INTERNATIONAL CONFERENCE ON ENGINEERING AND PRODUCT DESIGN EDUCATION
5 & 6 SEPTEMBER 2013, DUBLIN INSTITUTE OF TECHNOLOGY, DUBLIN, IRELAND

BEYOND STATIC: IDEATION USING INTERACTIVE PROTOTYPING TOOLKITS

Dries DE ROECK, Bart STOFFELS, Achiel STANDAERT and Stijn VERWULGEN
Artesis University College of Antwerp, Department of Design Sciences

ABSTRACT
This paper will focus on how digital aspects of design can be included in the very first phases of a design process. The research presented is based on a mapping of existing interactive prototyping tools, where both the 'physical design' and the 'digital design' side are taken into account. The physical design mapping focuses on the creation of an interactive prototype involving sensors and actuators. The digital design aspect focuses on how computational code is fed into these prototypes via a variety of programming environments. A central argument in the presented research is that existing tools and toolkits that allow for the creation of digital interactive prototypes are mostly oriented towards verification. This means that a toolkit is used after an idea or concept has been crystallised. For an industrial designer, the aim, however, should be at using interactive toolkits during the ideation phase. Therefore it is crucial to look into how these existing tools can be used hand in hand with renowned ideation tools such as storyboarding and wizard-of-oz techniques, in order to enhance the quality of the outcome. The feasibility and effective value of this methodology was explored and preliminary evaluated during a one week exercise with both Bachelor and Master industrial design students.

Keywords: Digital creation, interactive prototyping, toolkits, interaction design

1 INTRODUCTION

One of the results of our rapidly evolving socio-technological context is that physical products are increasingly becoming 'connected' to digital networks. Previously, this network was mostly used to connect people together via a phone call or a social network. This specific interpretation of a network is characterised by the fact that it is invisible, there is no physical interaction with this network. More recently, this invisible network has expanded towards the physical realm. The implication of this evolution is that physical products become increasingly digitally interactive and thereby constructing an 'internet of things' [9] in a 'connected world'.

These digitally connected products, where streams of digital data have an impact on the physical representation of the product, are referred to as 'meta-products' [7]. In 2012 alone, the popular online crowdfunding platform Kickstarter has funded over twenty successful projects involving 'internet of things' applications and products. The corresponding funding is more than 4.5 million dollars [6]. This observation confirms the market interest in the potential of the 'internet of things' [4]. However a careful analysis of the product characteristics incorporating digital interaction reveals important discrepancies between the product as intended and its actual context of use. It is apparent that most of these products are created with a technically dominant mindset and would benefit a design driven-approach. An example of this is the 'smart fridge', a product concept that has been surfacing every so often for the some twenty years. The basic idea is that a fridge keeps track of its contents and is able to automatically propose recipes or order

2 THE INTERNET OF THINGS, ANNO 2013

The internet of things originated from a very technical community about a decade ago. Initially, it was mostly linked to objects being digitally tagged using RFID tags providing 'hidden' information about a certain product [1]. Over time, the application field of the internet of things has kept expanding. More recently, design communities have picked up the term and have embarked on a totally different, yet complementary, approach to this domain. One of the reasons that the design community has become involved is because the tools required to create internet of things related applications are increasingly becoming accessible for people without a profound technical background. A widely

known example of this is the Arduino microcontroller platform. The Arduino enables people to create digitally connected, physical, objects using sensors and actuators. It has become very popular as a tool in interaction design, and is taught at several industrial design schools across the world. Recent developments are even stretching it further, bringing digital creation tools to a larger public. A well known example of such an initiative is the Scratch environment (MIT), which is specifically built for children, but shows that it is possible to engage a wider audience in creating 'digitally connected' things.
It is important to highlight that the current interpretation of the internet of things has evolved and is moving away from a heavy focus on technology. Technology creators have showcased the possibility of connecting objects via a variety of networks, but have in most cases undervalued the importance of the context of use or providing a meaningful way for people to interact with the created technologies. Shove, Watson et al [8] compare this evolution to what happened to interior decoration of houses in the past. It used to be almost impossible to paint a door if you were not a professional, but as home decoration tools became accessible to a larger audience, people could decorate their houses to their own taste and preferences.

3 DESIGNER COMPUTER CONFLUENCE

Although digital creation tools are gaining momentum, they do introduce some issues that are sometimes overlooked. Bannon [2] points to a very crucial point, which can be summarised that if we stick to 'known' technologies and 'known' tools to create new products and services, we often result in 'known' concepts and ideas. When we talk about the internet of things, the known technologies are mostly based on sensors and actuators which are linked to each other through a processing unit (such as a microcontroller). Current digital creation toolkits are constructed from within this logic. The risk when using these tools as a designer, who does not necessarily have a profound understanding of computational structures, is that the things created using these tools are limited by the understanding of interpretation of the tool. For example, a designer wants to create a product with fading status LEDs in order for the product to express a more 'gentle' feel. Although the concept of a fading LED might seem easy, it requires some specific knowledge about electronical systems (eg pulse width modulation, FOR loops and the declaration of variables) to prototype this.
A very different approach to this problem is to introduce abstracted technology to design with. Related work such as Magical Bits [5] or Lilidots [3] enables the exploration of digital products on a higher level. The clear advantage of this approach is that several technological constraints are removed, which allows for more creativity. The downside, however, is that the ideas resulting from using this type of methods are harder to verify if they are actually feasible to create.
Based on these insights, it becomes clear that a challenge we are facing is to introduce digital creation to designers, without impacting the aspect of creativity. Additionally, it is becoming clear that commonly used design tools such as storyboarding are reaching their limits when digitally interactive design is being merged with the design of physical objects. In order to explore possible solution spaces to this problem, we created an overview of available tools and introduced a selection of these tools to a group of industrial design students.

4 TOOLS AND TOOLKITS MAPPING

In order to gain a better understanding of the characteristics of digital creation toolkits, a mapping was made. The selection of interactive prototyping toolkits is based on a selection made by Knörig in "Design Tools Design" [10]. In his master thesis, he selects useful toolkits for product designers and places them in different phases of the design process. However, this selection focuses on finding a toolkit to cover all the phases in the design process. In order to show a more diverse selection, extra toolkits are selected from renowned online shops such as SeeedStudio, Sparkfun and AdaFruit. Despite the fact that the selection of toolkits is not complete, it tries to cover the whole spectrum of available toolkits and give designers a guide to place their own interactive prototyping toolkits within the map.
The goal of this mapping was to identify one or more dimensions, which could lead to a selection of toolkits to be used in a further exploration. Overall, digital design toolkits can be split into two categories; physical and digital. These categories are evenly important to take into account, but mapping them on the same scale would create a chaotic presentation of the different toolkits. Therefore Figure shows a mapping of toolkits on a physical vs. digital axis. Within each axis they are

mapped according the visualization of data- and information-flows through the toolkits. Depending on how visual functions and adjustments are shown in the prototypes or software programs, it will be easier for the designer to understand the structure of the prototype. The physical tools are ranked on a scale from black box toolkits to toolkits that visually display the progress of information and data. To compare the digital components of the toolkits there is a small shift in the use of comparison parameters. The digital elements of the toolkits are placed on a scale from abstract programming code to, again, visually displayed progress of the processed data.

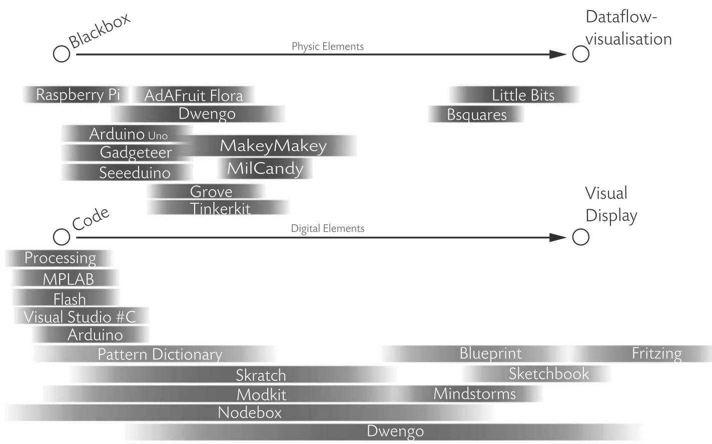

Figure 1. Overview of digital creation tools

5 EXPLORATION IN PRACTISE

When evaluating the overview of digital creation toolkits, it is clear that there visual programming tools are less common compared to code based environments. Since the coding and the understanding of computational logic is typically something designers struggle with, offering a more visual way of creating and programming holds a lot of potential. This point of view was explored before within the context of user interface design by Hartmann et al. [11], and showed that is possible to include digitally interactive hardware during an ideation phase. Within the context of the research presented in this paper, we wanted to gain insight if a similar integration of digital components in a hardware product could be achieved and what the added value of doing so could mean for the (industrial) designer.

5.1 Approach and setup

We selected three different toolkits beforehand, which were selected based on their capabilities to be programmed in a visual way. An additional requirement was that the chosen toolkits did not require a deep understanding of electric circuit design. The used sensors, actuators and other components needed to be ‘plug and play’ in order to keep the focus on the actual creation and ideation. Based on these criteria, following toolkits and programming were selected:

- *Scratch & MakeyMakey*
  Scratch is a visual programming language that can be constructed using a linear stacking of building blocks. The makeymakey platform is an Arduino based system that is capable of sending keyboard characters to the computer by closing a circuit between specific conductive zones. When combining scratch with makeymakey, it is very simple to have a computer system reacting to events. Makeymakey is mostly limited to screen based output, this means that events in the real world can trigger things that are displayed on a computer screen.
- *Grove shield & Scratch for Arduino*
  Grove is a set of pre-built sensors and actuators, that can be plugged into an Arduino compatible extension board (shield). Using scratch for Arduino, these sensors can be easily imported in the scratch environment mentioned before. The difference with the makeymakey platform is that

hardware sensors can be added to the prototype.

- *Dwengo & Dwengo blocks*
  Dwengo is a microcontroller platform that has a lot a functionality on-board such as built in LEDs and a built in LCD display. It can be programmed using Dwengo Blocks, which is, compared to Scratch, a more organic and less linear way of programming in a visual way.
- *Milcandy*
  Milcandy is a toolkit that does not require any software programming. All programming is done on the hardware device. Milcandy is limited to creating 'if this … then that' statements.

5.2 Process

Over the course of five days, a group of seventeen students used the before mentioned toolkits to create functional prototypes. During the first day they were introduced to the toolkits and the visual programming environments. The last day was used to create a small exposition of the work done. This left three days of actual work.
At the start of each workshop day, the whole group was introduced to a predefined scenario, consisting of a context description and a persona. In total three context were explored, (i) a senior person in need of assistive living systems (ii) a young professional with an interest in cooking and (iii) a child which needed to be motivated to live a more active life.
In order to compare the functionality and potential of the toolkits, the group was split up in two smaller groups. Each group was handed a different toolkit, which they were to work with for half a day. After that time, the groups switched toolkits and worked for the rest of the day using the other toolkit within the same case.

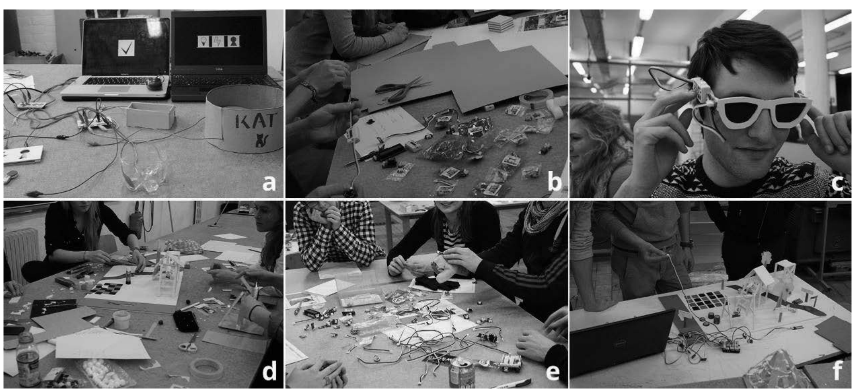

Figure 2. Workshop progress: using three different digital creation toolkits in a creative exploration of three different scenario contexts

5.3 Preliminary results

At the moment of writing a complete qualitative analysis has not been completed. Based on the observations and questionnaires there are however several important insights that have surfaced;

5.3.1 If … then

Most prototypes were reactive systems to certain events. A typical example of this can be seen on figure 2c: a pair of glasses that switches on a built in light when it gets dark. There were very few complex interactions created. This implies that the participating students managed to understand the basic structure or a computational input-output system, but were lacking more advanced reasoning to create more complex programs.

5.3.2 Verification and result oriented

During each half day session, most groups would first discuss a general idea orally. After they came to a consensus of a certain idea, this idea would be prototyped and built for the remainder of the time. Whereas we had expected the teams to first try smaller systems out, they worked very result oriented instead of experimentally.

5.3.3 Imagination friendly

We deliberately chose to have some toolkits that used hardware sensors (Grove, Milcandy) and others that did not (MakeyMakey, Dwengo). When using the hardware sensors, it was apparent that the ideas generated were a direct result of the 'default' interpretation of the provided sensors. For example, a water sensor was only used to detect water. There was no creative interpretation of how these sensors could be used differently for example. The toolkits that were not bound to sensors allowed for a much more creative exploration, which resulted in ideas such as a cat bowl food detector (figure 2a)

6 DISCUSSION & CONCLUSION

It is becoming increasingly difficult to neglect confluence of digital systems and industrial design activities. This ongoing socio-technological shift shows that objects and digital systems are becoming connected, which opens a lot of opportunities for industrial design. Recent technical evolutions have allowed designers to design and create digitally functional products and systems more easily. Although this is an interesting change, most existing digital creation toolkits used by designers are very result oriented and do not stimulate creative exploration. We explored the use of four digital creation toolkits during a weeklong workshop in the early phases of the design process. During this workshop it became clear that the used toolkits are ideal for communicating a basic idea to a project team, but do not necessarily stimulate creative exploration. The continuation of this research will mostly consist of developing a method that provides a balance between generating ideas in an imaginative 'digital' world and generating functional concepts incorporating technological constraints.

ACKNOWLEDGMENTS

The described workshop could not have happened without the support of Jan Van Goey, Vincent Nulens, Kristof Custers and Carlo Van Hove of the Artesis University College of Antwerp.

REFERENCES

[1] Ashton K., That 'Internet of Things' Thing., 2009, in RFID Journal, 22 July 2009. http://www.rfidjournal.com/article/view/4986 (Retrieved 22 February 2013)

[2] Bannon L., Forgetting as a feature, not a bug: the duality of memory and implications for ubiquitous computing. *CoDesign*, 2006, 2(1), 3-15.

[3] De Roeck D., Slegers K., Stappers P.J. and Standaert A. Infusing digital creativity in design, the low fidelity way. In *Proceedings of CHI Sparks 2011*, Arnhem, June 2011.

[4] Fenn J., LeHong H., Hype cycle for Emerging Technologies, 2011 (Gartner Research, Stamford)

[5] Grufberg K., Holmquist L.E., Magical Bits: Designing Through Experiencing the Future End Product in *Proceedings of DESIRE '11,* Eindhoven, October 2011

[6] Harwood T., Charalampos D., Back A., Internet of Things and Kickstarter, http://postscapes.com/internet-of-things-and-kickstarter (Retrieved 22 February 2013)

[7] Hazenberg, W., Rubino, S.C., Huisman, M., 2012. Meta products: Building the internet of things, BIS Publishers, Amsterdam

[8] Shove E., Watson M., Hand M., Ingram J., 2008, The Design of Everyday Life, Bloomsbury Academic

[9] van Kranenburg, R., 2008. The Internet of Things: A critique of ambient technology and the all-seeing network of RFID. Amsterdam, Institure of Network Cultures.

[10] Knörig, A. 2008. Design Tools Design, M. Sc thesis, FH Potsdam.

[11] B. Hartmann, S. R. Klemmer, M. Bernstein, and N. Mehta, 2005, d.tools: Visually prototyping physical UIs through statecharts in Extended Abstracts of UIST'2005.