Designing Design Exercises – From Theory to Creativity and Real-world Use

Paula Kotzé and Peter Purgathofer
[1] Meraka Institute, Centre for Scientific and Industrial Research,
and School of Computing, University of South Africa,
P O Box 395, Pretoria, 0001, South Africa
Paula.Kotze@meraka.org.za
WWW home page: http://www.meraka.org.za
[2] Design and Assessment of Technologies Institute,
University of Technology Vienna,
Favoritenstraße 9-11, A1040 Vienna, Austria
purg@igw.tuwien.ac.at
WWW home page: http://igw.tuwien.ac.at/

Abstract. This paper discusses a framework for design exercises for interaction design and HCI based on two theoretical frameworks and a set of knowledge transformers. The model scope design exercises on a continuum ranging from creativity to real-world use based on the argument that students must experience design to enable them to learn effectively.

1 Introduction

Interaction design is very much a design discipline, albeit with a different approach than that of scientific and engineering disciplines, schools where it often resides. There is still no agreement (though change is being noticed) in the academic community about what the core elements of an interaction design curriculum might be, or how to approach the teaching of that curriculum [1]. The arts disciplines lean towards advancing interaction design as a creative art, focusing on sensorial elements and a means of personal or brand expression, rather than as an approach to solving product definition and usability issues. On the other end, the more technically inclined disciplines, such as computing, lean towards teaching interaction design from the point of view of exploring and implementing technologies rather than discovering and concentrating on human goals and needs. If one studies current curricula focussing on human-computer interaction (HCI) issues is clear that most focuses on user research and cognitive issues, with less emphasis on the crafts of design (methods and practices). Then again many design programmes in the arts still focus on design tools rather than methods and actual use of these methods in real-world situations.

This paper proposes a framework for design exercises for use in the teaching of interaction design aiming at positioning them with regards to both creativity and real-world use. The idea for the framework was conceived during the CONVIVIO Faculty Forum in Austria during December 2006 [2, 3].

The aim of section 2 is to clarify the domain of interaction design, while section 3 highlights the elements or tools of interaction design. Section 4 proposes CTUDE, the framework for design exercises. Section 5 positions design examples presented at the CONVIVIO Faculty Forum within CTUDE, while section 6 concludes.

2 What is Interaction Design

Interaction design is a young field and approach to designing interactive experiences [4]. It is a design discipline aimed at defining the behaviour of artefacts, environments, and systems [1]. According to Preece et al. [5: p.6] interaction design is about 'designing interactive products to support people in their everyday working lives' and interactive experiences that enhance and extend the way people communicate, interact and work. The interactive experiences can be in any medium (such as live events or performances, products, services, etc.) and not necessarily digital media. It involves interactivity. Interactivity is concerned with being part of the action of a system or performance and not merely watching the action passively, which makes it different from animation [4].

As a discipline, interaction design borrows theory and techniques from traditional design disciplines and practices (arts, industrial, communication, etc.), psychology and other social sciences, more scientific and technical disciplines (computing, information sciences, engineering), and also from various interdisciplinary fields. It does, however, not merely borrow, but also synthesise (it is more than a sum of its parts) to create its own unique methods and practices [1, 4, 5].

Interaction designers are involved in all the *interactive aspects* of a product, not only the graphics design of an interface but also with building interactive versions of the designs so that they can be communicated and assessed. They are involved with [1, 5]:

- Defining form: the form of products as they relate to their behaviour and use.
- Anticipating use: how the use of products will mediate human relationships and affect human understanding.
- Exploring dialogue: the dialogue between products, people, and contexts (physical, cultural, historical).

Interaction design is also a perspective that approaches the design of products in several *different ways* [1, 5, 6]:

- Identifying user needs, goals and requirements: forming an understanding of how and why people desire to use products and enable products that are self-aware in terms of behaviour or usage.

- Creating user experiences: aesthetically pleasing, supportive of creativity, rewarding, emotionally fulfilling, fun, satisfying, enjoyable, entertaining, helpful, motivating, etc.
- As gestalts (i.e. more than the sum of its parts), not simply as sets of features and attributes.
- Making sure affordances and seams are clear and malleable.
- By being creative: looking to the future, seeing things as they might be, not necessarily as they are now.

Interaction designers must therefore have a variety of *abilities and skills*. They must be able to [1, 5]:

- Understand the strengths and limitations of both humans (users) and technology.
- Learn new domains quickly.
- Solve problems both analytically and creatively.
- Be able to visualize and simplify complex systems.
- Empathize with users, their needs, and their aspirations, and their experiences.
- Share a passion for making the world a better place through ethical, purposeful, pragmatic, and elegant design solutions.

If we aim to have a successful training programme for interaction designers or HCI experts, we have to make sure our students are equipped with all the necessary skills, also on a practical level. This paper focuses on the development of design exercises that will serve this purpose: exercises on a continuum ranging from pure creative work to real-world applications.

3 The Elements of Interaction Design

Most design disciplines use raw materials. Industrial design, for example, uses simple 3-dimensional shapes such as cubes, spheres, cylinders, etc., while communication design uses basic visual elements such as the line. Interaction designers create products and services that can be digital (e.g. electronic software), analogue (e.g. a karaoke machine), or both (e.g. a mobile phone) [7]. Their design elements are more conceptual, yet powerful. Other design disciplines, however, offer a set of *components* that can be borrowed by interaction designers to support the design of their projects: motion, space, time, appearance, texture, sound [7] and context [8]:

- Motion: Interaction is a sort of communication, and communication is about movement: our vocal cords vibrating as we speak our hands and arms writing or typing as we send e-mail or short messages, sound and data moving between two entities. Interaction designers are concerned with behaviour: the way that products behave in response to the way that people behave. All behaviour can be seen as motion: motion coloured by attitude, culture, personality, and context. [7].
- Space: Space provides a context for motion. Movement happens in some kind of space, even if the boundary of that space is unclear (for example the Internet). Interaction designers work in both 2D and 3D space, whether that space is a

digital screen or the analogue, or the physical space we live in. Interaction design usually involves a combination of physical, analogue and digital space [7].

- Time: All interactions take place over time. Sometimes that time can be near instantaneous, like the time it takes to click a mouse, or it can involve very long durations. Time also creates rhythm and interaction designers can control this rhythm. Interaction designers therefore need an awareness of time [7].
- Appearance: How something looks gives us cues as to how it behaves and how we should interact with it. Appearance is the major source (texture is the other) of what cognitive psychologist James Gibson [9] called affordances, but it was Don Norman's book [10] that spread the term to interaction design and design in general. Affordance is a property, or multiple properties, of an object that provides some indication of how to interact with that object or with a feature on that object. Except to the visually impaired (for who texture often substitutes), appearance also conveys emotional content. Appearance has many variables for designers to alter, for example, proportion, structure, size, shape, weight, colour (hue, value, and saturation), etc. All of these attributes come together to form appearance, and nearly every design has some sort of appearance [7].
- Texture: How an object feels in the hand, i.e. the sensation of an object can provide clues as to how it is to be used as well as when and where. Texture can also convey emotion. Designers can also work with texture variables such as vibration and heat to signify actions [7].
- Sound: Sounds possess many variables that can convey information as well. Sounds are made up of three main components, all of which can be adjusted by a designer: pitch, volume, and timbre or tone quality [7].
- Context: Interaction design is about dialogue with the environment in which it is used. Context is more of a reference element, while the ones listed above are tangible. Designing context, or for context, most likely will speak to reflective emotional responses. Some of the sub-elements of context are: physical environment of use, personas of stakeholders, and culture of use [8].

These components are the interaction designer's toolkit and while interaction designers may not consciously manipulate them, they are the building blocks of interaction design. These elements are mixed and changed and skewed toward learning to master them as ingredients in a designer's kitchen. They should therefore form part of our teaching of interaction design, but they rarely are.

4 Framework for Design Exercises

Various academic institutions with new or established interaction design and HCI programmes are, however, starting to develop an understanding of interaction design and the qualities and skills required of interaction designers [1].

However, teaching students how to develop interactive experiences is not an easy task. There are a myriad of aspects involved that should be considered. Think for a moment how tricky it is to construct a meaningful experience for others. You must first understand your audience, their needs, abilities, interests, and expectations, and how to connect with them. Empathy with users and the ability to conceptualize effective solutions (and then refining them mercilessly) are difficult skills to teach. The same apply to teaching students to be creative and allowing them to experience such skills in giving them the opportunity to ‘experience design’.

In the past much teaching attention was focussed on usability, but although usability research is extremely important, it isn't design. Usability identifies problems, but usually doesn't suggest solutions, although the principles of usability can be used to guide design.

Designers all need some basic skills. Interaction designers should be able to draw (sketch) and write well, be at ease with the elements of interaction design (described in section 3), and must be able to communicate excellently with their colleagues and their clients, as well as the users of their designs through their designs.

Although interaction designers seldom code [1] they need to be sensitive to the limitations of the technology they design for. In order to create products, interaction designers will have to communicate coherently with software developers (non-designers). This means understanding code (not necessarily doing the coding themselves) [6].

Increasingly there is also a role for designers to craft systems which in turn enables design, contributing tools, not solutions [6, 11]. There is therefore evidence of designers and developers producing frameworks for others to create with.

The toughest skill to acquire is the combination of creative insight and analytical thinking.

As lecturer, where do you start, how do you define your syllabus and the exercises to equip your students with the practical skills to be successful interaction designers? In this paper we are not going to explicitly focus on the syllabus (although we identify possible shortcomings along the way), but the syllabus should address all the elements of design mentioned in section 3 and all the abilities and skill mentioned in section 2. This paper instead focuses a framework that educators can use to guide them in designing design exercises to enable students to experience design. Since it is generic, it can be adapted to fit design exercises for any syllabus. The issues of what makes a good or bad design exercise, and guidelines to follow in setting up good design exercises are addressed in another paper [2] and will not be repeated here.

4.1 Background

The proposed framework for design exercises is roughly based on two theoretical frameworks: the graphic thinking for architects and designers framework proposed by Laseau [12] and the framework for teaching and learning design proposed by Kotzé et al. [13].

4.1.1 Laseau’s Graphic Thinking for Architects and Designers Framework

The first part of our theoretical foundation is based in design theory and practice. Laseau [12] captures a number of interesting aspects of the design process in a simple model, as illustrated in Fig. 1. The model consists of two funnels: an expanding funnel (elaboration) and a contracting funnel (reduction). In this model he balances permanent creativity and idea generation, on the one side, with the reduction resulting from decision-making as main forces in design, on the other side. These two ingredients of design benefit from quite different approaches and methods that can be characterized as sketching and prototyping, respectively.

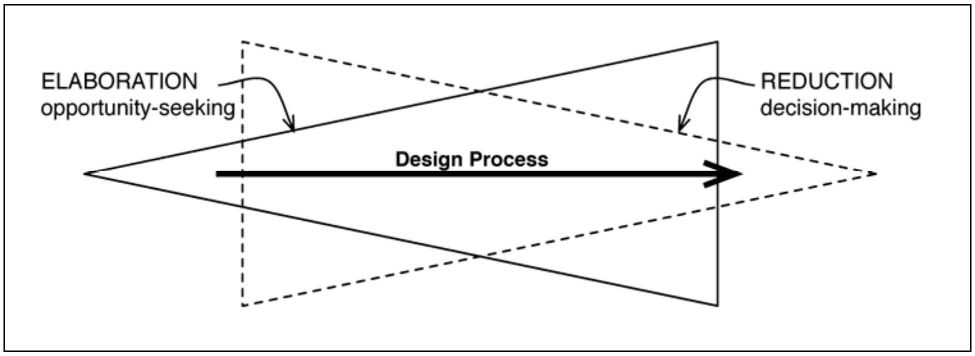

Fig. 1. *Laseau's Model*

In this context, it is necessary to understand sketching as an activity that is not necessarily tied to pen and paper but encompasses any generative design work that share a number of characteristics with pen-and-paper sketching. More specifically, sketches (in any medium) are (among others) quick, inexpensive, plentiful and disposable, and have distinct gesture, constrained resolution and ambiguity [14]. It is obvious that these are properties that prototypes normally don't have.

Buxton [14] draws a separator between sketches and prototypes using a table of opposite concepts, as illustrated in Table 1.

Table 1. *Attributes of Sketches and Prototypes*

Sketch	**Prototype**
Invite	Attend
Suggest	Describe
Explore	Refine
Question	Answer
Propose	Test
Provoke	Resolve
Tentative, non-committal	Specific Description

It is important to understand that while Buxton's definition of the terms 'sketch' and 'prototype' might collide with their traditional use, it makes much more sense. The 'prototype' is much more restricted in Buxton's classification. In the context of Laseau's model, this table explains why sketching is the primary approach for the expanding funnel of elaboration and opportunity seeking, while prototyping is the ideal method for the contracting funnel of reduction by decision-making. Put differently, it shows that we need to teach two opposite sets of skills to HCI designers in training.

Becoming good in prototyping, on the one hand, is based on skills that are usually subsumed under the term ‘problem solving’. Teaching of problem solving skills often makes up the major part of engineering and computing programs. Exercises that let students practice problem solving characteristically have clear and unambiguous objectives, coherent and defined settings and explicit, traceable evaluation criteria. It is understandable that such exercises are preferred by teachers since they are much easier to define and assess and lead to less misapprehension and discussion among the students. Such exercises could be characterized as being relatively ‘low-maintenance’. With the design problem ‘given’, students can go ahead and (taken from Table 1) answer and resolve these problems by testing and refining specific descriptions of a solution.

For sketching, on the other hand, problem solving is of relatively little interest and can even be seen as detrimental. Schön [15] coined the term ‘problem setting’ as a contrast to problem solving, thus describing the (at that time) missing half of this dualism. To train the skills needed for problem setting, coherent, unambiguous, defined and clear exercises are the exact opposite of what is needed. Students need to work on open and even inconsistent assignments, in undefined and ambiguous contexts. Obviously, this type of exercise is much harder to evaluate, and it is often impossible to publish clear evaluation criteria up-front. Performance in the expanding funnel of Laseau’s diagram is often impossible to assess by objective means.

Even when seen as rational activities, problem solving and problem setting require different sets of skills, as e.g. Holt [16] proposes: problem solving as an abstract, symbolic, analytic and verbal activity, versus problem setting as a concrete, holistic, spatial and non-verbal practice.

Still, it is important to see that this is not impossible. Following Laseau, exploration can be conducted systematically. He suggests a number of sketching techniques that are all based on the idea that to change an image enables us to get a new look at them, thereby expanding our thinking [12]. Laseau describes a number of sketching techniques that could be a starting point for the design of exercises, e.g. transforming, structuring and ordering images.

These techniques, however, only work if, as Laseau writes, we are ‘comfortable with exploration that is not tightly focused, let the mind wander, and be open to unexpected results’ [12: p.115]. This might be the real challenge in any kind of education, as McKim [17: p.45] recognizes: ‘For most people, breaking lazy, category-hardened, fear-inducing habits of seeing is an educational task of considerable magnitude’.

4.1.2 Kotzé et al.’s Framework for Teaching and Learning Design

The second part of our theoretical foundation is based in learning theory. People ‘learn’ by repeated exposure to concepts using one of two major *types of learning*: implicit or explicit [13]:

- *Implicit learning*, or unintended learning or tacit (silent) learning [18, 19], can be seen as a passive process where people, when exposed to information, simply acquire knowledge of the information by means of that exposure, i.e. it is unconscious and always active [19-21]. Invoking implicit knowledge involves the indirect application of the knowledge without the requirement of knowledge declaration [20].

- *Explicit learning,* or intended learning, in contrast, is characterised by people actively seeking out the structure of any information presented to them, i.e. it is intentional and conscious [20-22]. For example, explicit learning would be involved if a designer is instructed to acquire some target knowledge and then to explicitly apply and state the knowledge acquired in the design phase [20].

Kotzé et al. [13] proposes a pyramid of competence model for learning (and consequently teaching) design for HCI based on the models of learning proposed by Gorman [23] and Miller [24]. Both Miller and Gorman communicate the concept of different kinds of knowledge building onto each other, and the acquisition of the knowledge being acquired in a particular sequence over a period of time.

The pyramid of competence model, as illustrated in Fig. 2 identifies four *types of knowledge* in design or technology knowledge transfer [13]:

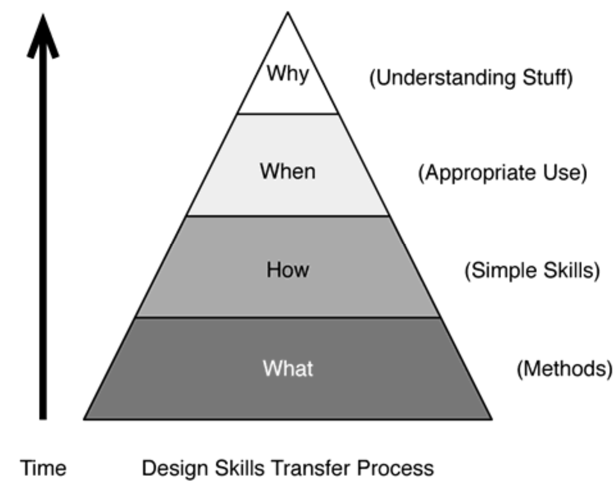

Fig. 2. *Kotzé et al. pyramid of competence model with a link to interaction design skills*

- *Declarative knowledge* (*what*) refers to the recall of facts and events? Declarative knowledge is composed of chunks, consisting of a number of slots each of which can hold a value (which can also be another chunk) [25]. In the context of interaction design this is the process of learning about design techniques and the elements of interaction design (as mentioned in section 3).
- *Procedural knowledge* (*how*) that refers to the skill of knowing how to do something. Procedural knowledge is usually encoded as declarative knowledge first and then translated into procedures (algorithms) [26], but can also be learned by feel or intuition. Procedural knowledge therefore consists of productions,

which are condition-action pairs specifying the action to be taken if a particular condition is satisfied [25]. In the context of interaction design this is the process of learning how to use the design elements and design techniques.

- *Judgement knowledge* (*when*) that involves the ability to recognise when knowledge is applicable to a particular instance, i.e. recognising that a problem is similar to one for which a solution is known and knowing when to apply a particular procedure or solution. Judgement knowledge is therefore structured in a way that facilitates problem solving and creativity, and is usually applied by experts in a particular context. Whereas novices would rely more on declarative and, to a lesser extent, on general or weak heuristics based on procedural knowledge, experts rely more on judgement knowledge [25]. In the context of interaction design this is the process of learning to recognise situations where the previously learnt technique or design element should be applied.
- *Wisdom* (*why*) knowledge refers to meta-cognitive monitoring which may lead to a new course of action. It is related to judgement knowledge referring to the ability to reflect, question, and come up with new courses of action. It involves an element of moral reasoning [25]. Wisdom is the most vague and intimate level of understanding [4]. It is much more abstract and philosophical than the other levels and less is known about how to create or affect it. Wisdom is a kind of 'meta-knowledge' of processes and relationships gained through experiences. We cannot create wisdom as we can with declarative and procedural knowledge and we cannot share it with others as we can these levels of knowledge. In the context of interaction design this is the process of understanding the rationale of the technique or design element, and understanding why it comprises a good and effective design, why when applied creatively it would enhance the user experiences, etc.

In terms of design skills these four *levels of learning* can be defined as [4]:

- *Knows*: Factual knowledge about interaction design elements or simple skills. It is the product of research or creation (such as writing), data gathering and discovery.
- *Knows how*: Methods (ability) enabling the application of the knowledge or skills. It is about presentation organisation.
- *Shows how*: Appropriate use, i.e. the ability to identify situations where skills or knowledge, or particular design elements can be applied. It is about conversation, storytelling and integration.
- *Knows why*: Understanding stuff, i.e. the ability to argue about why a specific skill or method or design element will be appropriate or not. It is about contemplation, evaluation, interpretation and retrospection.

Interestingly, although we have to keep these four levels of knowledge in mind when teaching interaction design, the same four *levels* should also be kept in mind when *designing interactive experiences* [4]:

- The *what* knowledge is just data: It is the raw material we find or create and then use to build our communications. Most of what we experience is merely data. It is often boring, incomplete, or inconsequential. Data isn't valuable as communication medium on its own, because it isn't a complete message.

- The *how* knowledge makes data meaningful for audiences because it requires the creation of relationships and patterns between data. Transforming data into information is accomplished by organizing it into a meaningful form, presenting it in meaningful and appropriate ways, and communicating the context around it.
- The *when* knowledge is communicated by building persuasive interactions with others, or with tools, so that the patterns and meanings in the contained information can be learned by others.
- The *why* knowledge cannot be created, as we can with what, how, and when, and we cannot share it with others as we can these other levels of knowledge. We can only create experiences that offer opportunities and describe processes. Ultimately, wisdom is an understanding that must be gained by oneself.

4.2 A Framework for Design Exercises

The proposed framework for design exercises for interaction design and HCI is based on the theoretical frameworks discussed above.

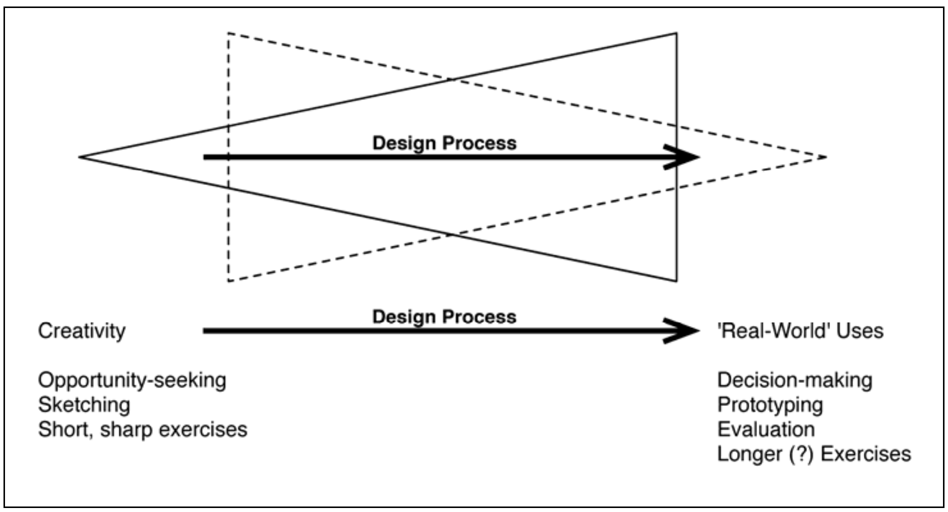

Fig. 3. *Framework structure for design exercises*

Fig. 3 illustrates the idea that design is in some ways a continuum ranging from purely creative work to real-world implementations, i.e. sketches to prototype to use the terminology from section 4.1.1. Design exercises can be aimed at any point of this continuum, either starting at a specific point, or ending at a specific point. There is no wrong or right way of 'designing' design exercises entrenched in this model. When aiming at the creative side, exercises should be short, sharp and varied, requiring students to work with a variety of interaction design elements (see section 3).

If we superimpose the Kotzé et al. [13] pyramid of competence on the reduction funnel of the framework illustrated in Fig. 3, we found that it has a natural fit, as illustrated in Fig. 4. It follows the traditional way we teach in the fields of engineering and computing and the way in which HCI courses has been taught in the past. It also follows the arguments of learning theory, on which the Kotzé et al. pyramid of competence is based.

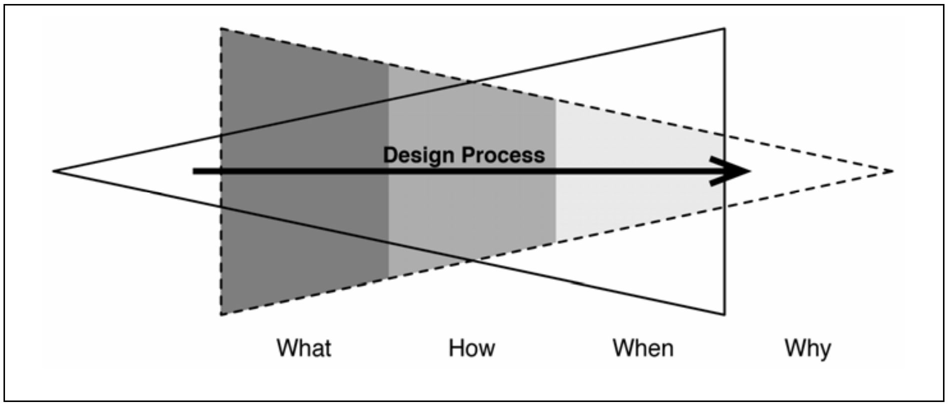

Fig. 4. *Pyramid of competence mapping to reduction funnel*

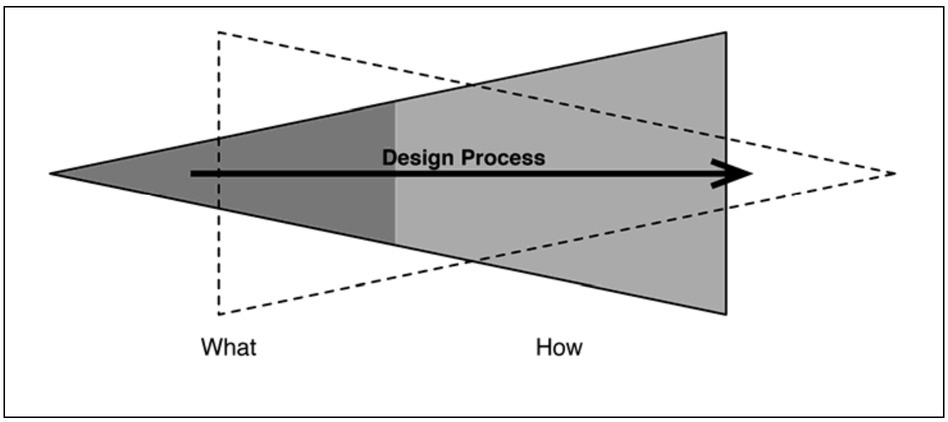

Fig. 5. *Pyramid of competence mapping to elaboration funnel*

When we are creative we take particular thoughts, ideas and elements and connect them together in a fresh way so as to produce an experience which is novel, interesting and possibly of value. The elaboration funnel is therefore different from the reduction funnel in the way it links to the pyramid of competence, as illustrated in Fig. 5. As it is focused on opportunity seeking, it does not necessarily reach the ‘why’ or even the ‘when’ stages, although they are not completely excluded (in fact the reduction funnel will take over to reach these stages). The primary focus of the elaboration funnel is using skills and methods to manipulate the skills (‘what’) in a new way ('how') to create something that is not pre-specified (and not necessarily usable or useful).

Creativity is therefore a judgement of a different kind, and involves a product, process, and a situation, the same aspects that are often present in the reduction funnel, although often used in opposite ways of a range.

Table 2. *Sim and Duffy's knowledge transformers*

Knowledge Transformers	**How is knowledge transformed**
Abstraction / detailing	Abstraction generates a new, less detailed version through the use of abstract concepts and operators. Detailing generates new knowledge with more detail.
Association / disassociation	Association determines a dependency between entities based on some logical, causal or statistical relationships. Disassociation asserts lack of dependency.
Derivations / randomization	Derivations derive some knowledge from another piece of knowledge (based on some dependency between them), Randomizations transforms one knowledge segment into another by making random changes.
Explanation / discovery	Explanation derives additional knowledge based on existing domain knowledge. Discovery derives new knowledge without existing underlying domain knowledge.
Clustering/ decomposition	Clustering involves the grouping of past designs according to their similarities when considering certain perspective and criteria. Decomposition removes the groupings.
Generalization / specialization	Generalization creates a description that characterises the entire concept based on a conjunction of all the specialisations of that concept. Specialisation increases the specificity of the description.
Similarity comparison / dissimilarity comparison	Similarity comparison derives new knowledge about a design on the basis of similarity the design and similar past designs. Dissimilarity comparison derives new knowledge on the basis of lack of similarity between two or more designs.

Sim and Duffy [27] propose a set of knowledge transformers that could assist in making the link between learning and creativity on the design continuum. They distinguish between seven opposite knowledge transformers:

- Abstraction vs. detailing.
- Association vs. disassociation.
- Derivations (reformulation) vs. randomization.
- Explanation vs. discovery.
- Group rationalisation (clustering) vs. decomposition.
- Generalisation vs. specialisation.
- Similarity comparison vs. dissimilarity comparison.

The reduction funnel aims at the knowledge transformer mentioned first, while the elaboration funnel aims at the knowledge transformer mentioned second. Table 2 elaborates the meaning of each of these knowledge transformers.

As already mentioned in section 4.1.1, Laseau [12] describes three approaches to creative exploration:

- Open-ended images that suggest a number of different perceptions or interpretations.
- Transformation of images.
- Structuring or ordering of images.

These approaches can also be seen as knowledge transformers, as Laseau [12: p.115] explicitly states: 'These approaches are aimed at re-centrering visual thinking'. According to Laseau, re-centrering visual thinking means 'unlearning' a certain viewpoint and thus enabling the ability to find new, unexpected viewpoints. Table 3 defines these three knowledge transformers (and provide our own definition for the opposites for each transformer).

Table 3. *Laseau's knowledge transformers*

Knowledge Transformers	How is knowledge transformed
Open-ended images (vs. constrained images)	Open-ended images allow for ambiguity, collage and multi-valency of sketches and invite new interpretations and ideas. Constrained images remove this ambiguity by limiting interpretations.
Transformation of images (vs. preservation of images)	Transformation is invoked by open-ended images, and changes perspective or perception, making familiar seem strange. Preservation maintains the meaning and structure of images.
Structuring or ordering of images for non-specified context (vs. real-world context)	Create artificial context within which new responses can be made vs. specifying a real-world context.

Comparing the two sets of knowledge transformers, we immediately notice some similarities (though not complete overlap), for example, between Sim and Duffy's derivations vs. randomization and Laseau's transformation, and between clustering vs. decomposition and structuring.

Integrating Laseau's model, Kotzé et al. framework for teaching and learning, and both Sim and Duffy and Laseau's sets of knowledge transformers, we propose CTUDE (Creativity to Real-world Use Design Exercises), our framework for design exercises, as illustrated in Fig. 6.

To use CTUDE, the lecturer has to decide where to position the exercises on the continuum and on which funnel, and then plan the experiences to fit this purpose using/suggesting appropriate knowledge transformers. If the purpose of the design exercise is experiencing creativity, one will aim for the left hand side of the continuum and equip one's students with the tools and techniques matching the knowledge transformers suitable for this side of the continuum. If the aim is for students to design a small-scale real-world application, one would design the experience around tools, techniques and knowledge transformers at the opposite side of the continuum.

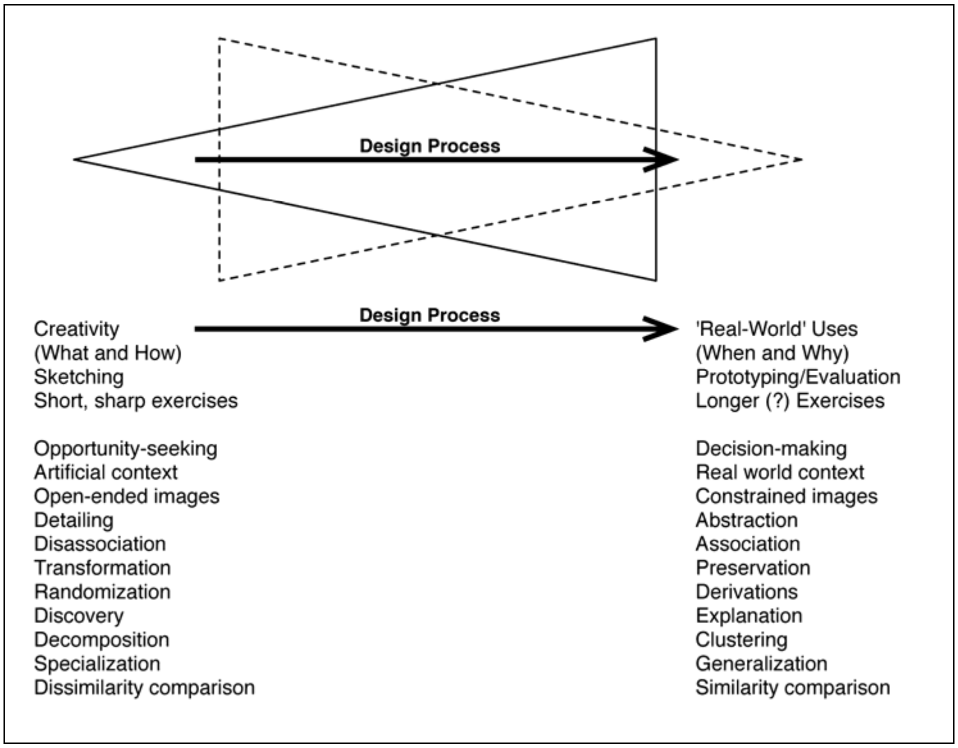

Fig. 6. *Framework for design exercises and knowledge transformers*

5 CONVIVIO Faculty Forum Design Examples

We analysed the various design exercises presented at the CONVIVIO Faculty Forum [3] for their fit onto this framework for design exercises. The exercises varied from purely creative, at the start of the elaboration funnel (for example the Varey, Petrie, Ozcan, and Baumann cultural probe examples), to examples with deliverables at the other side of the spectrum of the reduction funnel with mock-ups or working prototypes (for example the Garay-Vitoria examples). Several of the exercises started off in the elaboration funnel but concluded in the reduction funnel (for example the Oestreicher, Baumann circular handover, Van der Veer, Silva, and Kotzé examples), or were positioned mid-way along the design continuum, requiring students to design or analyse something, but not requiring any kind of implementation (for example the Van Greunen, Jounila, Mavorommati examples). A variety of knowledge transformers were used, with discovery and transformation of images the most popular for the creativity exercises, and clustering, association and similarity comparison the most popular for 'real-world exercises'. We provide detail on two of these examples to illustrate the differences.

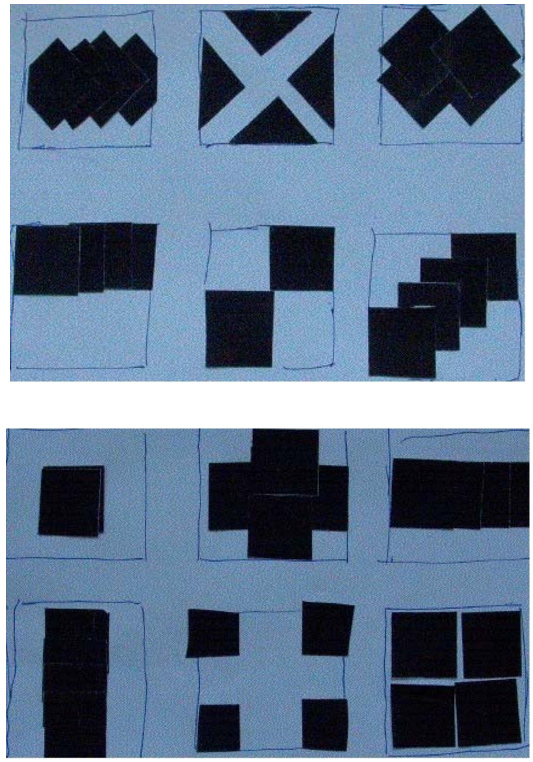

***Fig. 7**. Examples of Graphic Design problem undertaken by students as part of tutorial programme.*

The first is an example provided by Varey [3], using short visual thinking exercises solving communication problems through experimentation and exploration to allow her students to think creatively, and therefore aims at the opportunity seeking funnel. Possible solutions must be generated and concepts represented in a visual form. The exercises are designed not to rely on draughtsmanship skills, but rather to encourage the generation of solutions that have impact and are memorable, visually interesting and communicate the appropriate message. For example, students have to create graphic images, using four black papers squares of the same dimensions, to express the meaning of words such as order, tension, playful and bold. These blocks are their sketching tools. The task intentionally limits the variables to encourage students to be creative and to develop design skills based on the principles of illusory space, framal reference, contrast and the dynamics of relationships. The exercises involve opportunity-seeking, discovery, transformation and structuring of images, all creative knowledge transformers. After completion of the design exercise, it also requires students to select the most effective solution, developing their critical and reflective skills, i.e. moving into the reduction funnel. Fig. 7 illustrates some of the examples produced by students.

Fig. 8. *Visual interface of the reduced keyboard application to be developed.*

Garay-Vittoria [3] describes an exercise for the design of reduced keyboards (the number of keys are less than the number of standard characters that can be produced), aimed at the reduction funnel. Although it starts off with some creative elements, the final aim is to develop a working prototype within a relatively constrained specification. The exercise involves decision making, constrained images, association, preservation and similarity comparison, all aimed at the right hand side of the design continuum. Reduced keyboards usually work on the principle of pressing a key repeatedly to obtain a particular character (e.g. the keypads of mobile phones). For example to type the word 'KEY' you have to press 44229. However, this method has a problem with ambiguities. For instance, if you want to write 'MOON', you have to write 555555555, but this combination can be interpreted as 'NOMO', 'MONO', 'ONMO', 'OMNO', 'NOOM', 'OOOO', etc. Some of the possible interpretations may have no sense in natural language. The exercise requires the students to develop an interface that work with scanning input with a single-button, using a reduced keyboard with 24 keys that are distributed in 6 rows and 4 columns (as illustrated in Fig. 8). This distribution is due to the fact that they use a remote control from Creative (Creative CIMR 100) to compose messages. The interface must be designed so that interface options are selected after a period of time when a key is pressed or when pressing another different key. The selected character is the last one shown in the display. Although pre-specified design instructions for the prototype were given, some students did try to enhance interaction strategy, searching for other operational ways, i.e. moving into the elaboration funnel.

6 Conclusion

The main contribution of this paper is a CTUDE, a novel framework for design exercises with it foundations in graphical thinking, learning theory and knowledge transformers.

Apart from providing a structure for positioning design exercises, the proposed framework also exposes a number of bare spots in interaction design and HCI design education. While some of the aspects covered by the model (especially the 'prototyping' or engineering facets) are typically represented quite well, sketching and especially the creative 'how', are mostly missing from these programmes. Our study also suggests that there is need to extent the problem space, or the knowledge set, over which some of the transformers operate.

One more direction to follow would be to research whether the isolated training of the use of knowledge transformers could make a difference in design education. It would be interesting to see if students that practice the application of knowledge transformers with suitable exercises would benefit in their design skills. Such exercises would be quite simple to design following CTUDE, and, as an additional advantage, quite easy to evaluate. The problem of evaluating designs is often cited because the lecturers find it difficult to assess the methods used in reaching the design: CTUDE will provide a framework to position the designs and the methods used and thus a template for assessing them.

Acknowledgments

Our sincere thanks go to all the participants of the CONVIVIO Faculty Forum, who sparked the idea for the framework during the brainstorming session, and to the CONVIVIO European Network for Human-Centred Design of Interactive Technologies (www.convivionetwork.net) for sponsoring the Faculty Forum financially.

References

1. R. Reimann, So you want to be an interaction designer*, Cooper Interaction Design Newsletter*, **June 2001** (2001).
2. K. Baumann, P. Kotzé, L. Oestreicher, L. Bannon, A. Varey, D. Van Greunen, G. Van der Veer, H. Petrie, I. Jounila, I. Mavrommati, N. Garay-Vitoria, O. Ozcan, P. Purgathofer, and P.A. Silva, EISH - Exercises in Studying HCI, in: *Creativity3: Expereincing to Educate and Design - Proceedings of HCI Educators 2007*, edited by P. Alexandra Silva, A. Dix, and J. Jorge, (Designeed, Aveiro, 2007), pp. 134 – 137.
3. CONVIVIO, Design for HCI, Proceedings CONVIVIO Faculty Forum: Teaching Design for HCI, (cited 2007-01-05); http://www.hcieducation.com/pmwiki.php?n=WorkShops.CONVIVIO2006 (CONVIVIO, Graz, Austria, 2006).
4. N. Shedroff, Information interaction design: A unified field theory of design, in: *Information Design*, edited by R. Jacobson, (MIT Press, 1999), pp. 267 - 292.
5. J. Preece, Y. Rogers, and H. Sharp, *Interaction Design: Beyond Human-computer Interaction* (John Wiley & Sons, Inc., New York, 2002).
6. D. Hill, Architecture and Interaction Design, via Adaptation and Hackability, (cited 2006-12-22); Http://www.cityofsound.com/blog/2006/05/architecture_an.html, 2006).

7. D. Saffer, *Designing for Interaction: Creating Smart Applications and Clever Devices* (New Riders, 2006).
8. D.H. Malouf, Elements of Interaction Design, (cited 2006-12-22); http://synapticburn.com/comments.php?id=143_0_1_0_C (Synaptic Burn, 2006).
9. J. Gibson, The Ecological Approach to Visual Perception, (1979).
10. D. Norman, *The Design of Everyday Things* (MIT Press, London, 1998).
11. P. Dourish, *Where the Action Is: The Foundations of Embodied Interaction* (MIT Press, 2001).
12. P. Laseau, *Graphic Thinking for Architects & Designers* (Wiley, 2000).
13. P. Kotzé, K. Renaud, and J. Van Biljon, Don't do this - Pitfalls in using anti-patterns in teaching human-computer interaction principles, *Computers & Education*, DOI: doi:10.1016/j.compedu.2006.10.003, (2006).
14. B. Buxton, What sketches (and prototypes) are and are not, in CHI'06 Workshop, City, 2006), pp.
15. D. Schön, *The Reflective Practitioner* (MITPress, Cambridge, MA, 1983).
16. J.E. Holt, Practising Practice by Design, *International Journal of Engineering Education*, **18**(3), 256 - 263 (2002).
17. R.H. McKim, *Experiences in Visual Thinking* (Brookes/Cole, Monterey, CA, 1972).
18. M. Polanyi, *Personal Knowledge - Towards a Post-Critical Philosophy* (Routledge and Kegan Paul, London, 1958).
19. A. Reber, *Implicit learning and tacit knowledge* (Oxford University Press, Oxford, 1993).
20. M.W. Kirkhart, The nature of declarative and nondeclarative knowledge for implicit and explicit learning, *The Journal of General Psychology*, **128**(4), 447 - 461 (2001).
21. N.A. Taatgen, Learning without limits: from problem solving towards a unified theory of learning, (cited 2005-06-05); www.ub.rug.nl/eldoc/dis/ppsw/n.a.taatgen/ (Universal Press, 1999).
22. D.C. Berry, *How Implicit is Implicit Learning?* (Oxford University Press, Oxford, 1997).
23. M.E. Gorman, Types of knowledge and their roles in technology transfer, *Journal of Technology Transfer*, **27**(3), 219 - 231 (2002).
24. G.E. Miller, The assessment of clinical skills/competence/performance, *Acad Med*, **65**, 563 -567 (1990).
25. C. Lebiere, D. Wallach, and N.A. Taatgen, Implicit and explicit learning in ACT-R, in: *Proceedings of the Second Conference on Cognitive Modelling (ECCM 98)*, edited by F. Ritter and R. Young, 1998), pp. 183 - 189.
26. J.R. Anderson, *Rules of the Mind* (Lawrence Erlbaum Associates, Hillsdale, NJ, 1993).
27. K.S. Sim and H.B. Duffy, Knowledge transformers - a link between learning and creativity, Learning and Creativity Workshop - 2002, (cited 2006-12-22); http://www.cad.strath.ac.uk/AID02_workshop/Workshop_webpage.html, Cambridge, UK, 2002).