Sketching Sketching:
Outlines of a Collaborative Design Method

Dr Brock Craft
London Knowledge Lab
23-29 Emerald Street, London WC1N 3QS, UK
+44 (0)20 7763 2137

thatbrock@gmail.com

Dr Paul Cairns
University of York
Heslington, York, YO10 5DD, UK
+44 (0) 1904 434751

p.cairns@cs.york.ac.uk

ABSTRACT
In this paper, we describe three key areas in the literature where sketching has been seen as being beneficial to designers. We applied this knowledge in the user interface design of a visualization system, conducting a qualitative study using sketching and design patterns. Our findings help to identify why sketching was useful in this context and we relate these to the literature on the efficacy of sketching in the design process.

Categories and Subject Descriptors
D.2.10 [**Design Methodologies**].

General Terms
Design, Human Factors, Theory.

Keywords
Sketching, collaborative design, visualization

1. INTRODUCTION
We report here on the use of sketching in a collaborative design project for creating information visualization software in the domain of computational biology. The literature in information visualization research has little to offer in terms of methods to approach the early phases of design. Drawing upon experiences and literature in Human-Computer Interaction, we realized that sketching could play a central role in our design activities. Sketching is such a successful method for creativity and problem-solving that it has traditionally been central to the design-oriented disciplines of architecture, engineering, and visual communication. It is also a significant part of the pedagogy of these disciplines [1] [2]. However, the reasons that sketching is effective have received little attention. Our research has uncovered some of the evidence that outlines why sketching is useful and why it is therefore so central to design activities in these disciplines. We describe here the three key benefits of sketching to designers and show how we applied it in our information visualization design project. Preliminary results from this work are discussed.

Sketching is regarded as an important part of design activity, primarily because it supports cognition in ways that lead to creativity. In addition to the practical experience of designers over the years, which has shown that sketching is useful – indeed essential – to design activity [3], there is also experimental evidence to support this assertion. This evidence is based upon research in cognitive science and psychology, which has shown some of the benefits that sketching provides.

2. BACKGROUND
In its most general form, sketching is a way of playing with ideas. It is a visual-motor activity that involves visually rendering the world of ideas that are in the mind of the person doing the sketching. It is a way of externalizing the internal. Whatever the medium, sketching involves an activity of creating and eliminating, modifying and compromising, and weighing possibilities. It permits creative exploration of ideas that may not be fully formed, but where there may be a specific goal in mind. Current research has provided concrete evidence for the value and mechanisms of sketching in three ways: by empirically demonstrating the design value of sketching, articulating the cognitive support it provides, and describing how this is useful for fostering creativity.

2.1 Experimental Evidence
A small number of empirical experiments have sought to uncover measurable evidence for the design benefits of sketching. Of these, three notable examples have showed the value of sketching in: producing better designs, supporting design thinking, and enhancing collaboration. Schütze, et al., [4] report the results of a study that showed that sketching during design activity produces significant benefits. Their experiment evaluated the use of sketching by industrial designers to solve the problem of creating a backyard barbecue grill. Participants were asked to create a grill meeting specific constraints. The resulting designs were evaluated against metrics produced by a panel of expert designers. Participants were either prohibited the use of sketching, permitted sketching for part of the design time, or allowed free use of sketching until they arrived at a solution. The results showed that groups using sketching produced designs of significantly better functional quality. Participants who used sketching reported experiencing less difficulty in the design process, and that sketching acted as an aid to memory during problem solving.

Van der Lugt [5] conducted an experiment, which analysed the functions of sketching, as opposed to written notes, during meetings for generating design ideas. Employing a technique referred to as 'brainsketching', participants generated individual sketches and presented them for group evaluation. These were used as a source of inspiration for additional iterations of sketching and evaluation activity. Sketching was found to be valuable because it supports individual re-interpretive cycles of

idea generation, and enhances individual and group access to earlier ideas.

Heiser et al., [6] also studied the use of sketching to enhance the activity of collaborative design groups. They conducted an experiment in which participants designed route plans (i.e. maps from point 'a' to point 'b') both face to face and remotely. Participants in the study who used collaborative sketching produced more efficient routes in less time than groups who were not able to do so. Their work also showed that by providing an external representation with which team members could interact, sketching enhanced collaborative activities. Sketches established a focus of attention for design groups and eased communication.

Beyond these studies, Landay and Myers [7] note that empirical research has shown sketching to improve user interface design in two important areas: design evaluation and formation of ideas. It aids evaluation by preventing designers and teams from focusing on trivial issues such as fonts and alignment and allowing them to concentrate on the larger conceptual issues at hand. It encourages ideas by supporting lateral thinking and 'allows the designer to focus on the proper design issues'. The work of these researchers suggests the value of sketching by demonstrating empirically that sketching produces measurable improvement in designs, but they do not address the creative dimension of design activity, which is a key element of the design process.

2.2 Cognitive Support

Design activity is often characterized as a process of creating a new form to meet specific needs. According to many authors, sketching is essential to design creativity. In terms of creativity and cognition, two concepts recur in the literature: lateral transformations (i.e., lateral thinking) and cycles of generation and interpretation, or "dialectic". These two concepts are seen as cognitive benefits which sketching provides and which foster activities generative of creativity.

The notion of 'lateral thinking' or 'lateral transformations' refers to related but distinctly different drawings, which are produced during sketching activity. A 'dialectic' refers to iterative cycles of idea generation and consideration of alternatives, using sketching. A designer can use sketching to bring forth into the world an idea to be considered, shared, embellished or perhaps discarded.

The idea that sketching acts as a cognitive aid has been suggested by Plimmer and Apperly [8], who observe that sketching is useful 'as a cognitive support tool during the design process' which aids memory, makes mental images concrete, and enables the designer to 'describe the overall concept and then reorganize, refine and explore the details'. This allows an unstructured problem to be slowly modified and resolved into a final design, and 'makes good use of our innate visual intelligence'. Fällman [9] also describes sketching as having important cognitive effects because it parallels designers' thinking processes. He sees sketching as 'not primarily a tool, technique, or skill that is available to designers, but rather as the way in which designers think'. Because of this, Fällman (like Buxton), considers sketching to be archetypal of design.

Tversky et al. [10], also argue that there is a cognitive basis for the use of sketching in the design process, drawing a direct connection to creativity. They characterize sketching as a means of constructive perception: 'Rather than inducing uncertainty or confusion, ambiguity in design sketches is a source of creativity, as it allows re-perceiving and reinterpreting figures and groupings of figures'.

Other researchers have accumulated evidence of the cognitive mechanisms of sketching during design activity and have focused on the way that designers think about problems. Goel [11] identifies important distinctions between ill-structured problems and well-structured problems, and argues that 'some ill-structured problems require "ill-structured" representations to prevent premature crystallization of ideas and facilitate the generation and exploration of alternate solutions'. Further, he suggests that the uses of ill-structured (i.e. sketched) and well-structured representations are related to different cognitive functions. To make this argument, he describes seven properties of symbol systems for representing well-structured problems, and uses the game of chess as an example of how well-structured problems require accurate symbol systems in order to make them well-structured. By contrast, ill-structured problem spaces not only do not require well-structured symbol systems, but to facilitate cognition about design problems, they require the sort of looseness that sketching provides. Goel bases his assessment upon two protocol studies of design problem solving among industrial and graphic designers, which compare freehand sketching versus a computer based paint program with predefined symbols. The studies showed that freehand sketching generated more lateral transformations than the well-structured representations offered by the paint program. Furthermore, he highlights the ambiguity and density of symbol systems as key properties of representations that are associated with successful problem solving. More ambiguous and dense representations are seen to lend themselves to ill-structured problem spaces.

2.3 Enhancing Creativity

While Goel observes the importance of lateral thinking in generating design ideas, he does not draw direct conclusions about creativity. Suwa and Tversky [12] address this issue head on. They report the results of experiments on sketching activity, which evaluated the complexity and robustness of designed components among novice and experienced designers. Their experiment used a protocol analysis of architects' retrospective reports of their videotaped activities during the design of an art museum. Experienced practitioners used sketches to create rich sets of laterally-related ideas which they refer to as 'dependency chunks', which allowed them to more successfully explore and consider related design alternatives. They report that because of this, sketching enhanced the subjects' creative thinking. This lateral thinking is similar to the lateral transformations referred to by Goel and Landay.

Drawing upon their own work and others, Suwa and Tversky also emphasize the importance of the sketch as an external representation. By physically manifesting a mental idea or image, external representations such as sketches provide three cognitive benefits that enhance creativity: memory, calculation, and modeling. First, sketching aids both long-term and short-term memory. External, sketched representations reduce the load on working memory by '…providing external tokens for the elements that must otherwise be kept in mind'. (ibid, 1997). This allows the

designer to use working memory for kinds of mental activity other than simply bearing in mind the elements under consideration. Drawing on the work of Goldschmidt [13], they observe that externalizations also provide support cognition because they '…remind the user of conceptual knowledge necessary for problem solving and of other similar situations that may promote creativity'.

A second benefit provided by sketches is for '…visuo-spatial and metaphoric calculation, inference, and insight'. The visual, spatial nature of sketches and diagrams makes them effective for '…stimulating visual and spatial associations'. External representation of ideas allows them to be compared to one another based on their physical manifestations. For example, a sketch of a room full of furniture can help a designer to make rough calculations about the size of the room, the spatial arrangement of the objects, the numbers of objects which the room can support, etc. Other judgements such as the visual gestalts of 'proximity, grouping, and common fate' are made possible by visual inspection and comparison. This can lead to insights that are based upon such attributes.

Finally, sketching supports modeling by forcing the designer to make specific decisions about the 'organization, specificity, and coherence' of design ideas, 'which, in turn, by inspection, may lead to new discoveries'. This is similar to modeling things in the real world. But rather than building a physical model, the designer can use sketching to try out ideas quickly and easily by putting them together. In order to make design ideas act in unison, tradeoffs and considerations must be made about what is possible and what is not possible. Grappling with these compromises leads to creative ideation.

In addition to these specific cognitive supports outlined by Suwa and Tversky, a recurring theme is that designers use sketching during iterative cycles of generation and interpretation. Goldschmidt encapsulates this idea by referring to it as a dialectic of sketching, involving a discourse between the designer and the sketch, using two types of reasoning. She draws her conclusions based upon protocol studies of architects. Describing their sketching activity, she makes a distinction between 'seeing-as' and 'seeing-that'. 'Seeing-as' refers to analogical or metaphorical thinking about the sketches, and deriving new meaning from the sketched entities. 'Seeing-that' refers to developing an understanding of the design consequences of proposed sketched ideas. Goldschmidt suggests that this type of sketch-based reasoning occurs in rapid oscillation and is an important component of design activity that leads to creativity. That sketches are, by their nature, vague plays a role here. They are 'sketchy' and allow designers who are 'seeing-that' to reinterpret individual figures and groups. During this activity, preconceived notions of a design problem as represented on paper might be changed after a designer has inspected and reviewed the sketched design. Recombination can facilitate detection of new features, new inferences and insights [14]. In group design, the unstructured nature of the sketch opens opportunities for others to participate and to modify design ideas.

Though he does not address the effect of sketching per se, Schön's [15] work has also examined the dialectical nature of the design process. Schön describes this as 'seeing-drawing-seeing', a cognitive act giving rise to creativity, which characterizes a process of 'reflection-in-action'. By this, he means that in considering problems during the course of their work, architects, engineers, designers, and other professionals engage in design as a 'reflective conversation with the situation'. In this way, design reflection is a cognitive activity which gives rise to creativity. Schön uses a case study to illustrate the properties of this activity. In the study, an apprentice architect engages with the problem of creating buildings for a kindergarten. She uses drawings to help her think about solutions to the problem, as a way of externalizing ideas for consideration.

Rather than analyzing the effects of drawing, Schön's analysis centers on the reflective thinking of the architect. The drawings produced are seen as part of a larger dialectical activity in which she considers the design problems, tradeoffs, and possible alternatives. This results in creative decision-making. It is the process of drawing that reveals the parts of the proposed design that are possible and the parts that are not. For example, the pitch of a hill on the building site has implications for both the arrangement of kindergarten rooms and the amount of light that they will receive. Her sketching of the possible building layout helps her to consider what will be the best compromise among the various contending requirements. Similarly, Robbins [16] and Lawson [2] refer to architects having conversations with themselves about design problems, using sketches and drawings as a means of supporting this interpretive dialogue.

3. RESEARCH

Armed with these perspectives of the design benefits of sketching, we aimed to apply them to the practical problem of designing a new graphical user interface for information visualization software. The purpose was not to measure the benefits of sketching *per se*, in the style of a controlled experiment, but rather to use sketching as a generative design method, in concert with design patterns for visualizations. Our results showed that sketching played an important role in the early phases of design ideation, whereas design patterns were useful in later stages of refinement. We discuss both of these methods here, though our focus is on the outcomes of using sketching in the design process.

Interactive graphical systems such as scientific and information visualizations help people to understand complex data because they overcome limitations in human reasoning, and support knowledge crystallization [17]. But because visualization systems allow for complex interaction with large amounts of information, using novel visual representations, they are difficult to create. What's worse, for the systems developer there are currently few methodologies that comprehensively describe procedures for creating the graphical user interfaces of these tools. In particular, techniques for creating interactions and visual representations early in the design process have not been discussed. This leaves people who seek to create new visualizations with few resources to draw upon in order to build new systems.

Researchers in the discipline are have been laying the groundwork to remedy this situation, in four major research areas: Design Examples, Taxonomies, Guidelines, and Reference Models [18]. These areas of research are attempts to codify visualization knowledge in a meaningful way and to assist others in the creation of visualization systems. They do this by giving solutions (Examples), categorizing and listing artefacts (Taxonomies), recommending best practices (Guidelines), and describing how

visualization systems work, as a whole (Reference Models). Though each offers an increasing level of depth and robustness, none constitutes a methodology. In terms of creating new visualization systems they describe "what" rather than "how to".

For people who are not intimately familiar with the information visualization knowledge domain, the diversity and discontinuity of design knowledge presents a high barrier to understanding how to create new tools and representations. Much of the guidance offered by reference models in terms of actually designing a system is implicit rather than explicit. For example, it is obvious from the visualization reference model of Card, et al. that visual mappings between data and on-screen visual structures need to be made during visualization design. But little guidance about how to do this is offered in the accompanying text. It is also clear that human interaction needs to figure, in some prominent way, in the design process. Implicitly, the process of knowledge crystallization should be supported. But although this reference model accounts for human tasks, it is left to the designer to interpret which tasks they might perform, how they might perform them, and what components should be made available so that users can achieve their goals. It is up to the designer to fill in the gaps in this high-level model, and indeed, in all of the reference models, even if that designer is inexperienced or unfamiliar with the visualization domain. Crucially, the experience and expertise of visualization designers is unaccounted for. The designer's skills, which are necessary to address design problems and generate creative solutions, are almost never the focus of research papers and journal articles.

This raises several questions. Is such design knowledge essential, merely preferable, or indeed necessary at all? Moreover, how can this expertise be shared with a larger community, particularly with non-experts, so that visualizations can be beneficial to more people? Rather than relying upon an array of disparate knowledge sources, is there a useful, principled approach that they can use to create successful visualizations? Understanding how to create visualizations with novel representations and interactions remains a problem. How can this activity be described and supported? What steps are necessary?

One effective way to remedy this problem is to turn learn from the experiences of these other design disciplines. Happily, other designers and engineers have encountered the need to explore, understand, and resolve design problems. Since user interaction and visual presentation are the key elements that set visualizations apart from other kinds of software, user-centered design techniques that support these should be fostered and described by visualization design methodologies..

In the major design disciplines, two user-centered techniques have emerged that may prove useful to visualization designers: sketching and design patterns. In computer science fields, diagrammatic representations are often made with UML or ER diagrams. Using diagrammatic reasoning to enhance design thinking is also beginning to be explored in the visualization community [19]. However, by "sketching", we do not mean diagrammatic representation of the major systems components but rather, simply marking on paper with pens and pencils..

Design Patterns have also emerged as a practical, participatory technique for solving design problems. In the intervening decades since they were first developed in the domain of architecture, computer scientists and software engineers have adopted the idea of design patterns as a useful way for structuring knowledge about the design of computer software. Most notably, Gamma et al. [20] have developed a set of software patterns that attempt to bring some order to knowledge about design of software systems. These patterns have been taken up by others, particularly in the HCI community [21] [22], who have used them to create their own solutions to software engineering problems and who have extended them to address new problems and solutions. Building upon this, Wilkins has developed a set of 36 design patterns for Information Visualization [23]. The purpose of these is to overcome the limitations presented by heuristics and guidelines, to begin to bring some structure to the knowledge of visualization design solutions, and to aid in the software design process of creating novel visualizations.

3.1 Method
In selecting a method for inquiry, we wished to investigate the whole experience of design activity, and to avoid using artificially determined problems. This is necessary to reveal the dimensions of human experience based on actual design activity, rather than the results of controlled laboratory experiments. We therefore selected an open, collaborative research approach suitable for exploring design problems "in the wild", and informed by the methods of Action Research (AR) [24, 25, 26]. Action Research seeks to reveal this information by ensuring that the original context is preserved. That is, the researcher comes to the organisation to do the research, rather than setting up controlled conditions in a lab. The key assumptions are that social settings cannot be reduced for further study and that action brings about understanding in social systems.

In addition, the research is conducted with the full consent and awareness of the participants of the study. Further, participants are encouraged to assist in making decisions about the research process and about the outcomes that are meant to be achieved. The researcher does not observe as a "fly on the wall" but actively engages with participants. Guided by our research objectives, we engaged participants in the selection of design tools, the design process, and our approach to data collection. Indeed, we were accepted by our participants as active participants in the research process.

For data analysis, we used a Grounded Theoretical approach (GT) [27], which has developed from considerable work in the social sciences, and which has gained wide acceptance in the literature of Human-Computer Interaction (e.g., [28], [29]) and socio-technical systems engineering (e.g., [30], [31]).

Our research was part of a two-pronged investigation, which included both freehand, collaborative sketching and information visualization design patterns. The aim was to develop theory about the effects of using sketching and design patterns to support creativity, ideation, and problem-solving for novel visualizations. To do this, we studied a group of designers working on a real visualization design problem, using GT. The resulting data were analyzed to reveal factors surrounding sketching and design patterns at design time, and to build theory about their relevance to creating novel visualizations. In this work, the purpose of using GT was to produce such a theoretical description of the emergent factors involving ideation and problem-solving for visualizations.

3.2 Participants, data collection and analysis

The design group under study was a within computational biology project called the "Beacon Project", based at University College London, Centre for Mathematics and Physics in the Life Sciences and EXperimental Biology (CoMPLEX). The Beacon Project team members were creating an abstract visualization of an integrated computational model of liver processes. The creation of such a model involves advanced mathematical knowledge that can accurately express how real liver cells behave in very specific circumstances. As metabolic processes occur, the activity among liver cells involves dozens of different biochemical signals and changes, which fluctuate over time. To describe this behaviour, very complex mathematical models must be constructed. The team's objective was not to create a scientific visualization of a liver, but rather, an interactive visual abstraction of information within a computational model of various parameters of liver function. The visualization would aid communication among team members leading to refinement of the model's accuracy. The nine design team members included stakeholders from all parts of the project, including software engineers, mathematicians, and biologists. These participants met in participatory, collaborative design sessions over a period of several months. The data collected for this study were obtained during 4 two-hour design sessions over a three-month period. Following the four design sessions, one final requirements specification meeting was held to define possible visualization projects resulting from the design activities.

During the study, the team employed a design and review process following conventional models of iterative software design, prototyping, and evaluation. The design techniques under study, sketching and design patterns, we used throughout this process. The sketching tools consisted of a large (A3) drawing pad, as would be used for figure drawing or watercolour painting. A selection of coloured pencils, some drawing pencils of varying hardness, and an eraser were also used. The rationale for this was to try to avoid any electronic devices so that the tools themselves would not become a barrier to trying to sketch out ideas. We also used the visualization design patterns during the sessions. For ease of use, the design patterns were printed in colour onto 36 A4-sized laminated plastic cards. They could be shared conveniently among team members, stacked into categories, or set aside for later use. These design activities occurred in a small conference room and the primary data collected consisted of the sketches made by the group and audio recordings.

The data analysis was conducted according to the grounded theoretical methods described by Strauss and Corbin (ibid.). The total duration of the design sessions was 8 hours, 14 minutes, with an average session duration of 2 hours, 3 minutes. Approximately eight hours of audio recordings of the four design sessions were transcribed. Sketches, in the form of 17 A3 pages, were scanned and converted to digital files. These data items, the primary data set, were imported into the Atlas.ti software system, which is commonly used for GT data analysis. Following the methods for coding described above, the conversations were analysed and then coded. To reduce ambiguity and for consistency of coding, each of the *open codes* was clearly defined. The number of instances of each code is referred to as its density. These open codes or categories, were then aggregated to identify their sub-categories during the *axial coding* process, which links categories at the level of their underlying processes and dimensions. The axial codes were then used to guide the development of a *central category*. When no further analysis will reveal additional instances, the category is said to have reached "saturation". The central category is the result of theory development about the phenomena surrounding the use of the SoViz approach by the Beacon Project design team.

3.3 Results

Results from Grounded Theory research are typically presented in the form of conversational excerpts, as well as in theoretical propositions arising from coding the transcribed data. Following this convention, we first present notable excerpts from the design sessions, followed by our summary coding and analysis of the activities across the design sessions. Relevant sketches are also presented to support our interpretations.

Because of the volume of data generated by such investigation, it is not possible to present all of the results in this article. Therefore, we must highlight the most promising area of our research relevant to sketching, that the efficacy of sketching in particular is also reflected in the general design literature on sketching, which we will discuss in the next section. At this point, we offer some key examples, which are representative of the data upon which we base our conclusions. We show how sketching was used, and through our grounded theoretical analysis, how it takes precedence over design patterns, during early design work.

3.3.1 Observations

A recurring observation was that sketching was effective because it enabled participants to move ideas from their internal thought processes to a public space where they could be explored and modified. While many participants on the project had privately arrived at good ideas about solutions to specific problems and had perhaps encountered novel solutions in their research, these ideas tended to remain internalized. They had not yet shared them with one another on the project. This excerpt from the conversation in Design Session 4 serves to illustrate this point:

Design Session 4 (4:627-633) (design session : line numbers)

627	B:	In other words, you take the ideas out of your head and put them somewhere so that you can look at them and act on them.
628	P:	Yeah. And then you can stand on them and get higher.
629	J:	And I don't think you would have come up with the idea of making that disappear when you put something into that, if I hadn't –
630	P:	No. You can't see it quite so much. I don't see pictures or words in my head, I see abstract concepts.
631	B:	You wouldn't have come up with the idea of making this triangle shape disappear without having it drawn on the page.
632	P:	Yeah.
633	J:	Yeah.

This passage shows how sketching allowed individual ideas to become part of the collaborative problem-solving process. Sketching ideas together facilitated design decisions and allowed team members to modify solutions to more tightly address project

needs and user tasks, as illustrated by J's remarks in 4:629. Moreover, the participants reported that sketching enhanced their effectiveness in a way that was not possible through verbal dialogue alone. Sketching had prompted changes in thinking about and exploring design problems, which would not have otherwise occurred.

The team members offered many subjective opinions about the benefits and drawbacks of using both sketching and design patterns. Particularly in the early design work, sketching was the preferred design technique. Three examples serve to show how participants felt that sketching was useful to them. In the first example, J suggests that sketching supported creating a new idea:

Design Session 4 (4:617-622):

617   B: So let me ask you this: if you're used to thinking in that way, and not visually, then in what way – if any way – did using this drawing part, what way was that helpful in this process?

618   J: It was really helpful. Let's take an example from today. I was going to describe this bit –

619   B: The puzzle piece. [Figure 3.1, below.]

620   J: – in words. I was thinking of it mainly algorithmically. And I was thinking of it in terms of the maths that you would need to do to wrap the RC interface into the WR [Waveform Relaxation] interface. And then when you made me write it down, I was forced to draw it like this. And this idea would never have come to me, if you hadn't made me do that.

621   B: So you actually had a new idea as a result of –

622   J: And I would not have done that if you had not said, 'No, [J], draw it'.

In the second example, the design team were able to use sketching for creative purposes. This is an example of using an abstract visual metaphor to constrain user input. One visualization idea involved allowing non-mathematicians to construct mathematical models from sub-components. However, the mathematical models in the software have numerous parameters that only interact in certain ways. The team were looking for a way to force users to construct only valid models out of constituent sub-components but needed to allow users to match interfaces among different models. Through sketching out this design problem, they were able to arrive at a jigsaw-puzzle metaphor that would constrain the user's interaction possibilities. Figure 3.1 shows this novel representation, which was created by the participants early in the course of exploring this problem.
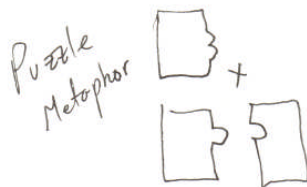

Fig. 3.1: Creating a visual metaphor for software behaviour.

In the third example, J explains that the design team were able to

use sketching to solve complex problems in order to create more advanced features:

Design Session 3 (3:643):

643   J: But in terms of how this has also benefited us, I'd say that it's allowed us to develop more advanced features. Because you say, 'Oh what we had last time is this and we can build that'. And it's much easier to express an advanced functionality when you've got written down the basic functionality. Otherwise you have to − you're building some abstract concept within someone's head and you have to explain to them the bits that you've got so far before you can add on the pieces at the end. Whereas, if you can point to it on the piece of paper and say, 'This is what we've got so far. Now what we're going to add is that'. then you don't have to build such a large abstract concept in a person's head.

These examples, among many others, illustrate how sketching was useful for the design team. However, sketches also presented occasional difficulties. In this excerpt, the participants identified one of the key drawbacks to using sketching in the context of designing for dynamic interfaces: that sketches are static, whereas a visualization is, by definition, dynamic:

Design Session 3 (3:625-626):

625   J: See, I find that what we've been trying is to draw computer screens and dynamical evolution of computer screens on pieces of paper and I find that quite difficult.

626   P: Yeah, that's true. That's why we've got all the stupid things where it goes like that. And the box like that. And this thing with the person with the brain looking at the two bits of information.

Additionally, there were mixed opinions about the use of sketches as an historical record. One drawback was that although sketches were a useful reference during design time, it was sometimes difficult to interpret their meaning after extended periods of time. Compare the following opposite opinions regarding this, in excerpts from design sessions 3 and 4:

Design Session 3 (3:605-606)

605   B: You didn't talk about the drawings that we did. What did you think of those?

606   P: I think that those were amazingly helpful because they form a record of what we did.

Design Session 4 (4:280-282):

280   B: The question is, 'Do the drawings in the longer term, have some utility as well? Can we still talk about them outside that original context? Do they still carry the same meaning that they did?' Or are they merely a tool at the time and once we've done them, we've dispensed with them. And if you can't even identify what you were talking about in a drawing, then that would support the hypothesis that –

281   J: I don't know what that one was!

282   P: Oh, those are just sliders!

283   J: I know, but it's a rectangle with some lines on it.

These examples illustrate many of the benefits and some of the drawbacks of using sketching. It aided communication by facilitating the sharing of ideas, supported design ideation and creativity and aided problem-solving. Further GT analysis provides perspective on the relationship of sketching and design patterns to each other, during the design process.

3.3.2 Grounded Theoretical Coding and Analysis

The key finding of the analysis, expressed as a central category, was that sketching and design patterns support design activities by facilitating the elaboration and reduction of alternatives at key stages of the design process. The central category describes a relationship between sketching, design patterns, and the design process. Whilst sketching was useful for the opportunity-seeking, elaborative activities at the early stages of design activity, design patterns proved most useful in the reductive, decision-making phases towards the end of the design activities. Though the elaborative activities involve opportunity seeking and exploration, the reductive activities tend to require decision-making and elimination of alternatives so that a final product can be produced.

This is complementary to Buxton's model of design activity [3], which describes overlapping funnels of the design process. In this view, design consists of activities involving elaboration of design alternatives, followed by reduction of alternatives. Extending this design model to visualization, a theoretical model of design activity emphasizes the importance of sketching for the early stages of problem solving and exploration and situates design patterns at the later stages of the process. This interpretation of results is based on the times during the design process that the team was able to use these tools (Availability).

3.3.3 Availability

Timing played a significant role in how sketching and design patterns were used. The majority of categories representing processes of elaboration and exploration were those related to sketching activities. By contrast, a larger number of categories relating to decision-making and reduction of alternatives involved the use of the design patterns. Three factors are responsible for this. The first is that sketching was an activity that was immediately available to the design team. The familiarity of pencil and paper as a conventional means of communicating ideas made sketching a natural and easy activity for the team to adopt for visualization design. In addition, the physical arrangement of the group around a table where each member had easy access to communal pencils and sketching paper made it very easy for participants to use the tools.

Unlike sketching tools, the design patterns were more difficult to understand and use. Thus, the second factor related to availability is that the use of design patterns was not fruitful until the later stages of the design process, because the participants did not have a clear understanding of the design problem before using sketching and had not yet had sufficient time to discuss and apply design patterns. The density of information in the collection of 36 patterns was seen as overwhelming, as one participant observed: 'Yeah, I don't want too sound lazy, but in some ways it might be better if we just talk in this kind of random way and you say, "A-ha! That's a pattern".'. During Design Session 2, the design team decided to not to review all of the patterns and relate them to design problems, but rather to permit the expert on the team to offer timely advice. Later in the design process, however, after several design alternatives had been proposed through sketching, the group decided to revisit each of the patterns as a method of decision-making about design choices and to make sure that no visualization design knowledge captured by the patterns was overlooked. This activity, which occurred at the end of Design Session 3, would not have been possible in the initial design stages. In this way, sketching can be seen as supporting a cycle of interpretation though supporting other design methods.

A third factor relating to availability is that in contrast to sketching, the patterns represent a static repository of information. Sketching is a generative and active behaviour and is inherently oriented towards production. Conversely, the design patterns are oriented towards use, which requires reading and interpretation. They are not generative, in the sense of creating something new, and do not lend themselves to generation of ideas in the same way that sketching does. Even with the aid of a subject-matter expert, as in this case study, the reading and interpretation of patterns requires time, whereas the generation and association of ideas that sketching facilitates can occur immediately. This affected the availability of both methods, in the early and later stages of designing.

4   CONCLUSIONS

While it is intuitively reasonable that sketching is a valuable thinking tool – after all, many of us are likely to doodle idly while considering a problem – this research reaffirms that sketching is a valuable tool for designers of all kinds, including visualization designers. This is because building the user interfaces of graphical visualizations is a design problem. While sketching often may be used in practice, it is not often discussed and remains largely implicitly understood – though this is changing.

A counter-argument to our line of reasoning is that GUIs are somehow different to problems in Architecture or Engineering. However, building applications is called software engineering because software is an engineered solution to a concrete design problem. It involves weighing the technical problems against the various alternatives that lead to a successful software system. Determining the best route through these trade-offs is a classic design activity and one that has been traditionally supported by sketching. It is so much a part of the problem-solving repertoire, that sketching is part of the software design pedagogy.

Another argument is that sketching is so foreign to software engineers that it will be difficult for them to acquire this skill. However, our research on integrating sketching into the design process has shown otherwise [32]. Once any initial apprehension to use sketching is overcome, it is quickly adopted by software design teams, including those team members who do not consider themselves 'design-oriented'.

In summary, we believe that organizations and practitioners should encourage the use sketching as part of the design process, particularly in the early phases. It has several significant benefits. Sketching enhances design problem solving by:

- improving designs,
- supporting the design process,

- enhancing collaboration.

Sketching also enhances creativity by:

- supporting short- and long-term memory,
- supporting lateral thinking,
- supporting modeling,
- enabling cycles of generation and (re)interpretation

We do not mean to suggest that other design methods should be abandoned. Rather, sketching should be seen as an important element in the designer's toolbox. A logical consequence of this is that the design pedagogy should include sketching, as in other design-oriented disciplines. Also, researchers and practitioners should share their experiences through increased reporting of design work, and the use of sketching in developing designs. This argument is also supported by calls to enhance design activity though sketching. Incorporating sketching more fully and visibly into the methods of interaction design will lead to improved design outcomes.

5   REFERENCES

[1] P. Laseau, *Graphic Thinking for Architects and Designers*, Wiley, 2000.

[2] B. Lawson, *How Designers Think, Fourth Edition: The Design Process Demystified*, Architectural Press, 2005.

[3] B. Buxton, *Sketching User Experiences: Getting the Design Right and the Right Design*, Morgan Kaufmann, 2007.

[4] Schütze, Sachse, and Römer, "Support value of sketching in the design process," *Research in Engineering Design*, vol. 14, May. 2003, pp. 89-97.

[5] R.V.D. Lugt, "Functions of sketching in design idea generation meetings," *Proceedings of the 4th conference on Creativity \& cognition*, Loughborough, UK: ACM, 2002, pp. 72-79.

[6] J. Heiser, B. Tversky, and M. Silverman, "Sketches for and from collaboration," *Visual and spatial reasoning in design III*, 2004, pp. 69-78.

[7] J.A. Landay and B.A. Myers, "Interactive sketching for the early stages of user interface design," *Proceedings of the SIGCHI conference on Human factors in computing systems*, Denver, Colorado, United States: ACM Press/Addison-Wesley Publishing Co, 1995, pp. 43-50.

[8] B. Plimmer and M. Apperley, "Computer-aided sketching to capture preliminary design," *Proceedings of the Third Australasian conference on User interfaces - Volume 7*, Melbourne, Victoria, Australia: Australian Computer Society, Inc, 2002, pp. 9-12.

[9] D. Fallman, "Design-oriented human-computer interaction," *Proceedings of the SIGCHI conference on Human factors in computing systems*, Ft. Lauderdale, Florida, USA: ACM, 2003, pp. 225-232.

[10] B. Tversky et al., "Sketches for Design and Design of Sketches," *Human Behavior in Design: Individuals, Teams, Tools.*, 2003, pp. 79-86.

[11] V. Goel, *Sketches of Thought*, The MIT Press, 1995.

[12] M. Suwa and B. Tversky, "How do designers shift their focus of attention in their own sketches?," *Reasoning with Diagrammatic Representations: Papers from the 1997 AAAI Spring*, vol. Technical Report FS9703, 1997, pp. 102-108.

[13] G. Goldschmidt, "The dialectics of sketching," *Creativity Research Journal*, vol. 4, 1991, pp. 123-143.

[14] M. Suwa and B. Tversky, "Seeing Into Sketches: Regrouping Parts Encourages New Interpretations," *Visual and Spatial Reasoning in Design II*, 2001, pp. 207-291.

[15] D.A. Schon, *The Reflective Practitioner: How Professionals Think in Action*, Ashgate, 1991.

[16] E. Robbins, *Why Architects Draw*, The MIT Press, 1997.

[17] S.K. Card, J. Mackinlay, and B. Shneiderman, *Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann, 1999.

[18] B. Craft and P. Cairns, "Beyond guidelines: what can we learn from the visual information seeking mantra?," *Information Visualisation, 2005. Proceedings. Ninth International Conference on*, 2005, pp. 110-118.

[19] C. Ware, *Visual Thinking: for Design*, Morgan Kaufmann, 2008.

[20] E. Gamma et al., *Design patterns: elements of reusable object-oriented software*, Addison-Wesley Longman Publishing Co., Inc., 1995.

[21] J. Borchers, *A Pattern Approach to Interaction Design*, Wiley, 2001.

[22] A. Dearden and J. Finlay, "Pattern Languages in HCI: A Critical Review," May. 2006.

[23] B. Wilkins, ""MELD: A Pattern Supported Methodology for Visualization Design"," 2003.

[24] Baskerville, R., "Investigating Information Systems with Action Research," Communications of the Association for Information Systems, Vol. 2, Article 19, October 1999.

[25] Checkland, P. and S. Holwell. *Information, Systems and Information Systems: Making Sense of The Field.* John Wiley, 1998.

[26] Fléchais, I. *Designing Secure and Usable Systems*, PhD Dissertation, University College London, UK, 2005.

[27] J.M. Corbin and A.C. Strauss, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, Sage Publications, Inc, 2007.

[28] A.F. Monk et al., "New approaches to theory in HCI: How should we judge their acceptability?," *Proceedings of the IFIP TC13 Third Interational Conference on Human-Computer Interaction*, North-Holland Publishing Co., 1990, pp. 1055-1058.

[29] G.J. Elliott, E. Jones, and P. Barker, "A grounded theory approach to modelling learnability of hypermedia authoring

tools," *Interacting with Computers*, vol. 14, Oct. 2002, pp. 547-574.

[30] B. Kaplan and D. Duchon, "Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study," *MIS Quarterly*, vol. 12, Dec. 1988, pp. 571-586.

[31] B. Turner, "The Use Of Grounded Theory For The Qualitative Analysis Of Organizational Behaviour," *Journal of Management Studies*, vol. 20, Jul. 1983, pp. 333-348.

[32] Craft and Cairns, "Using Sketching to Aid the Collaborative Design of Information Visualisation Software - A Case Study," *Human Work Interaction Design: Designing for Human Work*, 2006; http://dx.doi.org/10.1007/978-0-387-36792-7_6.