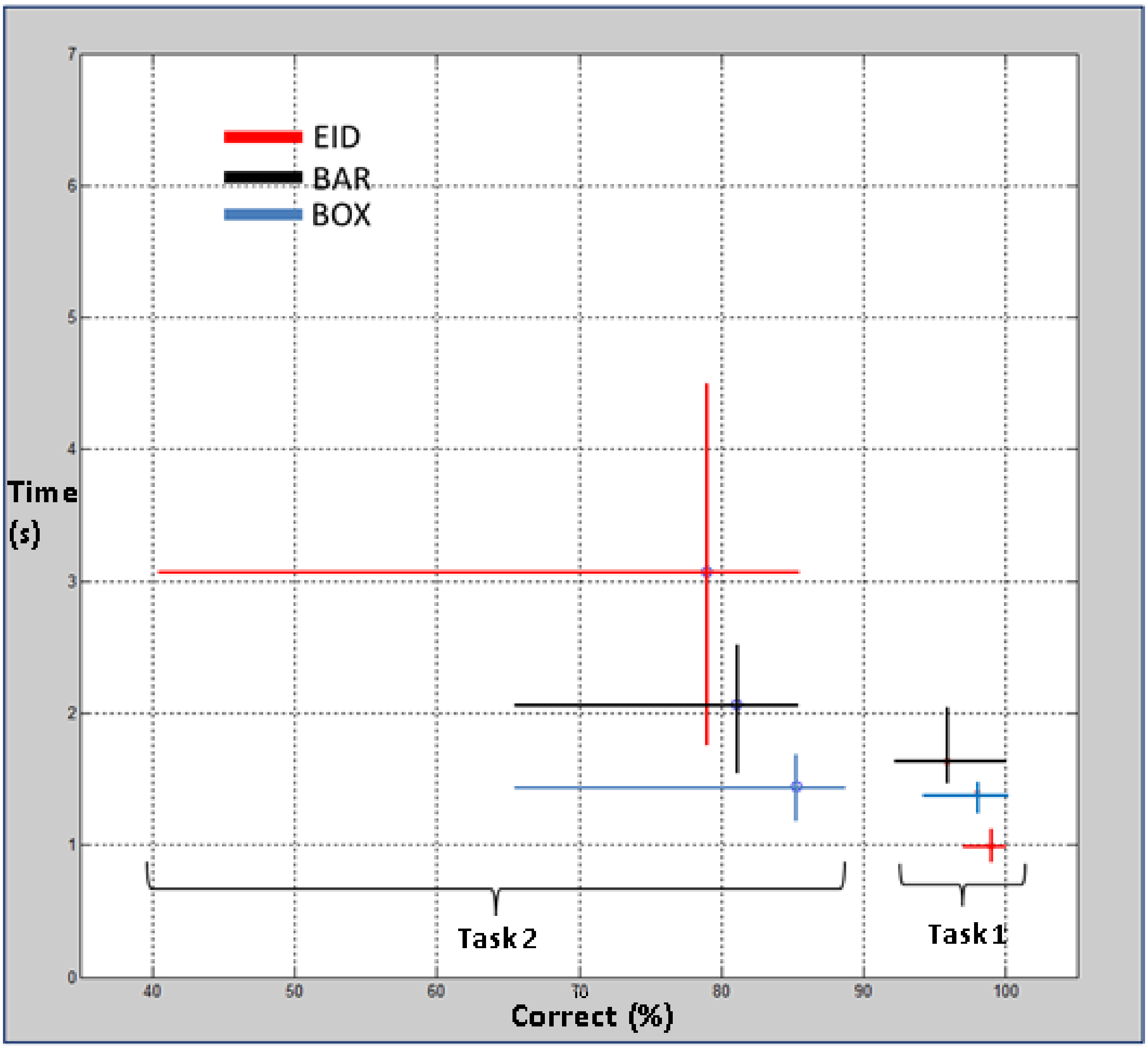


Figure 17. The results. The y axis depicts the latency and the x axis depicts the accuracy.

A Mann-Whitney U test was conducted in order to test the hypotheses listed in the Method section above. The Mann-Whitney U test is also called the Wilcoxon rank-sum test, and this method has been called the nonparametric equivalent to the 2-sample t-test method (Armitage, Berry, & Matthews, 2008; Frost, 2015; “Minitab 17 Support,” 2016), which is a parametric method. The difference between the methods is mainly that parametric tests assume normally distributed responses, while nonparametric tests do not (Armitage et al.; “Minitab 17 Support,”).

It was first examined whether there were any differences in accuracy between the conditions in each task by repeated paired tests. No significant differences were found with a decision criterion at p=0.05. This means that the hypotheses could be tested at the same level of accuracy.

Then the latency was investigated. The following hypotheses were tested using the same decision criterion:

1. EID is faster than Bar in Task 1.

   The null hypothesis that the conditions perform equally is rejected since the ranked sums 39 gives a p-value=0.0003, unilaterally. Hence, this hypothesis is supported. The median time for the EID display is 0.99 seconds and for the bar graph display 1.63 seconds, an increase by 65%.

2. Poietic is faster than Bar in Task 1.

   The null hypothesis that the conditions perform equally is rejected since the ranked sums 63 gives a p-value=0.025, unilaterally. Hence, this hypothesis is supported. The median time for the poietic display is 1.39 seconds and for the bar graph display 1.63 seconds, an increase by 17%.

3. Poietic is as fast as EID in Task 1.

   The null hypothesis that the conditions perform equally is rejected since the ranked sums 105 gives a p-value=0.0206, double-sided. Hence, this hypothesis is rejected. The median time for the EID display is 0.99 seconds and for the poietic display 1.39 seconds, an increase by 40%.

4. Poietic is faster than EID in Task 2.

   The null hypothesis that the conditions perform equally is rejected since the ranked sums 48 gives a p-value=0.00003, unilaterally. Hence, this hypothesis is supported. The median time for the poietic display is 1.44 seconds and for the EID display 3.08 seconds, an increase by 114%.

5. Poietic is faster than Bar in Task 2.

   The null hypothesis that the conditions perform equally is rejected since the ranked sums 66 gives a p-value=0.047, unilaterally. Hence, this hypothesis is supported. The median time for the poietic display is 1.44 seconds and for the bar graph display 2.07 seconds, an increase by 44%.

6. Bar is faster than EID in Task 2.

   The null hypothesis that the conditions perform equally is rejected since the ranked sums 62 gives a p-value=0.037, unilaterally. Hence, this hypothesis is supported. The median time for the bar graph display is 2.07 seconds and for the EID display 3.08 seconds, an increase by 49%.

7.2.4 Discussion

The EID design performed better than both the poietic design and the bar graph display in Task 1, but worse in Task 2. Hence, the results suggest that the EID design is indeed quite limited to the task that it was specifically designed for. Meanwhile, the poietic design performed better than the bar graph design in both tasks, indicating that poietic design can be useful in designing information displays and visualisations. The only hypothesis that was rejected regarding latency was the third one, that the poietic display would perform as well as the EID display in Task 1. It was rejected due to that the EID display performed better than the poietic display. This is perhaps not too strange, seeing as the EID display was designed to perform well in that task, while the poietic display was not designed for that particular task but rather designed to perform well in other tasks as well. Thus, it is not surprising that the EID display was difficult to trump in Task 1. While the results for hypothesis 3 is a slight disappointment for the poietic design, it still performed better than the neutral bar graph display, suggesting that poietic design is an effective and efficient way of visualising data. Besides, given more practice, the poietic design might be just as effective and efficient as the EID design even in Task 1.

It may be worth mentioning that hypothesis 3 and 9 were the only hypotheses to be tested double-sided. The reason for this is because they were less directed than the other hypotheses. As the other hypotheses were directed, they could be tested one-sided.

It may also be worth mentioning that even though the results were insignificant regarding accuracy, the trends were similar to the latency results, as seen in Figure 17. As mentioned above, there is a trade-off between latency and accuracy. This might explain why not both of the latency and the accuracy results were significant; if most test participants focussed on being quick rather than accurate, the results make sense. As seen in the instructions (Figures 11-16), the participants were encouraged to answer both as quickly and as accurately as possible, but one has to favour one or the other.

There was a reason for analysing the data with a nonparametric method instead of a parametric method such as a 2-sample t-test or an analysis of variance (ANOVA). Nonparametric tests are usually conducted when the responses are not normally distributed, the results are skewed (Armitage et al., 2008; "Minitab 17 Support," 2016). After a preliminary analysis of the test data, it was determined that this was the case of the current experiment. This is clearly visible by observing the asymmetrical values for the 25 and the 75 percentiles in Figure 17.

Unlike parametric tests that use mean values, nonparametric tests rely on median values. This is because the median is less sensitive to statistical outliers. While Holt et al. (2015) removed their outliers based on the latency scores, it was determined that one might as well conduct a nonparametric method instead. Limitations of nonparametric tests include that one is less likely to reject the null hypothesis when it is false compared to when using a parametric test, and nonparametric tests often require a modification of the hypotheses ("Minitab 17 Support," 2016).

Since an ANOVA was used by Holt et al. (2015), the closest nonparametric alternative would be a Kruskal-Wallis test (Frost, 2015; "Minitab 17 Support," 2016). However, as the groups' sample sizes were not identical, a Mann-Whitney U test was used instead. A limitation of the Mann-Whitney U test is that it is for comparing two independent groups (Armitage et al., 2008; Frost; "Minitab 17 Support,"). However, as we investigate targeted tests between two groups at a time, the Mann-Whitney U method worked well for the purposes of this experiment.

Even though the procedure of the experiment went smoothly, it did take a longer time than expected. Because the experiment took longer than anticipated, the results might have been affected. By the second task the test participants had got more time to learn the tasks, but many had probably started to get tired. Indeed, some participants showed signs of fatigue by the end of the procedure. This means that while many participants might have been faster in Task 2 compared to in Task 1, they probably made more errors. As such, a more truthful score of Task 2 might had shown a slightly better latency but a slightly worse accuracy. This could possibly explain the overall worse accuracy for Task 2 compared to Task 1 (Figure 17), but on the other hand it might just depend on that the objective of Task 2 was simple more difficult than for Task 1. Indeed, even if the accuracy had been better for Task 2, the latency would probably have been worse according to this theory. As the overall latency was worse for Task 2 already, an even greater difference between the tasks would have resulted.

Another unforeseen problem with the experiment was that a small part of the test scores did not become registered. This was probably due to a slight lag in the experiment software. The small loss was not considered to affect the overall results of the experiment, however.

To summarise, the results of the experiment support the following theories:

1. The EID design by Holt et al. (2015) is largely limited to the task that it was specifically designed for, and
2. The poietic design shows big potential for visualising data.

8 General discussion

In this thesis, much ground has been uncovered regarding poietic design. We have learnt that the design philosophy has many qualities that, while perhaps not unique on their own, creates a new way of looking at design. The attributes that define poietic design has been proposed and applied to examples. The final heuristics have been derived from those attributes. This answers the first research question: What attributes define poietic design?

The second research question reads as follows: What distinguishes poietic design from other design philosophies? In order to answer the second research question, a comparison has been made between poietic design and the concepts discussed in the Theory section. In short, the main things that distinguishes poietic design from related design philosophies are

1. A more holistic approach (having a technology-centred but also use- and user-centred approach),
2. A stronger focus on intuitive understanding while not hiding complexity, and
3. Designing for allowing autonomous decision-making while lessening the need for using System 2 thinking.

The third research question reads as follows: Is poietic design useful for designing user interfaces? The short answer is yes. A slightly longer answer would be that it has its uses. Poietic design might not be the ideal design philosophy to use for designing simple interfaces, systems for one-time use (where users only use it occasionally or just once or twice in their lives), or artefacts used leisurely (where the user does not need or want to have all their focus on it). In other cases, there is little doubt that designers can benefit from the design philosophy.

8.1 Ethical considerations

As described in the Poietic design section, poietic design lets the user decide for themselves whether the artefact's effects on the world or the user's decisions are good or bad. As such, the poietically designed artefact does not judge the effects, nor preach to the user what they should do. This is a very important point for poietic design. If we want to design for autonomy, we must let the users make decisions and let them decide for themselves whether their actions are good or bad.

One might argue that too much autonomy is bad because it is a slow and expensive way of thinking (see Decision-making above). However as previously mentioned, poietic design strives to reduce the need for System 2 thinking and thus it should make up for the cost of autonomous thinking. Furthermore, advantages such as heightened moral responsibility and control of decisions outweigh the disadvantages in at least some cases.

8.2 Complications

There have been some limitations in this thesis. Firstly, as poietic design is a largely novel concept, the theoretical background research has been wide and exploratory. As such, some threads might have been left loose, while others that might have been relevant have been missed. Secondly, the experiment that was conducted had a bit too few test participants than wished for due to lack of time and resources to find willing subjects. While the amount of participants was sufficient for a master thesis, one might want to redo the experiment with more participants for even more accurate results. Furthermore, with more participants it would be easier to account for outliers in the analysis and to use a parametric method instead of a nonparametric method, which might be considered an advantage as parametric methods are usually seen as slightly more powerful ("Minitab 17 Support," 2016). More identified issues of the experiment were that it took more time than expected to complete and that a few test scores did not get registered. Thirdly, due to the nature of the experiment the poietic redesign did not possess all the identified attributes of poietic design. As such, not all heuristics got to be tested. Since most of the hypotheses did not have to be rejected, all of the proposed heuristics seem to be working fine, however. Because of this, it was deemed that no updates of them are necessary for now.

8.3 Future work

As this is the very first study of poietic design, it is clear that more research is needed. More poietic designs need to be tested and evaluated in the future. Seeing as poietic design is less concerned with exact numbers, it is possible that it would be especially beneficial with a focus on qualitative research in the future. A longitudinal experiment where the heuristics of poietic design are evaluated would also be of great interest. It would also be interesting to investigate the connection between poietic design and emotions, and between poietic design and inductive reasoning. Perhaps new heuristics or new ethical considerations would emerge.

9 List of references

Armitage, P., Berry, G., & Matthews, J. N. S. (2008). *Statistical Methods in Medical Research (4)* (4th ed.). Chichester, GB: Wiley-Blackwell.

Bakker, S., van den Hoven, E., Eggen, B., & Overbeeke, K. (2012). Exploring Peripheral Interaction Design for Primary School Teachers. *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction - TEI '12*, *1*(212), 245–252. http://doi.org/10.1145/2148131.2148184

Bennett, K. B., & Flach, J. M. (2011). *Display and Interface Design: Subtle Science, Exact Art*. *Display and Interface Design*. New York, NY: CRC Press.

Borst, C., Flach, J. M., & Ellerbroek, J. (2015). Beyond Ecological Interface Design: Lessons From Concerns and Misconceptions. *Human-Machine Systems, IEEE Transactions on*, *45*(2), 164–175. http://doi.org/10.1109/THMS.2014.2364984

Brown, J. N. A. (2012). Expert Talk for Time Machine Session: Designing Calm Technology "… as Refreshing as Taking a Walk in the Woods," *4*(1), 423. http://doi.org/10.1109/ICME.2012.24

Bulk Rename Utility. (n.d.). Retrieved April 1, 2016, from http://www.bulkrenameutility.co.uk/Main_Intro.php

Carlsson, A.-L. (2010). The Aesthetic and the Poietic Elements of Information Design. *2010 14th International Conference Information Visualisation*, (2004), 450–454. http://doi.org/10.1109/IV.2010.69

Cooper, A., Reimann, R., & Cronin, D. (2007). *About face 3: The essentials of interaction design* (3rd ed.). Indianapolis, Indiana: Wiley Publishing, Inc.

Coury, B. G., Boulette, M. D., & Smith, R. A. (1989). Effect of uncertainty and diagnosticity on classification of multidimensional data with integral and separable displays of system status. *Human Factors*, *31*(5), 551–569. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/2625349

Dix, A. (2009). Human-Computer Interaction. In *Encyclopedia of Database Systems* (pp. 1327–1331). Springer.

Download.com. (n.d.). Retrieved April 1, 2016, from http://download.cnet.com/Bulk-Rename-Utility/3000-2248_4-10174242.html

Eldridge, A. (2012). The Poietic Power of Generative Sound Art for an Information Society. *Proceedings of the International Computer Music Conference*, *2*(7), 67–81.

Epstein, R. M. (2013). Whole mind and shared mind in clinical decision-making. *Patient Education and Counseling*, *90*(2), 200–206. http://doi.org/10.1016/j.pec.2012.06.035

Flach, J. M., Tanabe, F., Monta, K., Vicente, K. J., & Rasmussen, J. (1998). An ecological approach to interface design. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *42*(3), 295–299. Retrieved from http://pro.sagepub.com/content/42/3/295.short

Friedewald, M., & Raabe, O. (2011). Ubiquitous computing: An overview of technology impacts. *Telematics and Informatics*, *28*(2), 55–65. http://doi.org/10.1016/j.tele.2010.09.001

Frost, J. (2015). Choosing Between a Nonparametric Test and a Parametric Test. Retrieved May 17, 2016, from http://blog.minitab.com/blog/adventures-in-statistics/choosing-between-a-nonparametric-test-and-a-parametric-test

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. New York, NY: Psychology Press.

Gustafsson, A. (2010). *Positive Persuasion: Designing enjoyable energy feedback experiences in the home*. *University of Gothenburg*. University of Gothenburg.

Heidegger, M. (1977). The question concerning technology. *Technology and Values: Essential Readings*, 3–35.

Holt, J., Bennett, K. B., & Flach, J. M. (2015). Emergent features and perceptual objects: re-examining fundamental principles in analogical display design. *Ergonomics*, *58*(12), 1960–1973. http://doi.org/10.1080/00140139.2015.1049217

Hudson, S. E., Harrison, C., Harrison, B. L., & LaMarca, A. (2010). Whack gestures: inexact and inattentive interaction with mobile devices, 109–112. http://doi.org/10.1145/1709886.1709906

Janlert, L.-E., & Stolterman, E. (2010). Complex interaction. *ACM Transactions on Computer-Human Interaction*, *17*(2), 1–32. http://doi.org/10.1145/1746259.1746262

Jeppsson, C. (2015). Veckans Affärer. Retrieved from http://www.va.se/nyheter/2015/11/11/ny-produkt-the-pac/

Kagitcibasi, C. (2005). Autonomy and relatedness in cultural context: Implications for self and family. *Journal of Cross-Cultural Psychology*, *36*, 403–422. http://doi.org/10.1177/0022022105275959

Kaptelinin, V. (2014). Affordances. In *The Encyclopedia of Human-Computer Interaction* (2nd ed., p. 4000+). Aarhus, Denmark: Interaction Design Foundation. Retrieved from https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/affordances

Klein, G. (2008). Naturalistic Decision Making. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *50*(3), 456–460. http://doi.org/10.1518/001872008X288385.

Laaksoharju, M. (2014). *Designing for Autonomy*. *Uppsala University*. Uppsala University.

Minitab 17 Support. (2016). Retrieved May 17, 2016, from http://support.minitab.com/en-us/minitab/17/

Norman, D. A. (1981). The trouble with UNIX. *Datamation*, *IV*(I), 37–41.

Norman, D. A. (2004). Affordances and Design. Retrieved from http://www.liacs.nl/~fverbeek/courses/hci/AffordancesandDesign.pdf

Norman, D. A. (2009). *The design of future things*. New York: Basic Books.

Norman, D. A. (2010). *Living with complexity*. The MIT Press.

Norman, D. A. (2013). *The design of everyday things: Revised and expanded edition*. *Basic Books*.

Patmalnieks, A. (2016, February). Spel räcker inte för att skapa miljövänligt beteende. *KTH Magazine*. Retrieved from https://www.kthmagazine.se/artiklar/spel-racker-inte-att-skapa-miljovanligt-beteende

Rasmussen, J. (1983). Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models. *IEEE Transactions on Systems, Man, and Cybernetics*, *13*(3), 257–266.

Sunstein, C. R. (2005). Moral heuristics. *Behavioral and Brain Sciences*, *28*(4), 531–573. http://doi.org/10.1017/S0140525X05000099

Taylor, D. (n.d.). Affordances and Design. Retrieved March 2, 2016, from https://www.interaction-design.org/literature/article/affordances-and-design

The Pac. (n.d.). Retrieved March 30, 2016, from http://www.poweraware.com/en/

Todd, P. M., & Gigerenzer, G. (2000). Précis of Simple heuristics that make us smart. *The Behavioral and Brain Sciences*, *23*(5), 727–741; discussion 742–780. http://doi.org/10.1017/S0140525X00003447

Weiser, M. (1991). The Computer for the 21st Century. *Scientific American*. http://doi.org/10.1038/scientificamerican0991-94

Weiser, M., & Brown, J. S. (1996). Designing Calm Technology. *PowerGrid Journal*, *1*(1), 75–85.

Weiser, M., & Brown, J. S. (1997). The Coming Age of Calm Technology [1]. *Beyond Calculation*, 75–85. Retrieved from http://homes.di.unimi.it/~boccignone/GiuseppeBoccignone_webpage/IUM2_files/weiser-calm.pdf

Vicente, K. J. (1999). *Cognitive work analysis: Toward safe, productive, and healthy computer-based work*. Mahwah, New Jersey: Lawrence Erlbaum Associates, Inc.