Human Technology

An Interdisciplinary Journal on Humans in ICT Environments

Volume 6, Number 1, May 2010

SPECIAL ISSUE ON CREATIVITY AND RATIONALE IN SOFTWARE DESIGN

John M. Carroll, Guest Editor

Pertti Saariluoma, Editor in Chief

HUMAN TECHNOLOGY

An Interdisciplinary Journal on Humans in ICT Environments Volume 6, Number 1, May 2010

Contents

Original Articles

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments

Human Technology is an interdisciplinary, scholarly journal that presents innovative, peer-reviewed articles exploring the issues and challenges surrounding human-technology interaction and the human role in all areas of our ICT-infused societies.

Human Technology is published by the Agora Center, University of Jyväskylä and distributed without a charge online.

ISSN: 1795-6889
Submissions and contact: humantechnology@jyu.fi
Managing Editor: Barbara Crawford
Technical Support: Laura Fadjukoff & Fotini Bogiatzi

www.humantechnology.jyu

HUMAN TECHNOLOGY

An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889

www.humantechnology.jyu.fi Volume 6 (1), May 2010, 1–3

From the Editor in Chief

THE RIGHT CONCEPTS FOR THE RIGHT PROBLEMS

Pertti Saariluoma
Cognitive Science, Department of Computer Science and Information Systems
University of Jyväskylä, Finland

The concepts we use, in many ways, influence what we perceive. If a cardiologist investigating the human heart with an ultrasound device shows us the visuals, it is easy to see the same movements of pixels on screen that he or she does. We would see how white and black spots keep flowing in a specific pattern. If the cardiologist points out a white spot as a blood vessel, we could probably discriminate it easily. Over time, we might be able identify the regular movement in a white area as one of the valves and the black area reflecting the blood moving from one chamber to another. We also would see how the numbers along the side of the screen keep changing. Yet, even with these observations, we would not be able to make much of a diagnosis.

What we would not know about this experience is that the ultrasound tool is not necessarily very effective when looking at the vessels of the heart or what the numbers mean regarding blood flow. And we would not even know whether everything is in order. So although we could see various aspects of the heart and blood flow, we would not have the concepts nor the related systems of experts' knowledge, to fully comprehend the images projected onto the screen.

This same reality applies to modern technologies. Today, when ICTs are playing larger and larger roles in our lives, their design and development are becoming very complex issues requiring many types of knowledge. Creators of technologies need to understand electronics, signal processing, and information about raw materials. For example, designing devices for construction work requires the ability to know how to keep the equipment from becoming too dirty too fast. And in contemporary television technology, the focus is to find a way to save electricity, which is a goal quite distinct from the engineers' work a half century ago. As a final example, it is difficult to keep data signals in optimal form as they move around the globe. Thus, it must be remembered how difficult it can be to get technologies to always work reliably in society. Such work nowadays presupposes a deep understanding of the human dimension, which in turn presupposes in-depth knowledge of human research.

The development of technologies in past centuries has had quite a different emphasis. These machines and devices had been special purpose tools, which meant there had been a clearly definable user need that was intuitively recognized and understood through common sense. Even complex technologies such as engines, ships, or paper machines had a very clear

URN:NBN:fi:jyu-20105241903

need to serve and an easily identified user. Therefore they were easily and obviously positioned in the society: Paper is needed by publishers and private persons for a variety of everyday needs. Of course, some analysis was required, for example, to make a good paper handkerchief; specialized uses had their requirements. But these problems were primarily technical: How can it best fill its goal? Is it affordable? Does it look clean and can it stand, for instance, the high humidity or high temperatures in locations where it might be used? These are important questions in the design process, to be sure, but the act of using such a product is relatively elementary and intuitive.

Today's machines, however, are more likely to be technically general-purpose devices. This means that the same technology can be used for multiple—perhaps many multiple—different, and sometimes quite distinct, purposes. As a result, the primary goal is no longer definable in simple technical terms, meaning the physical, electrical, or chemical concepts. While these concepts are essential in creating the devices, they have practically no direct relationship with the actual human use. The set of possible user needs and the ways of using any given technology are growing exponentially in complexity. Because of this new reality, designers can no longer easily rely on traditional technical concepts. In fact, this reality is changing the basic technical concepts in some critical ways. In some cases, these traditional technical concepts can easily block development rather than aid it. This arises because traditional concepts do not help us in seeing what is happening on the "screen" of human life.

At times, the novelty of a design situation has been surprising, and perhaps the designers' concepts and assumptions were not what they should have been. In many of these cases, the design process was lacking sufficient information about human life, human needs and desires, and human interaction. It became clear that the concepts of human science were not implemented within in the technology design. In reality, it takes time to fully develop tools reliable for solving human technology interaction problems. The basic concepts of life and human sciences, therefore, often have been tapped for solving design problems that are connected with human–technology interaction.

In biology, the theories of evolution have been foundational concepts because they explain so many critical phenomena of life (Dawkins, 2009). However, they do not offer much understanding of the problems of human–technology interaction: Evolution operates at a too general level. Meanwhile, theories in psychology have provided insight into very important issues, such as infant-to-adult development and the nature of schizophrenia. Again, while these are vital issues, they have very little to do with human–technology interaction. The same applies to history and literary critique: How could they add their perspective to the whole understanding? Finally, sociologists have done much in identifying and investigating a wide variety of topics and issues that assist in understanding the differences between communities and societies (Tönnies, 1887/2002; Weber, 1922/1978). On the eve of fully developed social media practices, these concepts might become important, if only we knew how.

In general, the development of technologies has resulted in situations in which technical concepts provide very little to assistance to designers and engineers to helping them solve the problems technologies use: Human research has not yet reached a point that clearly articulates what design professionals should do. In our present positions—ever changing between the past and the future—we wrestle with how "what has been" can be readily adapted for what is and what is to come. This can best be observed regarding issues of law, and specifically copyright protection. Emerging technical possibilities create social situations that are not easily resolved through current laws. Copyrights, for example, were made to protect artists. However, technologies today often provide multiple ways to circumvent the restrictions. How do we, as societies and members of societies, address such challenges? Of course, the open-source movement, which values free access and the sharing of ideas and product, does not view contemporary law as the only means of creating and distributing technologies. Balancing the proprietary rights of creators and producers through legal means versus free access is one of the major conceptual changes for contemporary artists and knowledge producers.

We humans—and designers in particular—are living and working in situations that place enormous demands to our conceptual systems. And, in order to continue progressing as a technological species, our conceptual systems must be redesigned. Of course, such an adaptation need not be as revolutionary as were needed following, for example, the development of the printing press or the innovation of the steam engine. But such a renewal in our conceptual systems is required, whether we like it or not: We either learn to see clearly the important phenomena around us—and for us—or we fumble around like blind kittens.

In this special topic issue, we can again see work that has been done to improve our way of conceiving human technology interaction. The six papers published here reflect the work of the eminent John Carroll, our guest editor, as an outgrowth of a collaborative workshop that explored the intersection of creativity and rationale in software design. Each paper explores a perspective on the role of creativity in the application of design rationale, or how rationale can facilitate design creativity. Both are essential when our considering how conceptual systems—as design professionals and ordinary humans—can be expanded.

REFERENCES

Dawkins, R. (2009). *The greatest show on Earth: The evidence for evolution*. New York: Free Press

Tönnies, F. (2002). *Community and society* (C. P. Loomis, Ed. & Trans.). Devon, UK: Dover Publications. (Original work published in 1887; original translation published in 1957)

Weber, M. (1978). *Economy and society* (Vols. 1 & 2; G. Roth & C. Wittich, Eds.; E. Fischoff, H. Gerth, A. M. Henderson, F. Kolegar, C. W. Mills, T. Paarsons, M. Rheinstein, G. Roth, E. Shils, & C. Wittich, Trans.). Berkeley, CA, USA: University of California Press. (This translation based on the 4th ed. in German, J. Winkelmann, Ed.; original text published 1922)

Author's Note

All correspondence should be addressed to:
Pertti Saariluoma
University of Jyväskylä
Cognitive Science, Department of Computer Science and Information Systems
P.O. Box 35
FIN-40014 University of Jyväskylä, FINLAND
pertti.saariluoma@jyu.fi

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

Human Technology
An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889
www.humantechnology.jyu.fi Volume 6 (1), May 2010, 4–10

Guest Editor's Introduction

THE ESSENTIAL TENSION OF CREATIVITY AND RATIONALE IN SOFTWARE DESIGN

John M. Carroll
Center for Human-Computer Interaction
and College of Information Sciences and Technology
The Pennsylvania State University, USA

Creativity and rationale connote two faces of design that are sometimes viewed as complementary: envisioning new worlds through intuitive strokes of innovation versus analyzing reasons and tradeoffs to guide the development of new artifacts and systems. Because it is frequently the case that different practitioners and researchers, and different design disciplines, prize one or the other more highly, there is not only a contrast, but also a lack of integration between creativity and rationale.

Yet looking at the two, it also seems they are indivisible: What would be the point of building and/or using rationale in design if doing so were to result in anything other than greater creativity? And almost analogously, what good would be served by cultivating or purporting creativity that could never be interrogated, understood, or deliberately improved and applied, never be explained or conveyed to colleagues, never be passed on to students?

On the other hand, this is most definitely not to say that the only reason for rationale in design is to enhance creativity, or that sources of creativity that cannot be explicitly articulated (put into words) have no value. Rather, it is to say that designers and design researchers should want rationales and rationale practices that enhance creativity, and should want to be able to understand and to explain their use of creativity to students, to clients, to users, and to other stakeholders.

It is not hard to state how creativity and rationale could fail to have a mutually facilitative relationship. Rationale can easily become an obsession of documentation and formalization, excessively detailing issues, arguments, and alternatives to an extent or in a manner that no one would ever want to revisit, let alone create in the first place. And indeed, rationale practices are often cited as exemplifying a classic rationalist *mis*understanding of what design is about and how it moves forward. Rationale practices that suffocate design by enforcing a tedious documentation burden could appropriately be regarded as undermining possibilities for creativity.

But creativity has its challenges as well. It is sometimes characterized as necessarily arcane, inherently ineffable, and slightly (or even primarily) mystical. But this attitude unambitiously conflates the nuance and intellectual rigor required to pose and investigate subtle questions with reluctance to pose questions at all. It makes it a point of definition (or

URN:NBN:fi:jyu-20105241904

perhaps religion) that creativity cannot be fathomed or explained simply. It is true that such a view of creativity would have few or no implications for understanding, teaching, or practicing design. But we are not forced to this view. Perhaps, like learning, emotion, sociality, and other characteristically human capacities, creativity is embedded in activity, difficult to isolate for analysis, but quite real and principled.

Ironically, and tragically, research on creativity may have inadvertently vindicated the tendency towards know-nothing views of creativity by considering it in austere generality, and (perhaps as a result) producing fairly ethereal and obvious characterizations, for example, the somewhat underwhelming chestnut that creative activity requires both divergent and convergent thinking.

Given how easy it is to imagine, or just to see in the world, that creativity and rationale can have little to offer one another, it becomes all the more interesting to ask whether and how creativity and rationale can have mutually facilitative interactions.

A WORKSHOP ON CREATIVITY AND RATIONALE IN SOFTWARE DESIGN

A diverse group of designers and design researchers met at Penn State University, June 15-17, 2008, to exchange perspectives and approaches, to articulate and develop new research ideas and hypotheses, and to reconsider and reconstruct prior work and results toward new research directions.

The workshop included thought leaders from several software design research communities, such as human–computer interaction design, sociotechnical systems design, requirements engineering, information systems, and artificial intelligence: Mark Ackerman, University of Michigan; Eli Blevis, Indiana University; Janet Burge, Miami University of Ohio; John Carroll, The Pennsylvania State University; Fred Collopy, Case Western Reserve University; John Daughtry, The Pennsylvania State University; Umer Farooq, The Pennsylvania State University; Gerhard Fischer, University of Colorado; Jodi Forlizzi, Carnegie-Mellon University; Batya Friedman, University of Washington; John Gero, George Mason University; Steve Harrison, Virginia Tech; Sal March, Vanderbilt University; Raymond McCall, University of Colorado; Rosalie Ocker, The Pennsylvania State University; Colin Potts, Georgia Institute of Technology; Mary Beth Rosson, The Pennsylvania State University; Al Selvin, the Open University and Verizon; Alistair Sutcliffe, University of Manchester; and Deborah Tatar, Virginia Tech.

The workshop premise was that creativity and rationale should not be opposed worldviews, and that coordinating them and integrating them is a key to having more effectively reflective design practices, and absolutely essential to a serious science of design. Discussions of design in the computer and information science and engineering (aka CISE) disciplines are highly compartmentalized. In software engineering, design is often discussed as if it were nearly algorithmic, whereas in human-computer interaction it is often treated as nearly ineffable art. At a finer level, critical concepts like *rationale* and *creativity* are understood in multiple incompatible ways. Thus, rationale can be a designer's inchoate intent, an analyst's inference about overall intent or significance, a comprehensive representation of the design process (e.g., IBIS; Kunz & Rittel, 1970), or a detailed (e.g., propositional) representation of consequences for various sorts of users (elaborated by empirical results; Moran & Carroll, 1996). Similarly, creativity can refer to the personal experience of being creative (e.g., flow, Csikszentmihalyi, 1996; or eudaimonic well-being, Ryan & Deci, 2001), it can refer to the novelty of strategies and practices employed in design as problem solving, it can refer purely operationally to the proportion of novel ideas generated, or it can refer to the novelty of artifacts and other embodied products (cf. innovation; von Hippel, 1988).

The workshop started with seven orienting questions:

1. When and how can design rationale evoke creativity in design? For example, does/can design rationale function differently (more effectively) in end-user design, participatory design, pair programming/agile design, or open source design communities?
2. When and how can design rationale fail to evoke, or even undermine, creativity?
3. How can the construction of design rationale be construed and experienced as a creative activity? And how can this be enhanced?
4. What tools and methods for rationale can support or enhance the creativity of design products? For example, how much structure should design rationale tools provide/impose to maximize creative outcomes (e.g., contrast QOC, gIBIS, and design blogs).
5. How might valuing the creativity of rationales inspire new forms of design rationale? What would be characteristics of such new forms of rationale?
6. How can design rationale be used in the classroom to motivate and instruct students about reflection, idea generation, and evaluation?
7. What are useful models, theories, and frameworks for understanding and managing the relationship between rationale and creativity in design?

We specifically eschewed starting from definitions: That is such a formulaic workshop activity after all, and can implicitly filter out diversity of positions. But definitions of course crept in. To understand the relationships between creativity and rationale in design, perhaps one must fix a conception of design, creativity and rationale, at least to some extent.

We characterized design as involving the construction of frames or worlds within which designers work. The scope of this construction is broader than merely an artifact. It encompasses the designer's values and intentions, assumptions and knowledge about people and their activity, and the palette of materials and components that can be incorporated.

We characterized design as inherently iterative, that is, iterative beyond the prescriptive sense of "design one to throw away." New purposes, new requirements emerge from a design as soon as it is embodied, and continue to emerge as people (i.e., users) appropriate and adapt the design within their own activities. One way this was put was to say that software "changes the world." Another way was to say that new artifacts change people's expectations and values.

Another way this was described was using the task–artifact cycle: the notion that a design (artifact) responds to activities (tasks) in the world, directly transforming them in some ameliorative manner (i.e., achieving requirements), but also, most likely, introducing other transformations (creating new unanticipated affordances, and perhaps unfortunate side-effects).

We characterized creativity in design as playfulness, pursuing surprise, and unexpected outcomes. Another aspect of creativity in design is empathy: The exercise of putting oneself into the role of another. Another is liminality: Thinking and acting on the border between two contrasting concepts or rules, such as a rapid switching between convergent and divergent modes of thinking.

We characterized rationale in a variety of ways. One was to consider it a design representation: a way of presenting a design that contrasts with other ways (e.g., sketching, software prototypes), and resultingly evokes descriptive tensions (and perhaps creativity).

Rationale can be prospective (i.e., generated within design activity, as an enabling part of design work) or retrospective (i.e., generated after design activity, perhaps even after the design is embodied and in use). This distinction is important because retrospective design rationale can only evoke creativity for subsequent design work. And conversely, one cannot get the retrospective benefit of perspective and reflection just by "capturing" prospective rationale in situ.

We also characterized the role of rationale in design in a variety of ways. Most basically, rationale is a kind of documentation. This is actually a complex and problematic concept. For example, it is clear that there are many possible rationales for any feature, for any decision taken. Which rationale is to be codified? Rationale could be documented at many levels of detail: Should it be relatively sketchy, focusing on key ideas and issues, or should it be highly detailed?

Thinking of rationale as documentation also raises division-of-labor questions such as whose job is it to capture the rationale, whose job is it to validate the rationale, whose job is it use rationale created by someone else. These cost–benefit tradeoff questions arise whenever a workflow involves people extrinsically tasked to create value for others in an organization.

Rationale as documentation might of course limit creativity (see above) by anchoring thought, and limiting divergence or risk taking. But it could also evoke creativity by framing the design world in terms of the issues and choices that are being managed, and perhaps doing this in multiple ways. In other words, codifying the disciplined part of the designer's world might make it easier to problematize the parts of the world that are codified, by labeling them, but it could also make it easier to problematize the parts that are not yet codified, by contrasting them against the provisional frame.

But there are other ways to see rationale. For example, the discussions among stakeholders presenting, analyzing, and perhaps contesting, assumptions, decisions, values, roles, processes, and so on are also rationale. This is Rittel's (Kunz & Rittel, 1970) democratic conception of many authors contributing to making an argument space more visible for all.

Indeed, focusing on design as a potentially—and perhaps even typically—collaborative task changes the way one might characterize the activity of creating and using rationale. After all, collaborators must continuingly create common ground. This is never a matter of once and done. As the shared activity develops, as assumptions and commitments are made as interim outcomes are obtained, collaborators must make these things public at least to the extent required to allow effective coordination of individual contributions.

For example, Minneman (1991) reported that part of design collaboration is reaching agreement about issues that will not be discussed again (at least for some span of time). This is a highly specialized area of common ground management, and one that design rationale could support, just by providing a language to cordon off areas of discussion and debate.

Like most workshops, this one ended up posing, but leaving open, many questions and identifying projects that ought to be undertaken, but have not yet been started. For example, if rationale can support creativity in design through reframing, that is, through helping designers

designers see their design world in alternative ways, what specific properties of rationales can facilitate this function, what are the rules and heuristics of rationales that provoke insights? One future project we articulated was identifying cases where rationale evoked ideas that had not been raised before in a given design process. What are kinds of ideas are they? What kinds of rationale evoked them? What were the design process circumstances in which they were evoked?

THIS SPECIAL ISSUE

A key objective of the workshop was to facilitate longer term processes of scholarly interaction, and the development of more refined proposals, analyses, and results. One result, then, is this special issue of *Human Technology: An Interdisciplinary Journal on Humans in ICT Environments*, which presents six papers developed from presentations and discussions at the Creativity and Rationale in Software Design workshop.

The first two papers examine roles that codifications of design ideas and interactions can play in evoking creativity. In "Critical Conversations: Feedback as a Stimulus to Creativity in Software Design," Raymond McCall analyzes critical conversations among designers and other stakeholders as integrating ideation and evaluation, through both reflection and situated cognitive analysis, to provide feedback about consequences of design decisions that challenges designers to devise new ideas. McCall argues that exploiting the full potential of critical conversations requires rationale methods that are better integrated with software tools. The second paper, by Alistair Sutcliffe, "Juxtaposing Design Representations for Creativity," argues that the concurrent use of scenarios, prototypes and models can evoke creativity by juxtaposing complementary cognitive affordances.

The next two papers address design as collaborative work. In "Promoting Group Creativity in Upstream Requirements Engineering," Rosalie Ocker examines this topic by focusing on negative intergroup social processes associated with status differentials, in-group bias, and majority influence, which are known to undermine group creativity. She shows how creativity can be promoted by group support system tools that incorporate design rationale. Albert M. Selvin, Simon J. Buckingham Shum, and Mark Aakhus, in the fourth paper, "The Practice Level in Participatory Design Rationale: Studying Practitioner Moves and Choices," present a theory of practice, and analytical tools, to identify some of the creative dimensions in expert practice when constructing design rationale visualizations in meetings.

The final two papers examined the role of rationale in the development of design professionals. Janet E. Burge and Bo Brinkman, in "Using Rationale to Assist Student Cognitive and Intellectual Development," address the challenge students experience when they first encounter problems for which there is more than one "right" answer. They found that introducing students to design rationale techniques helped them consider multiple alternatives and to reflect on reasons for choosing a particular alternative. Finally, in "Does Design Rationale Enhance Creativity?" Jing Wang, Umer Farooq and John M. Carroll studied the design processes and outcomes of student teams in an advanced software engineering course. They found that greater use of design rationale by teams was correlated with more creative outcomes. In particular, they found that the comprehensiveness of tradeoff analysis and the feasibility of design alternatives in the rationales were critical to enhancing novelty, persuasiveness, and insightfulness of the designs.

Thomas Kuhn (1962) wrote that “Like artists, creative scientists must occasionally be able to live in a world out of joint” (p. 79). He called this the essential tension: Research always produces anomalies between theoretical concepts and empirical data; the possibility of crisis and breakdown is always present. A routine problem from one perspective can be a crippling counterexample from another. Faced with significant crisis, scientific communities may engage in what Kuhn calls extraordinary science, in which fundamental assumptions are questioned, conventions are abandoned, and innovative practices become routine.

Describing, developing, and fully enjoying the linkages between creativity and rationale in software design will entrain essential tension. Perhaps we are now at the threshold of a period extraordinary science. Indeed, Kuhn’s notion seems appropriate for what has recently been called “a science of design” for software-intensive systems (Freeman & Hart, 2004). Surely, a science of design would have to be extraordinary; it would have to question assumptions, innovate, reorient and recreate itself. The tensions between relatively discursive, qualitative, and conceptual social-behavioral art and science, and relatively formal, quantitative, and device-oriented computer science and software engineering are inherent and abiding. We must recruit it as an intellectual resource and not (only) experience it as a source of interdisciplinary conflict. Further and finally, I think people are indeed attracted to software design in part because it is exciting to live in a world out of joint, and to participate in a perpetually extraordinary endeavour.

REFERENCES

Csikszentmihalyi, M. (1996). *Creativity: Flow and the psychology of discovery and invention*. New York: Harper Perennial.

Freeman, P., & Hart, D. (2004). A science of design for software-intensive systems. *Communications of the Association for Computing Machinery, 47*(8),19–21.

Kuhn, T. (1962). *The structure of scientific revolutions.* Chicago: University Of Chicago Press.

Kunz, W., & Rittel, H. W. J. (1970, July). Issues as elements of information systems (Working Paper No. 131). Studiengruppe fur Systemforschung, Heidelberg. Germany. Also available on-line from the Institute for Urban and Regional Development, University of California, Berkeley. http://iurd.berkeley.edu/sites/default/files/wp/131.pdf

Minneman, S. L. (1991). The social construction of a technical reality: Empirical studies of group engineering design practice. Unpublished doctoral dissertation, Stanford University, California, USA, Department of Mechanical Engineering.

Moran, T. P. & Carroll, J. M. (Eds.). (1996). *Design rationale: Concepts, techniques, and use*. Hillsdale, NJ, USA: Erlbaum.

Ryan, R. M., & Deci, E. L. (2001). On happiness and human potentials: A review of hedonic and eudaimonic well-being. *Annual Review of Psychology, 52*, 141–166.

von Hippel, E. (1988). *The sources of innovation.* New York: Oxford University Press.

Author’s Note

The Creativity and Design Rationale in Software Design workshop was supported by the US National Science Foundation (IIS 0742392). I am grateful to members of the workshop who helped me develop this paper.

All correspondence should be addressed to:
John M. Carroll
Center for Human-Computer Interaction
and College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA, 16802, USA
Jmcarroll@psu.edu

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

Human Technology

An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889

www.humantechnology.jyu.fi Volume 6 (1), May 2010, 11–37

CRITICAL CONVERSATIONS: FEEDBACK AS A STIMULUS TO CREATIVITY IN SOFTWARE DESIGN

Raymond McCall

Department of Planning and Design

University of Colorado, Denver, USA

Abstract: *Three decades of creating software to support design rationale showed the author how rationale processes can promote generation of novel ideas. Rationale can promote creative design by promoting critical conversations among designers and other project participants. Critical conversations intertwine ideation and evaluation, using feedback about consequences of decisions to challenge designers to devise new ideas. Such conversations take two forms. The first is discussion involving feedback from speculation about consequences of design decisions for implementation and use. The second is discussion involving feedback from actual experiences of implementation and use of the software being designed. The former is purely a process of reflective discourse, the latter a process of situated cognition involving both action and reflective discourse. Thus, the former is pure argumentation, the latter situated argumentation. Exploiting the full potential of critical conversations for creative design requires rethinking rationale methods and integrating them into software supporting implementation and use.*

Keywords: *creativity, software, design, rationale, feedback, situated cognition, action, reflection, planning, reflective practice, design reasoning, argumentative approach, wicked problems.*

INTRODUCTION

This article presents a picture of how feedback-driven rationale processes promote creativity in software design. This picture derives from my three decades of experience in creating software supporting the documentation and use of issue-based rationale for design, that is, the type of rationale pioneered by Horst Rittel (Kunz & Rittel, 1970). This picture is not meant to portray all the ways creativity takes place in design, but it does seek to describe crucial processes that have been largely omitted from other accounts of rationale and creativity, especially the former.

To discuss how rationale promotes creativity in software design, it is useful to define some basic terms. In this paper *software design creativity* refers to the generation of innovative, high-quality ideas for the design of software. The term *ideation* refers to the generation of ideas, especially novel ideas, for artifact design. The term *evaluation* refers to

URN:NBN:fi:jyu-20105241905

determination of the value of such ideas. *Feedback* refers to any information about consequences of design decisions that a designer gets from external sources, such as persons or situations. These are narrow definitions, but they serve the purposes of this paper. Note that the definition of software design creativity involves both ideation and the evaluation.

The picture presented here is based on a number of notions that contrast with ideas advocated by others. First of all, it takes a process-oriented view of rationale, while many proposed rationale approaches either eschew process orientation—for example, the question, options, and criteria (QOC) approach (MacLean, Young, Belotti, & Moran, 1996)—or provide only a rationale schema with no indication of processes for eliciting and recording the schematized rationale—such as the decision representation language (DRL; Lee, 1991).

Second, the picture created here is prescriptive in that it not only seeks to record design processes but also to improve them. In particular, it seeks to increase the use of rationale processes that improve design creativity. Not all rationale approaches are prescriptive (Dutoit McCall, Mistrik, & Paech, 2006); some are purely descriptive and seek only to record rather than change what designers think and do, such as QOC (though they might unintentionally improve design).

Third, the picture presented here is based on the view that intertwining ideation and evaluation is a powerful method for promoting creativity. Yet there is much literature both on creativity and on rationale that treats ideation and evaluation as separate phases, that is, not intertwined. Of particular importance here is that Rittel (1966) saw no role for the intertwining of ideation and evaluation in design.

Finally, this paper takes the view that creativity is enhanced if design and its rationale are considered not merely as planning for future action—for example, implementation and use—but also as a type of situated cognition in which design is shaped by feedback resulting from action. Yet, Rittel, who pioneered the field of design rationale, viewed design strictly as planning, in the sense of thinking before acting (Rittel, 1966); he saw rationale as documentation of this preparatory thinking. Most existing approaches to rationale appear to share this view, since they provide no account of rationale being generated in response to actions taken.

The picture presented here of how rationale processes promote creativity in software design can be summarized as follows. Intertwining ideation and evaluation promotes creativity in software design because feedback about consequences of design decisions challenges designers to devise new ideas. This intertwining takes two basic forms. The first involves discussion among designers in which verbal evaluations of proposed ideas prompt them to devise new ideas. The second and more important involves situated cognition in which feedback resulting from actions, especially the actions of implementation and use, prompts designers to devise new ideas.

The commitment to using feedback-driven, critical conversations to promote creativity has crucial implications for rationale methods used in software projects. One implication concerns the type of processes that are modeled. Currently, none of the rationale methods that deal with design decision making explicitly models the ways in which evaluative feedback leads to the generation of new design ideas. When rationale methods cannot model these processes, they not only cannot promote them but may actually discourage them. A second implication concerns the sources of design rationale. Current approaches concentrate almost exclusively on rationale from design discussion (planning). This is sufficient to allow rationale based on speculative reasoning and the experience of previous projects, but not sufficient to allow rationale based on feedback from actions.

The picture of software creativity as being promoted by feedback-driven critical conversations extends and generalizes Schön's (1983) portrayal of design as a conversation with the situation. It is argued here that Schön's notion of design as both reflection and action provides a better picture of the role of rationale in design than Rittel's. While Rittel saw design as purely argumentation, Schön's theory implies that design is what we might call *situated argumentation*, that is, argumentation informed by feedback from action. Yet Schön's theory by itself covers only a small subset of the situated argumentation that stimulates creativity in software creation. Extending his theory produces a more complete picture of how rationale processes promote creative design. Ironically, extending his theory involves adding ideas of collaborative and participatory design advocated by Rittel.

The following sections of this paper expand on the above-stated ideas. The next section explains the background and motivation for the ideas presented here. The section following that explains the prescriptive and process-oriented approach used here to analyze rationale and creativity. I then look at the relationship between ideation and evaluation in both rationale processes and creative processes. I also contrast views of design as planning for action versus as situated cognition. After that, I identify implications for rationale processes that support creativity in software design. Finally, I summarize the conclusions of this paper and look at ideas for future work.

HISTORICAL BACKGROUND

Rittel (Kunz & Rittel, 1970) pioneered the field of design rationale with his work on Issue-Based Information Systems (IBIS). As a student of Rittel's, I devised a new approach to IBIS called Procedural Hierarchy of Issues (PHI; McCall, 1979, 1986, 1991) and began a series of software projects aimed at using PHI to improve the quality of designed artifacts. These projects revealed previously unforeseen potentials and limitations of rationale in design. In particular, they showed how the generation of novel ideas for software can be supported by processes in which the consequences of design ideas are identified. This paper describes what these projects revealed about the connections between rationale and creativity.

The PHI-based projects created the following software:

- PROTOCOL (McCall, 1979), a text-only hypertext system that elicited rationale from users in PHI form
- MIKROPLIS (McCall, 1989; McCall; Lutes-Schaab, & Schuler, 1984), text-only hypertext supporting user-controlled authoring and navigation of PHI rationale
- JANUS (Fischer, Lemke, McCall, & Morch, 1996; McCall, Fischer, & Morch, 1990), a system for kitchen design using loosely coupled subsystems for 2D computer-aided design (CAD), knowledge-based critiquing, and hypermedia for delivery of PHI rationale
- PHIDIAS (McCall, Bennett et al., 1990; McCall, Bennett, & Johnson, 1994; McCall, Ostwald, Shipman, & Wallace 1990), a system for building design using a hypermedia system to implement 3D CAD and knowledge-based agents, as well as authoring and delivering PHI rationale with multimedia

- HyperSketch (McCall, Johnson, & Smith, 1997; McCall, Vlahos, & Zabel, 2001), a pen-based system for designing by creating a linked collection of hand-drawn sketches.

The later systems were designed using lessons learned from the earlier systems. These projects are stages in a larger project meant to find out (a) how rationale can help designers create better artifacts, and (b) what software support is needed for such use of rationale.

In addition to documenting rationale for design of physical artifacts, all of the above-listed systems except JANUS were also used to document rationale for their own design. The experiences of this documentation effort revealed that the ways in which new ideas emerged involved processes not described anywhere in the rationale literature. In particular, the creative rationale processes in our projects were not supported either by Rittel's (Kunz & Rittel, 1970) IBIS or my PHI method. Furthermore, our creative processes were incompatible with parts of Rittel's theory about design processes and problems. This article looks at these differences and their implications for rationale approaches and software supporting creative software design.

The above-listed projects changed my understanding of rationale processes and creativity. To understand how, I should begin by describing what that understanding was at the start. Simply put, it was based on Rittel's (1972) ideas about (a) the need for an argumentative approach to design, and (b) how IBIS was to help achieve that goal. Rittel's advocacy of an argumentative approach was based on his theory that design problems are "wicked problems" (Rittel & Webber, 1973). By this he meant that they are ill-defined and ill-behaved in a variety of ways that, for example, go far beyond the difficulties of "ill-structured problems" (Simon, 1973). Wicked problems systematically violate conditions required for use of rigorous scientific method to understand and solve them. Rittel (1972) therefore called for a collaborative and participatory approach that involved stakeholders in defining requirements and evaluating proposed designs. Instead of relying on the unexplained judgments of "experts," however, he called for a process in which the reasoning of designers was open to inspection and criticism by others. This implied the need for an argumentative approach, that is, an approach in which all of design was treated as argumentation about design decisions.

Rittel used the term *argument* with the meaning of explicit reasoning, and not with the colloquial English meaning of heated verbal disagreement, as in, "We had an argument about who was to blame" (Rittel, personal communication, 1977). In other words, he used the word *argument* with the meaning it has in his native German language as well as in philosophical discourse in English. Unfortunately, his intentions were often misunderstood by his American students. In the later years of his life, he told his colleague Jean-Pierre Protzen that, because of this, he wished he had called his approach *deliberative* rather than *argumentative* (Protzen, personal communication, 1992).

Further promoting misunderstanding was the fact that, despite Rittel's insistence that the term *argument* was not a reference to disagreement, he felt that controversy was an intrinsic part of design and that forceful debate was the most valuable type of design discussion. He devised IBIS not as a general means of handing all argumentation in design but rather as a way of handing disagreement through debate. IBIS centered on the discussion of issues, but Rittel (1980) defined IBIS' issues as controversial design questions. All other design questions he labeled "trivial issues," and excluded them from IBIS discourse.

These days, all issue-based approaches to design rationale, as well as similar approaches like QOC and DRL, have abandoned Rittel's exclusive focus on controversy and adversarial argumentation. Rittel's focus on controversy, however, is more than an interesting historical footnote, because it apparently led him and others to neglect the collaborative, constructive argumentation described here as a driving force of design creativity.

To clarify discussion, it is useful to briefly describe IBIS and to explain how PHI differs from it. IBIS was intended both as a method for discussing issues and as a means for documenting the discussion. For each issue, participants in the design propose possible answers, called *positions*. Arguments for and against the positions are then given, along with arguments for and against other arguments. Finally, an issue is *resolved* by deciding which position to accept. Issues are linked to each other by various relationships to form a connected graph called an *issue map*. In Rittel's (1980; personal communication, 1975) version of IBIS, the inter-issue relationships included *logical-successor-of, temporal-successor-of, more-general-than*, *similar-to* and *replaces*.

IBIS provided no way of grouping issue-based discussions to represent higher levels of granularity in design processes. Thus, for example, the widely used description of design as being divided into larger-scale processes of analysis, synthesis, and evaluation (Lawson, 2005) could not be expressed in IBIS. This was no accident. Rittel (personal communication, 1975) was deeply suspicious of such higher levels of granularity. In particular, he argued that the belief in large-scale phases of design, such as analysis, synthesis, and evaluation, was the hallmark of the first-generation approach to design, which he judged a failure and sought to replace with a second-generation based on an argumentative approach (Rittel, 1972). He insisted that the only sensible level of description of design process was in terms of its *microstructure*—that is, the level of issue-based discourse (Rittel, personal communication, 1975).

Of course, it can be argued that analysis and synthesis might also be found at the microstructural level for the generation of positions on issues. And evaluation is certainly part of IBIS. Perhaps the generation of positions could be divided into processes of analysis and synthesis. Unfortunately, IBIS provided no account of any processes for devising positions. It may well be, therefore, that its picture of the microstructure of design is not complete.

PHI was meant to implement Rittel's argumentative approach more fully than IBIS by including noncontroversial issues and using a better structure for discussion. To accomplish the latter, PHI replaced the interissue relationships of IBIS with two types of *dependency relationships*: *serves* and *leads-to*. The former indicates that the resolution of one issue influences the resolution of another, while the latter indicates that the resolution of an issue influences the relevance of another. In PHI, a single *root issue* represents the project as a whole. Since all other issues are resolved in order to resolve the root issue, they serve the root issue directly or indirectly. PHI modeled design rationale as a quasi-hierarchy of issues connected by serves relationships, that is, a directed acyclic graph with some added cycles.

PHI showed the structure of discussion more completely than IBIS. In particular, its serve relationships provided a way of grouping issue discussions to represent higher levels of granularity of design process structure. These relationships also enabled representation of detailed processes by which positions on issues were devised—including processes of ideation—something not possible with IBIS. While PHI did not use terms such as *synthesis*, *analysis*, and *evaluation* to label its process structures, it did enable the representation of such processes at many different levels of granularity in issue-based discussion.

Because the quasi-hierarchical structure of PHI is far more orderly than the "spaghetti" structure of IBIS (Fischer et al., 1996), it enabled a substantial increase in the number of issues dealt with in a project. Rittel suggested that, for practical reasons, IBIS should deal with no more than 35 issues (Rittel, personal communication, 1975). But most of the dozens of PHI projects undertaken since 1976 involved more than 250 issues.

The initial goal of the series of software projects described above was to extend the use of PHI to all aspects of design, thus demonstrating Rittel's point that the entire design process was nothing but argumentation. A virtue of attempting to create software that achieves such a grand goal is that the attempt can produce feedback from reality that challenges the assumptions on which the goal is based. This is precisely what happened.

A PRESCRIPTIVE AND PROCESS-ORIENTED APPROACH

The central topic of this paper is the way in which rationale processes promote creative software design. More specifically, this paper identifies processes of rationale generation that reflect software life cycle processes that lead to the generation of important, new ideas for software design. In addition, this paper aims both to analyze and to promote such processes. Doing these things is impossible without using a rationale modeling approach that can represent the processes of interest. In other words, it is necessary, to use a process-oriented approach to describe rationale in software creation.

Using a process-oriented approach to describe how rationale promotes creativity limits which rationale approaches can be used. This is because these approaches differ in the degree to which they model process. Most approaches can be broadly categorized as structure oriented or process oriented (Lee & Lai, 1996). Structure-oriented approaches make no attempt to record the temporal order in which rationale is generated in design. They only record the logical relationships between statements, for example, that one statement argues against another. Process-oriented approaches record the temporal order, meaning the history, of the rationale generation, for example, that an argument arose in response to another.

Many approaches to rationale are structure oriented. For example, the authors of the QOC approach (MacLean et al., 1996) are adamant that QOC in no way records the manner in which rationale statements arise during design. The proponents of DRL (Lee & Lai, 1996) generally make no claims about design processes, but they insist that DRL does not deal with processes by which solution ideas are generated, meaning ideation. Certain applications of IBIS and PHI have also been structure oriented (McCall, 1991). In particular, the domain-oriented issue bases created using PHI (McCall, Fischer et al., 1990) and used in JANUS and PHIDIAS give no indication of the processes in which rationale is generated.

Relatively few rationale approaches are explicitly process oriented. IBIS is process oriented in its original form (Kunz & Rittel, 1970; Rittel & Noble, 1989) and in the form used by Conklin, Begeman and Burgess-Yakemovic (Conklin & Begeman, 1988; Conkin & Burgess-Yakemovic, 1996). In addition, when PHI is used to document individual design projects, it typically is used in a process-oriented manner that records the history of rationale creation. Carroll and Rosson (1992) used a very different type of process-orientation. Their rationale approach centers on the processes represented in usage scenarios. More specifically, it documents "claims," that is, user evaluations of the pros and cons of system features, as the users go through such scenarios. I refer to this approach here as *scenario-claims analysis* (SCA).

While process-oriented rationale contains temporal information not found in structure-oriented rationale, structure-oriented rationale generally requires more work to create. The reason is that process-oriented rationale is documented in the order and wording in which it is stated. Structure-oriented rationale must be edited to exhibit its logical structure and eliminate temporal information. Advocates of the structured approach, such as the authors of QOC, argue that it is worth spending the extra time to design the rationale statements and structure because it facilitates understanding (MacLean et al., 1996).

Since my analysis is process-oriented, it must employ process-oriented rationale methods. As is explained in the next section, the experiences that led to the understanding of how rationale relates to design creativity involved a series of projects that designed software supporting PHI and used it to document the software design. It seems only appropriate, therefore, to use PHI as the primary basis here for the analysis of rationale processes that support creativity in software design. But, since my analysis attempts to show how feedback from users promotes design creativity, SCA (Carroll & Rosson, 1992) also has a crucial role to play.

IDEATION AND EVALUATION: FROM SEPARATION TO INTERTWINING

Ideation and Evaluation in Design Rationale

In most approaches to design rationale—IBIS, QOC, and DRL being well-known examples—ideation takes the form of the generation of alternatives for decisions. In IBIS and its PHI variant, decision alternatives are *positions* and the things to be decided are *issues*. It should be noted, however, that not all issues in PHI deal with decisions about features of the artifact being designed. Any question arising in design is considered an issue, including questions about facts, goals, concept definitions, causes of problems, and effects of decisions. None of these other types of issues involve ideation as it is defined above.

QOC differs from IBIS in that it only deals with decisions about features of the artifact being designed, that is, decisions that involve ideation. In QOC the decision alternatives are called options and the things to be decided are called questions (MacLean et al., 1996). DRL is quite similar to QOC in many respects, but its decision alternatives are simply called alternatives, while things to be decided are called decision problems. From the examples that Lee (1991) gives, it appears that DRL's decision problems are identical to QOC's questions and thus deal exclusively with decisions about features of the artifact. As mentioned above, however, Lee and Lai (1996) make a point of stating that DRL does not represent ideation processes.

Evaluation in most rationale approaches is done by identifying pros and cons of decision alternatives. In IBIS and PHI this is done by stating arguments for or against the alternatives (positions), while both QOC and DRL perform evaluation by assessing how well the alternatives satisfy given criteria (called goals in DRL). In these and other approaches, the evaluation can be augmented by the stating of arguments that support or attack the statements of the pros and cons.

The Separation of Ideation from Evaluation

The Separation of Ideation and Evaluation in Approaches to Creativity

Literature on creativity frequently emphasizes the value of completing ideation before evaluation begins. The main argument for this phased approach is as follows. Criticizing ideas as they are generated inhibits the elicitation of new ideas, especially innovative ideas, which can sound risky and are often vulnerable to attack as first stated. Fear of being attacked can make people reluctant to propose creative ideas; so evaluation should be postponed until after ideas are generated.

The well-known creativity-enhancing methods known as brainstorming (Osborn, 1963) and lateral thinking (de Bono, 1973) focus on ideation. In both cases, it is treated as separate from evaluation. In fact, both methods have explicit prohibitions on evaluation during ideation, so as not to inhibit the free flow of ideas. In brainstorming, this prohibition is called "suspension of judgment" (Michalko, 2006) or "withholding criticism" (Osborn 1963). In defending this prohibition in lateral thinking, de Bono (1973, p. 7) explains, "One is not looking for the *best* approach but for as many *different* approaches as possible." He even adds, "In the lateral search for alternatives these do not have to be reasonable" (p. 7). Both approaches emphasize quantity over quality, in the belief that quantity leads to novelty. The writings of Osborn and de Bono have been very influential; thus many other creativity techniques come with warnings about not evaluating ideas as they are generated.

The Separation of Ideation and Evaluation in Rationale Research

Rittel's (Kunz & Rittel, 1970) work on IBIS has also been influential. Conklin and his colleagues have done extensive work with IBIS (Conklin & Begeman, 1988; Conklin & Burgess-Yakemovic, 1996). And PHI (McCall, 1979), of course, is a revision of IBIS. In addition, the Potts and Bruns (1988) approach to rationale is a revision of IBIS with the goal of fitting it better to software engineering. DRL is a revision of Potts and Bruns (Lee, 1991) and RatSpeak (Burge & Brown, 2006) is revision of DRL for software engineering—ironically, one that restores some features of IBIS. QOC (MacLean et al., 1996) was devised entirely separately from IBIS yet strongly resembles DRL. While there are many deviations from Rittel's approach, few of them stray far from it.

Because of Rittel's influence, it is important to understand his ideas about the relationship between ideation and evaluation in design. Simply put, Rittel saw no need to intertwine them. This is reflected in the following statement in which he briefly describes a phased model of how designers attack a decision task:

> A designer first tries to develop a set of alternative courses of action, then to figure out their potential outcomes and their likelihood, and then to evaluate them, finally to decide in favor of one of them. (Rittel, 1966, p. 13)

In this statement, the ideation part corresponds to the phrase, "to develop a set of alternative courses of action." Evaluation corresponds to the phrase, "to figure out their potential outcomes and their likelihood, and then to evaluate them."

Rittel further states that he sees design as "an alternating sequence of two kinds of basic mental activities" (Rittel, 1966, p. 17), the first kind being ideation, which he describes as follows:

> Initially, a phase of "generating variety": the search for a set of relevant possibilities which might solve the problem at hand. (This is the process of developing ideas. It ends with a set of alternatives which contain at least one element.) (Rittel, 1966, p. 17)

The second kind consists of evaluation and selection, which he describes as follows:

> This is followed by a phase of “reducing variety”: the alternatives are evaluated for their feasibility and desirability, and a decision is made in favor of the most desirable, feasible alternative …. (Rittel, 1966, p. 17)

Because of these statements, from an article published 4 years before his first paper on IBIS, it should not be surprising that ideation and evaluation became incorporated into IBIS as separate processes: first, generation of positions, and then argumentation to evaluate the already-generated positions.

Rittel’s commitment to separating ideation and evaluation appears to be mirrored in other rationale approaches that, like IBIS, center on the evaluation of alternatives for design decisions. Thus, for example, none of these other approaches contains a type of link that could be used to indicate that an alternative was suggested by an evaluation of another alternative or that any alternative is an improvement on another alternative. The latter is important for the simple reason that the notion of improvement implies evaluation. In short, there is no sign of any connection between ideation and evaluation in any of the major approaches for modeling rationale about design decisions. Whether intentional or not, all of these approaches, like IBIS, give the impression that ideation and evaluation are in no way intertwined. This similarity might not be entirely due to Rittel’s influence, however, because many early theories of design (Alexander, 1964; Jones, 1970; Simon, 1969) exhibited a similar separation of ideation and judgment.

The Intertwining of Ideation and Evaluation in Design Discussion

MIKROPLIS (McCall, 1989; McCall et al., 1984) was the first PHI project to reveal the intertwining of ideation and evaluation in design discussion. Whereas its predecessor, the PROTOCOL project (McCall, 1979), had only a single designer, MIKROPLIS had a team of people involved in its design. Much of their discussion was documented. Because users of PROTOCOL had complained about not having control over the order in which it elicited rationale, MIKROPLIS was aimed at giving users control over display and input. This led to discussion of many issues of user interaction.

While MIKROPLIS team membership changed over its 5-year history, it included at various points people with solid knowledge of IBIS theory and applications. These included Wolfgang Schuler (Schuler & Smith, 1990), Barbara Lutes-Schaab (Lutes-Schaab, McCall, Schuler, & Werner, 1985), Harald Werner (Reuter & Werner, 1984), and Wolf Reuter (1983). Reuter, in particular, had a decade of IBIS experience when he joined the project.

As we documented discussions of the MIKROPLIS design team, differences emerged between our rationale and the adversarial rationale that Rittel (1980, pp. 7, 8) wrote about. Discussions in our team had a fundamentally different character from the clash of worldviews that IBIS was meant to deal with. Rather than being adversarial, our discussions were generally cooperative and collaborative. This is not to say that proposed ideas were not subjected to strong criticism, but the thrust of this criticism was constructive and there was a general openness to it by the group. This was also characteristic of teams in the later PHI projects.

One strong pattern that emerged in group discussion was that new ideas often arose out of evaluations of proposed ideas. While the response to criticism of (arguments against) a proposed idea (position) was sometimes to argue against it, often the response was to accept the criticism and propose a new or modified position. The adversarial argumentation that

Rittel wrote of featured an uncompromising defense of positions; the collaborative argumentation in our teams featured a general willingness to rethink positions. Where adversarial argumentation responded to criticism with rebuttal, our collaborative argumentation responded with creative ideation. Thus, while the former tended to separate ideation from evaluation, the latter intertwined them.

One of the forms that the intertwining commonly took was arguments that proposed better positions. Such arguments would typically identify an undesirable consequence of a proposed position and then immediately suggest a new or revised position that avoided that consequence. In fact, it seemed that the inclusion of the new position at the end of an argument was, in effect, a demonstration that its criticism was constructive. Thus, new positions were contained within arguments on old positions. Unfortunately, neither IBIS nor PHI recognized such combined utterances, because neither recognized intertwining. The following simple example, taken from a recent project, shows how a new position, indicated in italics, arose in an argument critical of an existing position:

> ISSUE: What programming technology should we use to create our 3D, Web-based, educational game for Mars exploration?
>
> POSITION: Flash CS4, using open-source Papervision3D for the 3D graphics.
>
> > ARGUMENT FOR: Flash has 98% browser penetration. The new version of ActionScript runs up to 10 times faster, and Papervision3D looks promising.
> >
> > ARGUMENT AGAINST: The problem is that existing approaches to Flash 3D, such as Papervision3D, cannot make use of the GPU. This will prevent us from creating the complex graphics we need for the game. *It would be better to use a technology that doesn't have these limitations—such as Java. That way we could use Java3D or JOGL for the 3D graphics.*

Intertwining took many other forms as well. Sometimes complex negotiations would take place between the person who proposed an idea and those who criticized it. These sometimes turned into mini design projects, each with the goal of devising ways of overcoming negative consequences of a proposed idea. Often these discussions were aimed at "rescuing" a flawed proposal by figuring out how to defuse its undesirable consequences.

It was not just criticism of an idea that produced new ideas. Some arguments approved of the basic idea behind a position but advocated taking it further. Such arguments often had the form, "If you're going to do that, why not go all the way and do X."

Design ideas often went through considerable evolution as a result of many iterations of critical argumentation and revision. These tended to be long, critical conversations among the team members. Sometimes there were creative breakthroughs during meetings. Sometimes discussions dead-ended but breakthroughs occurred between meetings.

The MIKROPLIS project showed me that critical conversations promoted creativity in design. Since then I have seen this pattern of creative argumentation in a wide variety of design discussions, both in PHI-based projects and in other projects that made no use of rationale methods. It seems that the hallmark of successful collaborative discourse is the revision of ideas based on feedback from argumentative evaluation.

In retrospect, it is clear that our documentation of such creative discussions was inadequate. When a new position on an issue was generated in response to an argument, we simply connected the argument to the position with an argument-for link. When an argument contained a new position, we would extract the position and record it separately as a position linked to a revised version of the argument that omitted the statement of the position. The problem with this approach was that inspection of the documented rationale revealed no evidence of the intertwined processes by which ideas had in fact been generated. While we were in theory using a process-oriented approach to rationale, in fact we were misrepresenting the processes involved. This was because PHI had unwittingly inherited IBIS's built-in separation of ideation from evaluation—in the form of link types that treated arguments only as *responses to* rather than *generators of* positions. As a consequence, the impression that our documented rationale gave was that positions were generated intuitively and immediately as direct responses to stated issues and that the only role of arguments was to evaluate previously generated positions. There was no real indication that argumentation had played a crucial role in ideation.

The intertwining of ideation and evaluation in discussions among designers turned out to be merely one of a number of ways in which such intertwining promotes creative design. Discovery of other ways was made possible by a profound change in our understanding of the nature of design. The change was from Rittel's (1966) view of *design as planning* to Schön's (1983) view of *design as situated cognition.* This change in perspective solved major problems we encountered in creating the PHIDIAS software (McCall, Bennett et al., 1990; McCall et al. 1994; McCall, Ostwald et al., 1990). The following section begins by looking at the differences between these two views and their implications for the role of rationale in design. It then describes the problems we encountered and explains how these led us to adopt Schön's point of view.

DESIGN: FROM PLANNING TO SITUATED COGNITION

The term *situated cognition* is used with a number of different meanings. It is used here in the behavioral sense of "a transactional process of transforming and interpreting materials in the world" (Clancey, 1997, p. 23). It is in this sense of the term that we can say that both Suchman (1987) and Schön (1983) have written about situated cognition.

Two Views of Design

There are two fundamentally different views of design: as planning and as situated cognition. The former sees design as reasoning that precedes action, the latter as reasoning intertwined with and informed by action. The implication of the former is that design rationale is the documentation of the thinking and discussion of designers preparing for the actions of implementation and use. The implication of the latter is that design rationale is the documentation not only of planning by designers but also of (a) the feedback from actions that challenges design decisions, and (b) the creative thinking of designers in response to such challenges. The situated cognition viewpoint thus sees design as an intertwining of ideation and action-based evaluation. To date, the literature on all rationale methods except SCA (Carroll & Rosson, 1992) has dealt exclusively with rationale as planning.

Rittel's View of Design as Planning

Rittel clearly viewed design as planning, not as situated cognition. He declared, "Designing means thinking before acting," and he described design as a process of devising a plan (Rittel, 1966, p. 13). In fact, Rittel used the terms *designing* and *planning* interchangeably and saw design as a phase that is completed before feedback from action is available:

> The distinctive property of designing lies in the—frequently very long—interval between the design process (i.e., the construction of the plan) and the "feedbacks"—the effects of the execution of the plan. (Rittel, 1966, p. 14)

This lack of feedback implies that designers cannot test their ideas in real-world settings:

> …there is not the opportunity to approach solutions by trial and error; there is nothing like experimentation with real situations. (Rittel, 1966, p. 14)

Therefore, designers must rely solely on their imaginations to determine the consequences of their ideas:

> As a result of these characteristics, the designer operates in a world of imagination. He has to anticipate, to guess, to judge what *might* happen if a certain contemplated action will be carried out. (Rittel, 1966, p. 14)

The picture that Rittel paints is of design as speculative reasoning aimed at the production of a plan. In other words, Rittel's notion of design as purely a process of argumentation is a direct consequence of his view of design as planning.

Schön's View of Design as Situated Cognition

Schön's (1983) theory of design as reflective practice provides a fundamentally different view. Schön saw design as an alternation between an intuitive process he called *knowing-in-action* and a type of reasoning he called *reflection-in-action*. With knowing-in-action, the designer is engaged in performing a task without conscious reflection. With reflection-in-action, the designer stops acting and instead reflects on how to perform the task at hand. A designer cannot simultaneously engage in both knowing-in-action and reflection-in-action.

Knowing-in-action proceeds until a breakdown occurs. This happens when intuitive performance produces unexpected feedback from the situation at hand. In other words, there is a breakdown in the designer's expectations. Schön describes this by saying "the situation talks back" (1983, p. 131). A breakdown results when something goes wrong, but it also results when something unexpectedly good happens. Breakdowns occur when intuitive action produces either problems or opportunities that intuition cannot deal with. At this point, the designer switches to reflection-in-action to reason about how to deal with the unexpected results. If and when reflection is successful, the designer resumes knowing-in-action.

Reflective practice is repeated alternation between knowing-in-action and reflection-in-action. Schön describes the designer as engaging in an ongoing "conversation with the situation" (1983, p. 76). This is a view of design as a type of situated cognition, in that it sees design reasoning as intertwined with and informed by action.

Reflective practice models design as an intertwining of ideation and evaluation. When the situation "talks back," the "backtalk" is evaluative feedback that reveals consequences of the actions taken. The purpose of the resulting reflection-in-action is to devise new ideas for how to act; in other words, the purpose of reflective practice is ideation. Putting new ideas into action with knowing-in-action is how the designer resumes "talking to the situation." This eventually results in more backtalk that again triggers reflection that results in further ideation—and so forth.

Implications of the Two Views

To Rittel (1966), design is nothing but explicit reasoning, that is, argumentation; to Schön (1983), design is both explicit reasoning and intuitive action. Rittel's view implies that rationale can represent all design processes; Schön's view implies that it cannot. For Rittel design is reasoning in preparation for action in an external environment; for Schön design is reasoning triggered and motivated by action in an external environment. Rittel portrays design as a conversation among designers, Schön as a "conversation" between designers and a situation. As my colleagues, students, and I implemented Rittel's view of design in software, experiences in implementing and using prototypes ultimately led to rejecting Rittel's view of design as planning, in favor of Schön's view of design as situated cognition.

From Viewing Design as Planning to Viewing It as Situated Cognition

Limitations of MIKROPLIS

Towards the end of the MIKROPLIS project (McCall, 1989; McCall et al., 1984) in 1984-1985, user testing revealed two major shortcomings. One was that it did not solve the *rationale capture problem*, that is, the reluctance of designers to document their rationale. We originally thought this problem resulted from the copious and tedious secretarial work involved in documenting rationale. MIKROPLIS successfully eliminated most such work. Unfortunately, this merely revealed the enormity of the cognitive overhead in rationale capture. The other shortcoming was that when MIKROPLIS was used to design buildings, its users created rationale that failed to deal with decisions about the forms of the buildings. Without representing and editing these forms graphically, there was apparently no way for users to make decisions about them.

Ideas for PHIDIAS

In 1985 my colleagues and I began designing PHIDIAS (PHI-based Design Intelligence Augmentation System; McCall, Bennett et al., 1990; McCall et al., 1994; McCall, Ostwald et al., 1990) by extending MIKROPLIS. The new functionality supported design ideas aimed at overcoming the two major limitations of MIKROPLIS.

The first idea was for PHIDIAS to use *domain-oriented issue bases* to mitigate the capture problem (McCall, Bennett et al., 1990). Such an issue base is a collection of the issues, positions and arguments that commonly occur in a design domain—for instance, the design of a given type of building. The main goal was to reduce the work of creating a project issue base by "priming the pump" with a generic issue base for a domain—for example, design of lunar habitats—that could be tailored to a specific project, such as the design of a specific lunar habitat for four astronauts. In addition, to alleviate the capture problem, domain-oriented issues bases could help designers by providing useful design information.

The second idea was to have PHIDIAS enable decision making about building forms by adding functionality for CAD graphics. We created this functionality but failed to foresee that attempting to incorporate form-making into PHI would lead us to abandon Rittel's (1966) view of design as nothing but argumentative planning.

Unexpected Problems in Creating PHIDIAS

We had no difficulty creating domain-oriented issue bases and integrating them into PHIDIAS, and these issue bases greatly reduced the work of creating a project-specific issue base. Unfortunately, they were not effective in providing student designers with useful information. Since students did not know what information was and was not in the system, they did not know whether searching for information would pay off. As a consequence, they often searched for information that was not in the system, got frustrated and then stopped searching for any information. This was especially unfortunate, because the system had information that could have saved them from many of the mistakes they made in design.

We also successfully implemented basic CAD functionality, but we ran into profound difficulties in attempting to integrate CAD graphic editing into the interface for rationale creation. The problem was conceptual, not technical. It resulted from apparent conflicts between the activities of form making and verbal reasoning. To solve this problem we attempted to study how student designers reasoned about form making. This attempt was repeatedly frustrated. Asking students to document their own reasoning while they drew building forms produced little or no plausible rationale. Sending others in to document the rationale of designers also produced no significant results. They would explain their rationale right up to the moment they started drawing, at which point they would not talk about what they were doing. We did succeed in getting one talented student to record a think-aloud protocol about his form making over six weeks. Unfortunately, he felt that reasoning aloud had interfered with his ability to design; so he redid the entire design over a weekend without recording any rationale. So, while we made excellent progress on implementing CAD functionality in PHIDIAS, we made no real progress integrating form-making into rationale. This prevented us from completing the PHIDIAS interface.

CRACK

The solutions to the problems that PHIDIAS had encountered became obvious when I saw a demo of the CRACK (CRitiquing Approach to Cooperative Kitchen design) system created by Anders Morch under the supervision of Gerhard Fischer (Fischer & Morch, 1988). Fischer had been investigating the use of domain-oriented construction kits for design (Fischer, 1987; Fischer & Lemke 1988). A *construction kit* is a set of graphical building blocks that can be dragged and dropped into a workspace. He found that while such kits greatly facilitated the creation of designs, these designs were often functionally flawed. He concluded that construction kits had to be supplemented with some way of avoiding design mistakes. For this purpose, Fischer proposed using what he termed *knowledge-based critics* to guide design with construction kits. Morch's master's thesis implemented Fischer's ideas in the kitchen design domain, in which Morch had previously worked.

CRACK featured a CAD graphics editor for creating kitchen floor plans using a kitchen construction kit featuring such domain-level building blocks as walls, windows, doors, counters, stoves and sinks. This kit provided a direct and intuitive way for users to construct kitchen floor plans. Since each building block had an assigned domain-level meaning, knowledge-based critics could determine whether a constructed floor plan satisfied or violated rules of kitchen design. If rules were violated during the construction of a layout, critiquing messages popped up on the screen to tell the user which rules had been broken. For example, if a stove were placed where pans could be hit by an opening door, then the designer got a message saying that the stove should not be located next to a door.

CRACK was intended not to enforce its rules, for example, as an expert system would, but rather to empower the user to decide whether to accept or reject them. Unfortunately, it was often difficult for users to decide whether to break rules. I suggested that this was because such decisions required knowledge of the rationale underlying the rules. I therefore proposed the addition of a hypertext subsystem containing rationale for the rules of kitchen design in the form of a PHI-based, domain-oriented issue base. The decision was made to create a successor to CRACK that did just that. The successor was called JANUS (McCall, Fischer et al., 1990; Fischer et al., 1996), after the Roman god with two faces, because it had both a form-construction interface and an argumentation interface.

JANUS and PHIDIAS

From the perspective of the PHIDIAS project, the notion of coupling PHI hypertext to a CRACK-type interface was a revelation. It offered in one stroke a solution to two problems plaguing the PHIDIAS project. First of all, it showed how users could be alerted to the existence of useful information in a PHI issue base while they worked on a design problem. Secondly, it suggested that rather than attempting to integrate the editing of CAD graphics into the editing of a PHI hyperdocument, the solution was to have two separate interfaces—a form construction interface and an argumentation interface—and switch between these using critics. So while Morch and others constructed JANUS, my programming team constructed a similar coupling of CAD form-construction and argumentation in PHIDIAS. User testing showed that both systems successfully supported use of rationale to inform construction of floor plans.

From Argumentative Planning to Reflective Practice

It was not immediately clear that the new systems challenged Rittel’s (1966) theory of design as argumentative planning. Awareness of that challenge first surfaced when Morch wrote a working paper proposing that JANUS supported two different modes of designing: *constructive design* and *argumentative design*. At first, I balked at that distinction, which was heresy from the Rittelian perspective. But the failed attempts to integrate form-construction into PHI ultimately led me to abandon the notion that form making is purely an argumentative process. Morch’s names for the two design modes were therefore put in the title of our first paper on the new type of system (Fischer, McCall, & Morch, 1989).

Not long after this it became clear that the failures in integrating form-construction into PHI and the success of our dual-interface approach both fit Schön’s (1983) ideas about reflective practice. Constructive design with construction kits corresponded to knowing-in-action, critiquing corresponded to breakdowns, and argumentative design with PHI hypermedia corresponded to reflection-in-action. So we came to see JANUS and PHIDIAS as unintended demonstrations of the correctness of Schön's theory of design—a theory fundamentally incompatible with Rittel's.

While at first Schön's theory was merely a retrospective explanation for the success of our systems, later it became the central driving principle behind the design of PHIDIAS and HyperSketch (McCall et al., 1997; McCall et al., 2001). PHIDIAS implemented a variety of additional ways in which the existence of breakdowns could be detected by the system (McCall & Johnson, 1997) or volunteered by users of the system (McCall, 1998). An example of the former is shown in Figures 1 and 2. Here, knowledge-based agents are created by system users

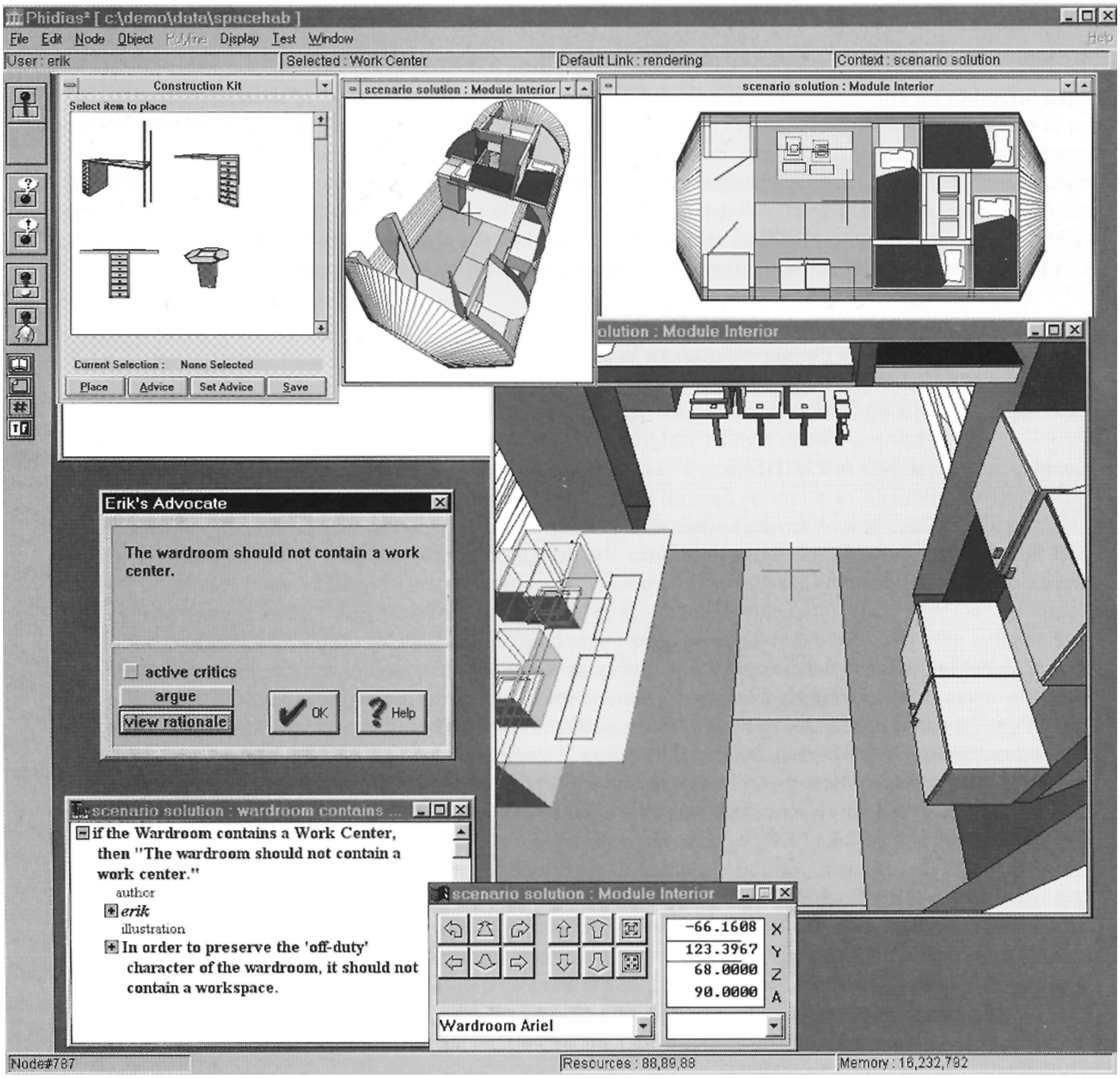

Figure 1. In PHIDIAS, designers working on the same project can create knowledge-based agents called *advocates*, which are critics that lobby for design principles that they believe in. In this figure, Patrick violated an advocate created by Erik, and thus received a critiquing message. Patrick has opted to view Erik's rationale for the advocate.

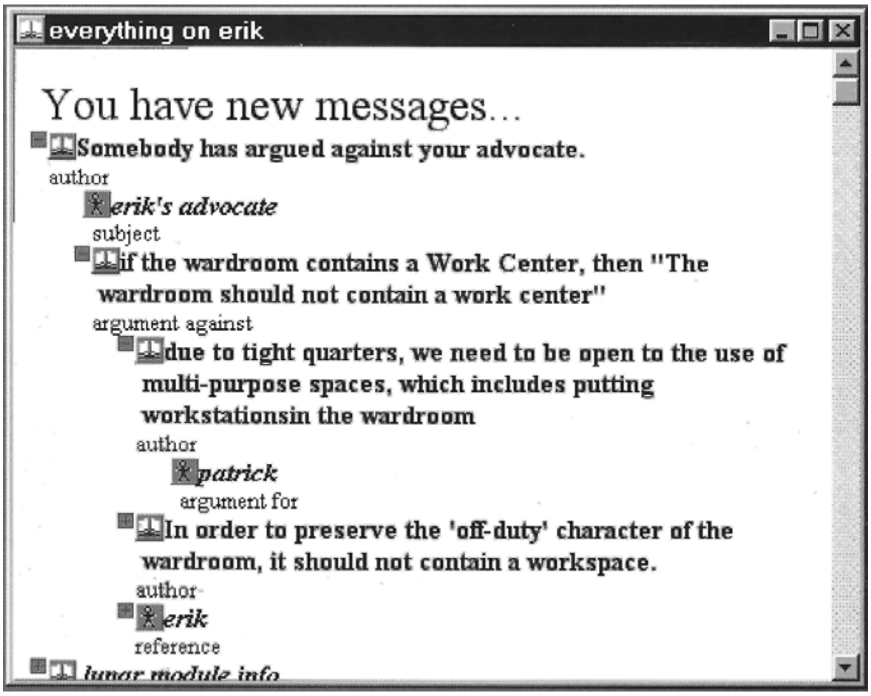

Figure 2. Patrick argued against the rationale for Erik's advocate, so Erik was notified and sent the argument. He was then given the option to participate in an issue-based discussion with Patrick about whether the advocate should be violated.

as advocates of their opinions in a collaborative design environment. Then when other designers use the system, they are alerted if they construct design features that conflict with any of these advocate agents, as in Figure 1. They also have the opportunity to view and argue with the rationale for the advocate. If a designer argues against the advocate agent, the designer who created it is alerted to this fact and offered the chance to discuss this situation with the designer who disagreed with the advocate. The resulting online discussion can be recorded in the form of issue-based argumentation, as in Figure 2.

Other research driven by Schön's ideas inquired into what sorts of interfaces were needed for intuitive knowing-in-action. HyperSketch (McCall et al., 1997; McCall et al., 2001) explored intuitive form construction through computer-supported sketching. This was in response to architecture students who complained that construction kits inhibited their intuitive exploration of building forms.

Schön's (1983) theory of design as situated cognition shows another way in which ideation and evaluation are intertwined. Previously we saw this only in argumentative design discussion; now we see it when argumentation is coupled with action. Furthermore, this intertwining can be seen as promoting creative ideation. When a critic reveals that something is wrong with the design, the designer rethinks a design decision and devises new solution ideas.

It should be noted that the criticisms here of Rittel's (1966) ideas about feedback and argumentative planning in no way imply a rejection of his theories in toto. Instead, this criticism is meant as a necessary corrective if design rationale, the field that Rittel pioneered, is to be successful. Nor does this criticism imply an unqualified endorsement of Schön. In fact, it is argued below that Schön's notions of reflective practice are too limited to account for several important ways in which creative design involves situated cognition. Accounting for these additional ways involves extending Schön's notions by bringing into the picture Rittel's ideas about collaborative and participatory design.

Software Design as Situated Cognition

How Our Software Design Experiences Differ from Rittel's Description of Design

Our experiences of software design and Rittel's description of design differ in the role of feedback from implementation and from use in informing design. For Rittel (1966), a distinguishing feature of all design is that it cannot be informed by such feedback. Yet our experiences provide numerous counterexamples to this claim.

Was Rittel completely wrong? Or was he simply referring to a different kind of design than we engaged in? His arguments against learning from feedback suggest the latter. Consider the following statement from the article that he wrote with Webber about wicked problems:

> One cannot build a freeway to see how it works, and then easily correct it after unsatisfactory performance. Large public-works are effectively irreversible, and the consequences they generate have long half-lives. Many people's lives will have been irreversibly influenced, and large amounts of money will have been spent—another irreversible act. (Rittel & Webber, 1973, p. 163)

Rittel (1966) claims that his theory applies to all types of design, yet the above-stated argument depends on properties found in some types of design but not others. In particular, the argument applies to large-scale design projects with large costs and large consequences. The specific example used, a freeway, represents an infrastructural level of design, meaning a very low-level of structure—*infra* meaning *below* in Latin (Hoad, 1996). Designing such a large-scale physical artifact might indeed be, as Rittel claimed, a one-shot operation in which feedback from implementation and use plays no role. Nevertheless, this does not imply that it plays no role in other levels of design.

If one substitutes a "high-level" artifact, such as a piece of furniture, into the Rittel-Webber argument, the credibility of that argument collapses. For example, an industrial designer can in fact build a chair to see how it works. If its performance is unsatisfactory, for example, if it is uncomfortable or structurally unsound, the designer can easily correct the bad design. Furthermore, its consequences are unlikely to have long half-lives. If any consequences are irreversible, they are unlikely to be severe and can be restricted to a small group of users who test the chair before it is made available to the public. The costs of redesigning and re-implementing the chair are likely to be small compared to profits made from selling thousands of well-designed chairs. In other words, feedback from implementation and use can play a significant role in the design of chairs and other high-level artifacts.

Difference in level, however, cannot explain all the differences between the design of software and the design of the sorts of low-level, large-scale artifacts that Rittel focused on. The design of new buildings and freeways generally might not involve learning from feedback about implementation and use, yet it is hard to find any level of software design that cannot learn from such feedback. The implementation and use of working prototypes and early versions play crucial roles in shaping the design of new operating systems, new browsers, and new rich Internet applications—three very different levels of software design. There are no comparable roles for usable prototypes or early versions of buildings or freeways.

Another limitation of Rittel's theory is that it ignores the redesign of artifacts. It is a truism that buildings and cities evolve over decades through many episodes of redesign (Brand, 1994). Such redesign is often informed by implementation and use. Successful software at all levels also goes through many episodes of redesign that are informed by feedback from implementation and use of previously released versions.

Our PHI-based software projects contained many cases where the design was changed in the middle of being implemented. The design of the PHIDIAS interface between PHI rationale and CAD graphic construction of form is the most conspicuous example of this. Current work by software engineers on iterative and incremental design also has this character. To be sure, software engineering for years militated against changes in decisions about requirements and design, because they were so costly. But in recent years, software engineers have become increasingly open to such changes.

How Feedback from Implementation Led to New Design Ideas

Over the history of the MIKROPLIS and PHIDIAS projects, a single type of phenomenon dominated the generation of design ideas: the repeated discovery of new affordances that arose as unplanned side-effects of implementing required design features. These discoveries influenced the design of the software in two ways. One was in suggesting ideas for the architecture of the system; the other was in leading us to re-evaluate and revise the requirements for the system.

Over the 18 years of the projects, the system architecture that emerged was a radically simple and integrated hyperbase management system (HBMS) with an operator-algebraic, functional language called PHIQL (PHIDIAS Hypermedia Inference and Query Language). This HBMS was coupled with subsystems for display of a wide range of multimedia data, including text, vector graphics, images, and video, together with subsystems for editing text and vector graphics. We came to call this a *hyperCAD architecture*.

The way in which ideas for PHIDIAS' architectural features emerged shows how feedback from implementation can shape the design of system architecture. For example, when we decided we needed to represent and edit vector graphics, the obvious approach was to buy or build a separate 3D graphics system and add it to the architecture. I started to do just that, but my knowledge of the implementation details of the graph-handling functionality of MIKROPLIS led to the insight that it could be used for scene graphs as well as textual networks. Once this new affordance of the MIKROPLIS system was discovered, it became clear that utilizing this affordance would make it possible to link any text to any vector graphic object in the system—thus enabling PHI-based discussion of all graphical objects and configurations. In other words, knowledge of implementation details led to discovery of an unplanned affordance of an existing system, which in turn led to the insight that exploiting this affordance served the goals of the larger project in ways that had not been foreseen. Here, both knowledge of implementation details and the affordances of those details provided feedback from implementation that led to the generation of design ideas.

As it turned out, once we had designed a system architecture that implemented scene graphs in the HBMS, additional unplanned affordances emerged as direct consequences of this decision. For example, since PHIQL could now construct arbitrary displays of linked text and vector graphics, it became trivial to construct in PHIDIAS the catalogues of completed designs that existed in JANUS—something which had previously been of interest to us but too far down on our priorities to appear in our system requirements. Using PHIQL and scene graphs also made it possible and easy to create a catalogue of reusable subassemblies, something that did not exist in JANUS. Though we had never before thought of creating such a catalogue, we quickly realized it would be a very useful feature for a designer. So we added this and a catalogue of completed designs to our list of system requirements.

The integrated hyperCAD architecture emerged as a consequence of repeated discovery of unplanned affordances. Over the history of the PHIDIAS project, we frequently found that desired new functionality could best be implemented by exploiting affordances of the existing system rather than by adding new code that implemented the functionality from scratch. It was a more efficient use of our time and knowledge, and it tended in turn to produce still more affordances. We kept discovering that we were able to generate valuable new functionality almost for free. We began talking not only of what we wanted the system to be but also of what the system itself "wanted to be"—a metaphorical way of referring to new affordances produced as side effects of implementation. This sort of metaphor, which anthropomorphizes the artifact being designed and treats it is if it were a partner in discussion, has been used by a number of well-known (building) architects, most famously Louis Kahn (Twombly, 2003). It is closely related to Schön's (1983) reference to the situation "talking back."

There are dozens of other examples of how feedback from implementation shaped the architecture of PHIDIAS and led to the addition of new system requirements, far more than there is room here to describe. While this sort of feedback was the most frequent source of new design ideas, many of the more profound ideas emerged in feedback about system use.

How Feedback from Use Led to New Design Ideas

Our PHI-based software projects contained a number of important cases where feedback from use led to new design ideas. These included the following:

- Users of PROTOCOL complained about lack of control of the order in which issues were dealt with. This led to the design of MIKROPLIS as a system where users had complete control over the order of rationale input and display.
- Use of MIKROPLIS indicated that it had not solved the rationale capture problem. This led to the use of domain-oriented issue bases in JANUS and PHIDIAS
- Tests of MIKROPLIS users attempting building design determined that they failed to deal with decisions about the building form. This led to the inclusion of CAD graphics in the redesigned version of MIKROPLIS that came to be called PHIDIAS.
- Tests of users of domain-oriented issue bases in PHIDIAS showed that they had difficulty finding useful information in these issue bases. This contributed to the use of critics in PHIDIAS to identify and retrieve useful issue-based information.

There were numerous other examples during all of our software projects. One early example of this happened in 1982 with the very first MIKROPLIS prototype. MIKROPLIS had originally been designed as a query-based retrieval system, but tests with users revealed this approach to be inadequate. In particular, almost all users of the system kept pointing to individual texts displayed on the screen and saying something like, “How do I find the information about this?” We repeatedly showed users how to use queries to find such information, but they continued to have difficulties. I finally got the idea of enabling them to place the cursor on the desired text and instruct the computer to traverse a link associated with that text—something roughly comparable to clicking on a link in a Web page. At the time we had no graphical user interfaces, so I had the user move the cursor to the text with the arrow keys and then press the Enter key to signal the computer to perform link traversal. Once we had implemented this feature, all users rapidly adopted this as the favored mode of interacting with the system. This was the first inkling we had of what was to become the future of interacting with hyperdocuments: clicking on links. The crucial point is that without feedback from users, we would not have come to this idea on our own.

Another example came from having design students use PHIDIAS to construct building forms. Many complained that construction kits were too restrictive and not sufficiently intuitive, especially since using construction kits in realistic projects requires browsing through many menus and panels of information to find the objects the designer wants to place in the scene. In response to these complaints, we created functionality for pen-based drawing and creating hyperdocuments of linked drawings (McCall et al., 1997).

Extending Schön’s View to Account for Feedback from Implementation and Use

Schön’s (1983) theory of reflective practice does not cover the sort of situated cognition in which feedback from implementation and use challenges a designer to revise the design of software. This is because Schön’s theory only deals with action in the sense of the purely intuitive process he calls knowing-in-action. According to reflective practice the designer is in this process when feedback occurs that produces a breakdown and a switch to reflection-in-action. There are a number of features of this account that do not fit crucial cases of situated cognition in software design. First of all, actions do not have to be intuitive to produce feedback that leads designers to rethink the design of the system. The actions of implementation and use may well involve complex combinations of knowing-in-action and reflection-in-action. In any case, the mental states of the implementers or the users are not relevant here. Nor is it relevant what mental state the designer is in when feedback arrives; the designer could be acting, reflecting, just browsing the web, or eating a sandwich. The only thing that matters is that the feedback produces surprises and that these constitute a breakdown of the designer’s expectations about the consequences of design ideas—either in the form of unexpected problems or unexpected opportunities. In such cases, the breakdowns will challenge the designer to rethink the design of the system and come up with new ideas that solve the problems or exploit the opportunities.

If we simplify Schön’s model of reflective practice, we can make it general enough to cover all the cases. Rather than talking of knowing-in-action and reflection-in-action, we can talk simply of action and reflection. We can then say that in all cases of design as situated cognition *action* produces *feedback* that results in a *breakdown of expectations*, and that this promotes *reflection* aimed at the generation of new design ideas (ideation) to deal with the source of the feedback.

We can further modify Schön's model to account for critical conversations in argumentation among designers. Here a designer proposes an idea to a group of participants and gets feedback from them in the form of critiques of the idea. These critiques are only based on *speculations about the consequences* of the proposed design idea but are still capable of causing a breakdown in the expectations of the designer who proposed it. Such a breakdown then leads that designer—and others participating in the discussion—to reflect on how to revise the proposed idea or to devise a new idea. Here we have feedback, breakdown, reflection and the generation of new ideas without any actions of any kind. And yet this type of critical conversation bears a clear resemblance to reflective practice.

IMPLICATIONS FOR RATIONALE THAT PROMOTES CREATIVITY IN SOFTWARE DESIGN

Critical Conversations That Promote Creativity in Software Design

This paper has identified three processes in which the intertwining of ideation and evaluation promotes creativity in software design. When design ideas are evaluated, this evaluation can produce feedback that challenges designers to generate new ideas that improve the quality of the design. The three processes are as follows:

- The intertwining of ideation and evaluative argumentation in design discussion,
- The intertwining of the action of software implementation with reflection on the feedback from implementation, and
- The intertwining of the action of use with reflection on the feedback from use.

The first process involves purely argumentative conversation. The second and third involve types of situated cognition that do not precisely fit Schön's (1983) model of reflective practice but which, nevertheless, can be described as designers' conversations with situations.

Rittel's (1972) idea about the importance of involving implementers and users in participatory collaboration with designers comes into play in creative design of software—but in a way that Rittel did not anticipate. While he envisioned participation as taking the form of argumentative discussion, understanding design as situated cognition leads to us to extend this participation to the provision of feedback by implementers and users about the actual consequences of implementation and use of the software being designed.

What Rationale Needs to Do to Support the Critical Conversations

Critical conversations are rationale processes that help designers to be more creative. Since they are processes, a rationale approach that recognizes and promotes them is by definition process oriented. Since these processes are for the purpose of improving design, the rationale approach is by definition prescriptive. To support the evaluation that promotes ideation, a rationale approach must represent how evaluations promote ideation. It must represent the evaluations and the ideas they lead to. It must also provide links that show which ideas were generated in response to which evaluations. Any approach to rationale that aims to support the full range of design creativity must encourage and document the generation of evaluative feedback from (a) design discussion, (b) implementation, and (c) use. To do this, it must capture rationale containing this feedback from designers, implementers, and users. It must also support the communication of this rationale to designers. If feedback from action conflicts with feedback from the pure argumentation, it is likely that the former should trump the latter—since evidence and experience trump speculation. Because of this, documented feedback should always indicate whether its source is argumentative discussion, implementation, or use. In addition, the author of the feedback should always be indicated so that follow-up conversations can be established.

Decision-centric approaches to rationale, such as IBIS (Kunz & Rittel, 1970) and PHI (McCall, 1979, 1986, 1991), are unlikely to be sufficient for collecting feedback from implementation and use, because such methods only model the design process as a coherent whole. A rationale method, such as SCA (Carroll & Rosson, 1992), is highly preferable for collecting feedback from use, because it models use processes as coherent wholes. It can thus systematically enumerate use situations and the feedback resulting from them in a way that decision-centric approaches simply cannot match. However, what needs to be done is to more closely integrate approaches like SCA with decision-centric rationale.

An open question is how the feedback from implementation should be collected. Should it be treated as a decision-centric rationale process or should a special method be developed? Whatever is done needs to be capable of systematically enumerating the feedback from implementation and it needs to be integrated with the decision-centric rationale for design.

CONCLUSIONS AND FUTURE WORK

Methods for rationale elicitation and documentation can promote creativity in software design by recognizing and promoting feedback-driven critical conversations in software projects. Critical conversations are rationale discussions in which the ideation, meaning the generation of design ideas, is intertwined with evaluation of those ideas in the sense that feedback from evaluation challenges designers to devise new ideas. There are three main types of such conversations:

- purely argumentative design discussions where designers get feedback from the speculative reasoning of other design participants,
- discussions where designers get feedback from implementers about the consequences of implementation of the software being designed, and
- discussions where designers get feedback from users about the consequences of use of the software being designed.

The first of these corresponds to Rittel's (1972) view of design as purely a process of argumentation, but it goes beyond the argumentative discussions that IBIS supports. The other two view design as a process in which argumentation is situated in the context of action that motivates and informs it. To maximize the potential of rationale to promote creative software design, we must move beyond Rittel's view of design rationale as pure argumentation and see it also as situated argumentation.

Considerable work needs to be done in revising approaches to rationale to support the critical conversations described above. Decision-centric rationale methods, such as IBIS and PHI, have to be revised to represent the intertwining of argumentative evaluation and idea generation. The changes made to represent this intertwining in pure argumentation will provide a basis for further changes needed to support situated argumentation in the context of implementation and use. In addition to modifying decision-centric approaches, usage-centric approaches to rationale such as SCA (Carroll & Rosson, 1992) need to be utilized as ways of systematically obtaining feedback from use situations. Research also needs to be done to determine how best to support the capture and communication of feedback from implementation. Finally, work needs to be done on integrating these various approaches to rationale.

A crucial lesson of the JANUS (McCall, Fischer et al., 1990; Fischer et al., 1996) and PHIDIAS (McCall, Bennett et al., 1990; McCall et al., 1994; McCall, Ostwald et al., 1990) projects is that both delivery and capture of rationale need to be integrated into the software that supports action. This means that rationale functionality should be integrated into the tools for modeling and implementing software. It also suggests that rationale capture may need to be integrated into the software artifacts being designed to enable feedback from actual use.

REFERENCES

Alexander, C. (1964). *Notes on the synthesis of form*. Cambridge, MA, USA: Harvard University Press.

Brand, S. (1994). *How buildings learn: What happens after they're built*. London: Phoenix Illustrated.

Burge, J. E., & Brown, D. C. (2006). Rationale-based support for software maintenance. In A. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 273–296). Heidelberg, Germany: Springer Verlag.

Carroll, J. M., & Rosson, M. B. (1992). Getting around the task-artifact cycle: How to make claims and design by scenario. *ACM Transactions on Information Systems*, *10*, 181–212.

Clancey, W. J. (1997). *Situated cognition: Human knowledge and computer representations.* Cambridge, UK: Cambridge University Press.

Conklin, E. J., & Begeman, M. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *ACM Transactions on Information Systems, 6*, 303–331.

Conklin, E. J., & Burgess-Yakemovic, K. C. (1996). A process-oriented to design rationale. In T. P. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 393–427). Mahwah, NJ, USA: Lawrence Erlbaum and Associates.

de Bono, E. (1973). *Lateral thinking: Creativity step by step*. New York: Harper Colophon.

Dutoit, A. H., McCall, R., Mistrik, I., & Paech, B. (2006). Rationale management in software engineering: Concepts and techniques. In A. H. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 1–43). Heidelberg, Germany: Springer Verlag.

Fischer, G. (1987). An object-oriented construction and tool kit for human-computer communication. *Computer Graphics*, *21*, 105–109.

Fischer, G., & Lemke, A. (1988). Construction kits and design environments: Steps toward human problem-domain communication. *Human-Computer Interaction*, *3*, 179–222.

Fischer, G., Lemke, A. C., McCall, R., & Morch, A. I. (1996). Making argumentation serve design. In T.P. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 267– 321). Mahwah, NJ, USA: Lawrence Erlbaum Associates.

Fischer, G., & Morch, A. (1988). CRACK: A critiquing approach to cooperative kitchen design. In *Proceedings of the International Conference on Intelligent Tutoring Systems* (pp. 176–185). New York: ACM.

Fischer, G., McCall, R., & Morch, A. (1989). Design environments for constructive and argumentative design. In K. Bice & C. Lewis (Eds.), *Proceedings of the SIGCHI Conference on Human Factors in Computing System* (CHI'89; pp. 269–275). New York: ACM.

Hoad, T. F. (Ed.). (1996). *Oxford concise dictionary of English etymology*. New York: Oxford University Press.

Jones, J. C. (1970). *Design methods: Seeds of human futures*. New York: Wiley-Interscience.

Kunz, W., & Rittel, H. W. J. (1970). Issues as elements of information systems (Working Paper 131). Institute for Urban and Regional Development, University of California, Berkeley. Available at http://iurd.berkeley.edu/sites/default/files/wp/131.pdf

Lawson, B. (2005). *How designers think: The design process demystified* (4th ed.). Burlington, MA, USA: Architectural Press (Elsevier).

Lee, J. (1991). Extending the Potts and Bruns model for recording design rationale. In *Proceedings of the 13th International Conference on Software Engineering* (ICSE 13; pp. 114–125). New York: ACM.

Lee, J., & Lai, K.-Y. (1996). What's in design rationale? In T. P. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 21–51). Mahwah, NJ, USA: Lawrence Erlbaum Associates.

Lutes-Schaab, B., McCall, R., Schuler, W., & Werner, H., (1985). *MICROPLIS – ein Textbank-Management-system* [MICROPLIS: A textbase management system; Final report]. Heidelberg, Germany: Gesellshaft für Information und Dokumentation (GID), Sektion für Systementwicklung (SfS), [Society for Information and Documentation, Section for Systems Development].

MacLean, A., Young, R. M., Belotti, V. M. E., & Moran, T. P. (1996). Questions, options and criteria. In T. P. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 53–106). Mahwah, NJ, USA: Lawrence Erlbaum and Associates.

McCall, R. (1979). *On the structure and use of issue-systems in design*. Unpublished doctoral dissertation, University of California, Berkeley. (Available from University Microfilms International, Ann Arbor Michigan)

McCall, R. (1986). Issue-serve systems: A descriptive theory of design. *Design Methods and Theories*, *20*, 443–458.

McCall, R. (1989). MIKROPLIS: A Hypertext system for design, *Design Studies*, *10*, 228–238.

McCall, R. (1991). PHI: A conceptual foundation for design hypermedia. *Design Studies*, *12*, 30–41.

McCall, R. (1998). World wide presentation and critiquing of design proposals with the Web PHIDIAS System. In S. van Wyck & S. Seebohm (Eds.), *Digital Design Studios: Do Computers Make a Difference? Proceedings of the 1998 Conference of the Association of Computer Aided Design in Architecture* (ACADIA 98; pp. 254–265). Quebec City, Quebec, Canada: ACADIA.

McCall, R., Bennett, P., & Johnson, E. (1994). An overview of the PHIDIAS hyperCAD system. In A.C. Harfmann & M. Fraser (Eds.), *Proceedings of the 1994 Conference of the Association for Computer Aided Design in Architecture* (ACADIA '94; pp. 63–76). St. Louis, MO, USA: ACADIA.

McCall, R., Bennett, P., d'Oronzio, P., Ostwald, J. L., Shipman, F. M., & Wallace, N. F. (1990). PHIDIAS: A PHI-based design environment integrating CAD graphics into dynamic hypertext. In R. A. Streitz & J. Andre (Eds.), *Hypertext: Concepts, systems, and applications* (pp. 152–165). Cambridge, UK: Cambridge University Press.

McCall, R., Fischer, G., & Morch, A. (1990). Supporting reflection-in-action in the JANUS design environment. In M. McCullough, W. Mitchell, & P. Purcell (Eds.), *The electronic design studio: Architectural education in the computer era* (pp. 247–259). Cambridge, MA, USA: MIT Press.

McCall, R., & Johnson, E. (1997). Using argumentative agents to catalyze and support collaboration in design. *Automation in Construction*, *6*, 299–309.

McCall, R., Johnson, E., & Smith, M. (1997). HyperSketching: Design as creating a graphical hyperdocument. In R. Junge (Ed.), *Proceedings of the 7th International Conference of Computer Aided Architectural Design Futures* (CAAD Futures 1997; pp. 849–854). Dordrecht, Netherlands: Kluwer Academic Publishers.

McCall, R., Lutes-Schaab, B., & Schuler, W. (1984). An information station for the problem solver: System concepts. In *Proceedings of the First International Conference on Application of Mini- and Microcomputers in Information, Retrieval and Libaries* (pp. 111-118). Amsterdam: North-Holland.

McCall, R., Ostwald, J. L., Shipman, F. M., & Wallace N. F. (1990). The PHIDIAS hyperCAD system: Extending CAD with hypermedia. In *From research to practice: Proceedings of the 1990 Conference of the Association for Computer Aided Design in Architecture* (ACADIA 90; pp. 145–156). Big Sky, MT, USA: ACADIA.

McCall, R., Vlahos, E., & Zabel, J. (2001). Conceptual design and hyperSketching: Theory and Java prototype. In B. de Vries, J. van Leeuwen, & H. Achten (Eds.), *Computer-aided architectural design futures 2001* (CAAD Futures 2001; pp. 285–297). Dordrecht, Netherlands: Kluwer Academic Publishers.

Michalko, M. (2006). *Thinkertoys: A handbook of creative-thinking techniques* (2nd ed.). Berkeley, CA, USA: Ten Speed Press.

Osborn, A. F. (1963). *Applied imagination: Principles and procedures of creative problem solving*. New York: Scribner.

Potts, C., & Bruns, G. (1988). Recording the reasons for design decisions. In *Proceedings of the 10th International Conference on Software Engineering* (pp. 418–427). Washington, DC, USA: IEEE Computer Society Press.

Reuter, W. D. (1983). *Thesen und Empfehlungen zur Anwedung von Argumentativen Informationssystemen* [Theses and Recommendations for the Application of Argumentative Information Systems; internal report; Working Paper A-83-1]. Institut für Grundlagen der Planung, Universität Stuttgart, Germany.

Reuter, W. D., & Werner, H. (1984). *Zusammenstellung und Beschreibung von Anwendungsfaellen Argumentativer Informationssysteme* [Compilation and Description of Applications of Argumentative Information Systems; internal report; Working Paper A-84-4]. Institut für Grundlagen der Planung, Universität Stuttgart, Germany.

Rittel, H. W. J. (1966). Some principles for the design of an educational system for design. Available as Reprint 54, Institute of Urban and Regional Development, University of California, Berkeley, http://iurd.berkeley.edu/sites/default/files/pubs/RP54.pdf

Rittel, H. W. J. (1972). On the planning crisis: Systems analysis of the "first and second generations" In *Bedrifts Okonomen*, *8*, 390–396. Also available as Reprint 107, Institute of Urban and Regional Development, University of California, Berkeley, http://iurd.berkeley.edu/sites/default/files/pubs/RP107.pdf

Rittel, H. W. J. (1980). *APIS: A concept for an argumentative planning information system*. Working Paper 324, Institute of Urban and Regional Development, University of California, Berkeley, http://iurd.berkeley.edu/sites/default/files/wp/324.pdf

Rittel, H. W. J., & Noble, D. (1989). *Issue-based information systems for design*. Working Paper 492, Institute of Urban and Regional Development, University of California, Berkeley, http://iurd.berkeley.edu/sites/default/files/wp/492.pdf

Rittel, H. W. J., & Webber, M. (1973). Dilemmas in a general theory of planning. *Policy Sciences, 4*, 155–169. Also available as Reprint 86, Institute of Urban and Regional Development, University of California, Berkeley, http://iurd.berkeley.edu/sites/default/files/pubs/RP86.pdf

Schön, D. (1983). *The reflective practitioner: How professionals think in action*. New York: Basic Books.

Schuler, W., & Smith, J. B. (1990). Author's argumentation assistant (AAA): A hypertext-based authoring tool for argumentative texts. In A. Rizk, N. Streitz, & J. Andre (Eds.), *Hypertext: Concepts, systems and applications* (pp. 137–151). Cambridge, UK: Cambridge University Press.

Simon, H. A. (1969). *The sciences of the artificial*. Cambridge, MA, USA: The MIT Press.

Simon, H. A. (1973). The structure of ill-stuctured problems. *Artificial Intelligence*, *4*, 181–202.

Suchman, L. (1987). *Plans and situated actions: The problem of human–machine communication*. Cambridge, UK: Cambridge University Press.

Twombly, R. (2003). *Louis Kahn: Essential writings.* New York: W.W. Norton & Co.

Author's Note

All correspondence should be addressed to:
Raymond McCall
Department of Planning and Design
University of Colorado, Denver
Boulder, CO 80309
USA
mccall@colorado.edu

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

Human Technology

An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889

www.humantechnology.jyu.fi Volume 6 (1), May 2010, 38–54

JUXTAPOSING DESIGN REPRESENTATIONS FOR CREATIVITY

Alistair Sutcliffe
Manchester Business School
University of Manchester, UK

Abstract: *This paper argues that the influence of design rationale on creativity is best achieved by concurrent use of scenarios, prototypes and models. A framework of cognitive affordances is introduced to discuss the merits and limitations of each representation. The paper concludes by discussing how different representations might complement each other in creative scenario-based design.*

Keywords: *scenarios, prototypes, cognitive affordances, design representations*.

INTRODUCTION

It is often argued that creative design is best supported by examples of good design, thought probes, and stimulating artifacts (Cross, 2000; Gaver, Beaver, & Benford, 2003). In contrast, the methodical engineering approach to design emphasizes a systematic process, models, and the reuse of design knowledge, criticizing less systematic approaches as "craft" (Dowell & Long, 1998). Design rationale may provide a middle ground between the two approaches as an easy-to-use notation that can stimulate creativity while preserving some of the generality and rigor of models. I will investigate the contributions that different design representations can make to the creative design process from the viewpoint of cognitive reasoning processes. The relative merits of design rationale, scenarios, models, and prototypes are investigated in terms of their roles in the design process and cognitive affordances.

The integrated use of different representations will be illustrated by the scenario-based requirements analysis method (SCRAM; Sutcliffe & Ryan, 1997). SCRAM advocates a combination of design rationale, scenarios, and early prototypes as a means of effective requirements analysis and design exploration. More recently we have used a merge of SCRAM and scenario-based design (Carroll, 2002) with a similar combination of design representations in eScience health informatics domains (Sutcliffe et al., 2007). The following section of this paper describes the properties of different design representations. Next, I discuss how the representations can support creative reasoning, with the following section elaborating the theme by investigating cognitive affordances. Then I review how representations can be integrated into the design requirements discovery process. Integration is illustrated with the SCRAM method, followed by a brief review of other approaches to creative design support. The paper

URN:NBN:fi:jyu-20105241906

concludes by reviewing the potential for juxtaposing different design representations for creative design, as well as requirements specification of systems.

DESIGN REPRESENTATIONS

This section reviews the role of the more common design representations in creative design from a human–computer interaction (HCI) perspective and from the more analytic view of software engineering.

Scenarios

One of the key distinctions between scenarios and any model is that the former are grounded examples of specific experience, whereas models are more abstract representations of phenomena in the real world. Unfortunately, the term *scenario* has been abused in the literature and a large number of definitions exist (see Rolland et al., 1998). Indeed, much of the scenario literature, especially in the software engineering tradition (Kaindl, 1995), is in fact describing event–sequence traces through state transition models. In object-oriented design it becomes difficult to distinguish between use cases, alternative paths through use cases, and scenarios, which are just another path through a use case (Cockburn, 2001; Graham, 1996; Jacobson, Christerson, Jonsson, & Overgaard, 1992).

Scenarios have several roles in design; according to Carroll, one of these is a “cognitive prosthesis,” or an example to stimulate the designer’s imagination. Scenarios and other techniques, such as claims, are lightweight instruments that guide thought and support reasoning in the design process (Carroll, 2002). Carroll has articulated several different roles for scenarios in the design process, including envisionment for design exploration, requirements elicitation, and validation (Carroll, 1995). Usage scenarios illustrate problems for analysis and initiating or visioning scenarios stimulate design of a new artifact, while projected use scenarios describe future use of an artifact that has been designed (Sutcliffe & Carroll, 1998). Scenarios can promote creative reasoning by stimulating examples and vivid illustration of real-life problems.

One problem with scenarios is that extreme examples might bias reasoning towards exceptional and rare events, or towards the viewpoint of an unrepresentative stakeholder. These biases are an acknowledged weakness of scenarios; however, some proposed scenarios are deliberately exceptional to provoke constructive thought (Djajadiningrat, Gaver, & Frens, 2000). Although scenarios are useful as cognitive probes for design, this is not their only role.

Scenarios arguably are the starting point for all modeling and design, and contribute to several parts of the design process. For instance, Potts (1999) has advocated scenarios to validate or check the acceptability of designs. The process of generalization inevitably loses detail, and the analyst has to make judgments about when unusual or exceptional behaviors are omitted, or explicitly incorporate them in task models as branches in action sequences. Hence one criticism that can be leveled at scenarios is that gathering detail comes at the price of effort in capturing and analyzing a “necessary and sufficient” set of scenarios.

Models

A prime role of models, either in the HCI tradition of task modeling or in software engineering (e.g., use cases, class diagrams, activity sequence diagrams, UML, etc.), has been to specify the system and represent the problem space to support design reasoning.

One criticism of models is that they do not capture the richness of interaction that occurs in the real world, compared with scenario narratives that concentrate on contextual description (e.g., Kuutti, 1995; Kyng, 1995). For instance, software engineering and task models may be criticized for not representing the relationships between agents, activity, and organizational structures, although these concepts are described in sociotechnical system design frameworks such as ORDIT (Eason, Harker, & Olphert, 1996). Meanwhile, a more comprehensive modeling language can be found in the i* requirements engineering method that analyzes the dependencies between agents, tasks, goals, and resources (Mylopoulos, Chung, & Yu, 1999; Yu, 1993). Models can expose design dilemmas and inconsistencies and thereby support the generation of creative solutions; however, how well models expose problems depends on the clarity of their notations and the reasoning mechanisms associated with the model. Models show an abstract view of problems so they might be accused of having a narrow scope of phenomena and omit detail, whereas scenarios might be able to represent phenomena in more detail, but they do so in an ad hoc manner and leave the responsibility of generalization to the analyst. Of course, models can be used with scenarios, and this theme is elaborated later in this paper.

Design Rationale

The essence of design rationale (DR) is to represent argumentation and knowledge within the design process. Hence DR can be viewed as models that are specialized to represent the problem space for decision making, including evidence for evaluating alternative designs. Various forms of DR have appeared since their genesis in Toulmin's (1958) argumentation semantics, notably issue-based information systems and the diagrammatic form gIBIS (Conklin & Begeman, 1988), which represents issues (design problems to be solved), alternatives (possible solutions), and evidence that supports or detracts from each alternative. The most influential HCI variant of DR recapitulates the semantics as questions (design problem), options, and criteria (QOC; MacLean & McKerlie, 1995; MacLean, Young, Bellotti, & Moran, 1991). DR can also be used to express generalizable knowledge accumulated during iterative design. Psychological DR, or claims (Carroll & Rosson, 1992), uses a simpler semantic representation of the claim (problem statement), a solution (expressed as an artifact/design pattern), and arguments divided into upsides and downsides. Claims may be used to support reasoning during the design process (Carroll, 2002) or present reusable knowledge by recording the results of evaluation, including the problem that motivated a general design principle—called a claim—with trade-offs expressed as upsides and downsides (Carroll, 2000; Sutcliffe & Carroll, 1999). When DR is used to support the design process, the trade-off concerns for a claim about DR representations might invite comparison of the cost of representing the design space versus the advantage gained in more effective reasoning. The juxtaposition of alternatives is a key affordance for creative reasoning. For collaborative decisions, DR diagrams can function as a shared representation to focus discussion, although the costs may well outweigh the benefits. The uptake of DR in industry has been slow. When representing reusable knowledge, the benefits of DR may be potentially larger, but reuse depends on an effective knowledge management and retrieval system.

Prototypes

This category includes a variety of design representations, ranging from paper (or computer-based) storyboards to mock-ups/concept demonstrators with limited scripted functionality and prototypes with a partial software implementation. Prototypes stimulate creative design because they engage the user (designer) with the material of the product, be that software or hardware. Experimentation becomes part of the implementation unless a rigorous specification in detail of the implementation process is adopted, as practiced in software engineering. The prototype artifacts all result from the creative design process and, unlike models and DR, show the user concrete aspects of a design. Prototypes, mock-ups, and storyboards are probably the most common ways of representing the problem space for creative design exploration. This applies not only in software-related products but also in many other areas of creative design. The variation between the techniques lies in the media used (paper, video, computer media, interactive software), the cost of production, and the fidelity and extent of the representation of the intended design. While very early creative brainstorming may be used to map out a space of ideas and concepts, once these have been prioritized, design realization becomes necessary to progress the user–designer dialogue. The power of the prototype lies in anchoring the focus of discussion in a concrete example, and stimulating user reaction to specific features.

REPRESENTATIONS AND REASONING PATHOLOGIES

In this section, the merits of different representations are reviewed in light of how they can stimulate and support creative design. Scenarios use language and concepts that are readily accessible to users and domain experts, whereas tasks and other conceptual models are expressed in a specialized language that users have to learn. Because scenarios invoke specific memory schema associated with experience or similar stories, they help to recruit specific knowledge (Carroll, 2000; Sutcliffe, 2002). This tunes our critical faculties, since detail tends to provide more subject matter to detect inconsistencies and errors when we reason about models and specifications.

In contrast, models are harder to comprehend because they represent abstract generalizations. While people naturally form categorial abstractions of physical things (Rosch, Mervis, Gray, Johnson, & Boyes-Braem, 1976), we are less efficient at forming categories of abstract concepts and functions (Hampton, 1988). Unfortunately, formation of conceptual-functional categories is a necessary part of the generalization process, so users can find reasoning with simple conceptual models, such as data flow diagrams, difficult (Sutcliffe & Maiden, 1992). Once learned, models become memory schema that represent abstract concepts removed from everyday experience, so their effectiveness depends on how well connected they are to more specialized memory schema representing scenario-based knowledge. The importance of the connection becomes clear when we try to validate models. Without any connection to specialized knowledge, I can accept the validity of the general concept simply because it has a wide scope of meaning. For example, I might accept the proposition that <all birds can fly> as a true type definition of the class <birds> in the absence of more specific knowledge of penguins, kiwis, rheas, ostriches, and dodos. Models therefore need to be integrated examples and scenarios and, furthermore, cannot exist profitably without them; indeed, human categorial memory is probably an integration of abstract models and specific examples (Lakoff & Johnson, 1999).

While scenarios might be effective in grounding reasoning, their downsides lie in reasoning biases and partial mental model formation. Confirmation bias is a well-known weakness of human reasoning (Johnson-Laird & Wason, 1983). We tend to seek only positive evidence to support hypotheses, so scenarios can be dangerous in supplying us with minimal evidence to confirm our beliefs. While problem statement scenarios and anti-use cases (I. Alexander, 2002) can counteract confirmation bias, we need to be wary of this downside. Another potential pathology is encysting, more usually described by the saying "can't see the wood for the trees." Since scenarios are detailed, they can bias people away from the big picture of important design issues and towards obsession with unnecessary detail. Models exist to counteract this pathology. Partial mental model formation is another weakness when we test hypotheses without sufficient reasoning (Simon, 1973). Scenarios can encourage this pathology by reassuring us that we have covered all aspects of the problem with a small number of scenarios. This exposes the Achilles heel of scenario-based reasoning: It is difficult, if not nearly impossible, to be confident that a necessary and sufficient set of scenarios has been gathered to escape from the partial mental model problem.

Prototypes and other concrete design realizations share many of the same pathologies with scenarios, such as encysting and confirmation bias, since users might be prone to accepting a design to please the designer. This may be critical when the power relationships give designers a de facto authority over users, which they should strive to avoid. Groups of users may also be prone to suppress criticism of a design and agree with the consensus, following a group-think bias. However, prototypes do afford concrete representations and detail that users can react to, as well as anchoring discussion to specific issues/features, which can facilitate users' participation in the creative design process.

Many of the same criticisms can be leveled at DR as a genre of models. Although DR represents the decision space with specific issues in some detail, arguments may make little sense without the background knowledge contained in other representations. Also, DR may bias problem exploration by presenting a ready-made set of alternatives. Furthermore, unless the author of the DR diagram is careful, the diagram can embed biases from the author's viewpoint in the relationships between the alternatives and supporting/detracting evidence (Karsenty, 1996; Sutcliffe, 2002; Sutcliffe & Ryan, 1997).

AFFORDANCES AND REPRESENTATIONS

While analysis of general properties of representations can provide some insight into their potential contribution towards supporting design, a more detailed view is necessary to unpack the nature of cognitive affordances. The term *affordance* was borrowed by Norman (1999) from Gibson's (1986) concept of physical features that suggested or afforded intuitive understanding, for instance, cliffs suggest the danger of falling. As Norman realized, when the concept of affordances is applied to design features, the meaning of the term becomes more complex, since it has to account for the general suggestibility of the external form towards some purposeful use and the cognitive internalization of the external form into an individual's plan of action; for example, a slider control on a user interface suggests movement of the control itself that then changes another component, such as panning a display.

A useful distinction, therefore, is to examine the external appearance of a representation (or design) and its integration into action plans after people have interpreted its meaning. To illustrate this line of inquiry, I will compare three exemplars of design rationale: gIBIS, QOC, and claims. The first two have a similar external form but differ in their semantics. Claims, in contrast, have a different (text-based) external form and semantics.

gIBIS diagrams, as illustrated in Figure 1, have a simple tree/network structure that can be traced from the root node representing the issue, to the branches (two or more design/solution alternatives), and then to leaf nodes representing supporting arguments. The graphical form intuitively suggests composition and relationships, as do most hierarchy diagrams. The semantics of the diagram nodes are easily explained so the representation can be used to trace relationships from the issues through each alternative solution to the supporting (or detracting) arguments. Thus the representation "affords" comparison of alternative solutions by pathway tracing. Furthermore, the external representation reduces working memory loading since different pathways can be reviewed at will.

QOC (see Figure 2) has a similar graphical notation, so the diagram also affords pathway tracing of questions or different design options. However, criteria and arguments have subtly different semantics. Criteria are more terse and represent concepts by which trade-off decisions can be made, rather than arguments that record the results of reasoning about different alternatives. Criteria therefore invite more in-depth reasoning about the options and their relationship to one or more criteria; hence, QOC may stimulate more creative thought by provoking reasoning. This conjecture would require experimental study to assess the quality of reasoning invoked by each representation; nevertheless, the comparison illustrates how graphical forms and the semantics attached to diagrams might influence reasoning.

Claims, also termed psychological design rationale by Carroll (Carroll & Rosson, 1992), do not share the diagram representation; instead, formatted text is used to illustrate the structure

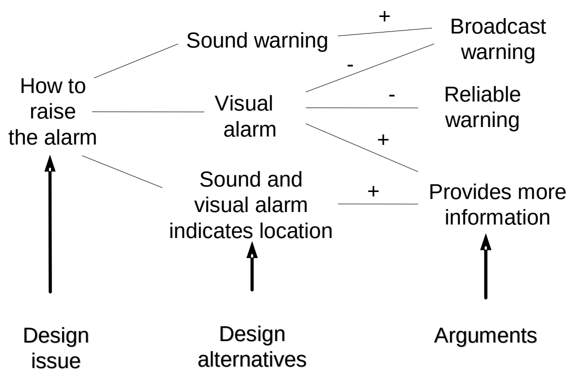

Figure 1. In the gIBIS design rationale, the + or – signs denote arguments that either support or hinder a particular alternative.

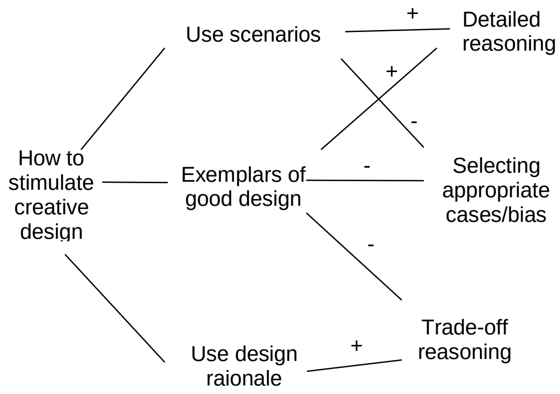

Figure 2. The QOC (questions, options, criteria) form of design rationale, applied to trade-offs between representations to support creative design.

of a claim (see Figure 3). The basic components of claims are (a) the claim (essentially a design principle); (b) upside and downside trade-offs that may arise from application of the claim; (c) a scenario of use; and (d) an artifact illustrating a design that embeds the claim. The juxtaposing of alternatives has been moved from alternative designs to the assessment criteria or arguments. Claims present essentially only one design alternative and then positive and negative arguments about its merits. While DR provides more structured arguments, claims use the combination of a design solution (a generalized design principle) with examples of use illustrated in scenarios and artifacts. Claims may therefore stimulate creative thought by the challenge posed from the general assertion about a design treatment (the claim), concrete illustrations of its interpretation and use, and the results of previous design experience recorded in the upsides and downsides. Design patterns (Borchers, 2001) follow a similar format with forces, scenarios and illustrations of exemplar design for the pattern.

So how do other representations compare with the affordances of DR? Models share diagrammatic notation with DR but have many more morphologies and semantics, ranging from the simple (e.g., use case diagrams) to the very complex (e.g., i* requirements modeling language; Yu, 1993). It is notable that most semiformal modeling notations rely on a restricted

Claim ID: Colour-coded Telegraphic Display
Author: Singley, M.K.; Carroll, J.M.
Artifact: MoleHill tutor - Goalposter tool
Description: A colour-coded telegraphic display of goals
Upside: Provides persistent feedback on the correctness of actions, as well as access to further information
Downside: Learners must learn the display's feature-language and controls
Scenario: The presentation of individual goals in the window is telegraphic, several words at most. However, the learner can expand any of the telegraphic goals (through a menu selection) to display a fuller explanation of why the goal is worthwhile pursuing or not. Thus the system provides both shorthand feedback on correctness and access to further help.

Figure 3. Claim showing components in structured text format.

number of graphical formats that afford intuitive interpretations, namely hierarchies (task models, class diagrams), networks (data flow diagrams, activity sequence diagrams), and timelines (Gantt charts, interaction diagrams). Complexity arises when diagram notations become overloaded with symbols to represent a large variety of relationships and objects, as illustrated in Figure 4.

Scenarios, in contrast, use natural media (text and image) for representing concrete examples of experience. Prototypes also represent concrete examples of designs as physical artifacts, with storyboards and mock-ups providing representations of the physical form early in the design process.

Cognitive affordances therefore emerge from intuitive understanding or representations in a variety of media, coupled with reasoning about the content of those representations. DR and models provide abstract representations of knowledge and design trade-offs to support creative reasoning, while scenarios and prototypes give grounded examples from which to abstract more general principles. However, creative thought should generate innovative designs, but these need to be based on general principles; otherwise, design is limited to a craft-style incremental improvement of specific examples (Dowell & Long, 1998). I argue that a combination of representations is the most productive way to stimulate creative design, a challenge addressed in the next section.

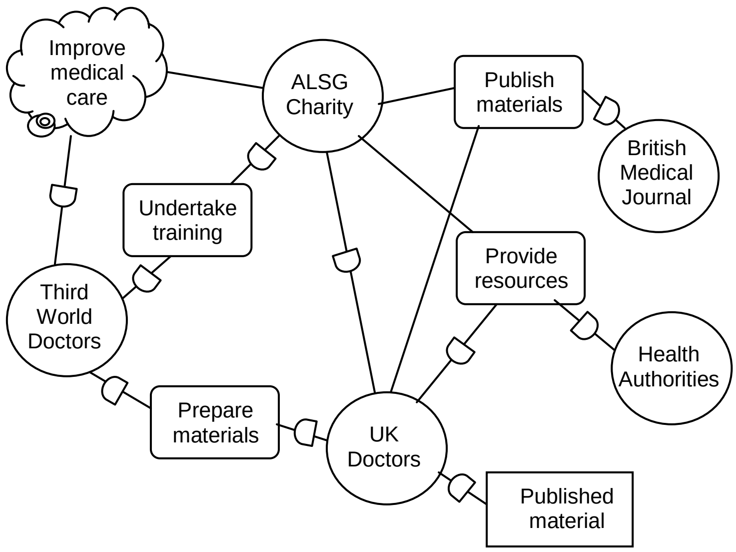

Figure 4. An i* strategic dependency model illustrating complex modeling. The circles denote agents, boxes are resources, rounded boxes are goals, and clouds are quality goals. The D symbols stand for "depends on" relationships.

INTEGRATING REPRESENTATIONS

One productive juxtaposition of scenarios and models is to use scenarios as test data to validate design models. This approach has been actively researched in the inquiry cycle (Potts, Takahashi, & Anton, 1994), which recommended using scenarios as specific contexts to test the utility and acceptability of prototype system output. By questioning the relevance of system output for a set of stakeholders and their tasks described in a scenario, the analyst can discover obstacles to achieving system requirements. Input events can be derived from scenarios to test validation routines and other functional requirements. This process stimulates reasoning by integrating two physical representations, operating a prototype to produce output with scenarios of potential use. HCI uses scenarios in a similar manner in usability evaluation, although the role of scenarios is not articulated so clearly. Nevertheless, task or test scripts in evaluation methods (Monk & Wright, 1993) are scenarios.

Claims have evolved through several iterations of more or less integrated representations. In their original form, claims united scenarios, illustrating a problem with DR presenting the upsides and downsides of usability arguments as trade-offs for applying a design principle with a concrete example of an implementation. Claims are situated in a context by a scenario of use and the artifact that helps designers understand how to apply usability arguments. Since claims have a domain-specific anchor in the artifact context, insight into more general design implications and trade-offs may be gained if they can be integrated with models.

By associating claims in this manner, the designer can have the best of both worlds. Claims with their associated artifacts and scenarios provide grounded examples of design advice while models represent a more general context within which to consider the implications of the design decision. This view of claims is similar to the schema of patterns that recommend that design advice is presented in the context of a motivating problem, and with an example of its application (Borchers, 2001). Although patterns do have a clause that indicates the range of problems the design advice can be applied to, this scoping is ad hoc. Advocates of patterns proposed relationships between individual patterns constructed into a hypertext-like pattern network or language (C. Alexander, Ishikawa, & Silverstein, 1977) to set the context. Unfortunately, pattern languages tend to be incomplete. Claims have been integrated with models that may be specific to the application, or generalized models of tasks to stimulate reuse of knowledge (Sutcliffe & Carroll, 1999); see Figure 5.

The scope of the claim is defined by models that may be related to particular applications, for example, task models, class diagrams for a telephone fault-finding application, or more generic models capturing a range of applications (e.g., generic models of diagnostic tasks, including fault finding). One of the problems with integrating claims with

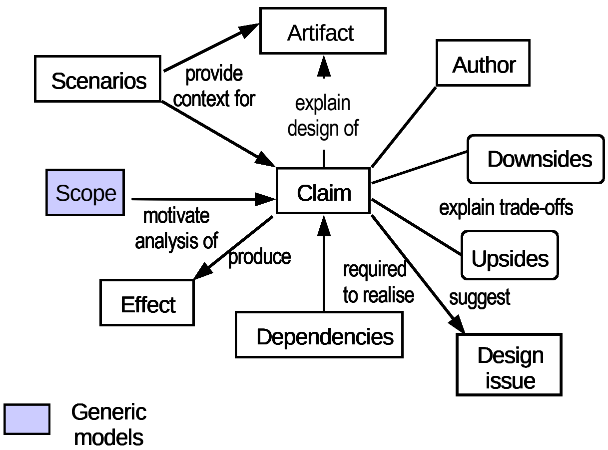

Figure 5. An extended claims schema, associating claims with reusable generic models and supporting arguments.

models and other arguments lies in the complexity of the number of representations, which in turn necessitates further guidance about how the representations may be combined in the design process. More elaborate representations therefore run into the criticism leveled at the engineering approach: The complexity of models and process advice militates against the creative freedom necessary in design. In the following section, I introduce the SCRAM method, which attempts to tread a middle path between creative use of multiple representations and a systematic approach.

The SCRAM Method

The SCRAM method (Sutcliffe & Ryan, 1997) for analyzing the requirements for interactive systems provides one way forward for integrating representations. The approach is based on integrating three representations:

Prototypes or concept demonstrators provide a designed artifact that users can react to;

Scenarios, in which the designed artifact is situated in a context of use, thereby helping users relate the design to their work/task context;

Design rationale, where the designer's reasoning is deliberately exposed to the user to encourage user participation in the decision process.

The representations are combined with a method to provide process guidance, composed of advice on setting up sessions, and more detailed guidance on fact acquisition and requirements validation. The method consists of the following phases:

1. Initial requirements capture and domain familiarization. This is conducted with conventional interviewing and fact-finding techniques to gain sufficient information to develop a first-concept demonstrator.
2. Specification and development of the concept demonstrator. I define a concept demonstrator as a very early prototype with limited functionality and interactivity, so it can only be run as a script to illustrate a typical task undertaken by the user. *Scripts* illustrate a scenario of typical user actions with effects mimicked by the designer. Concept demonstrators differ from prototypes in that no real functionality is implemented and the user cannot easily interact with the demonstrator since many functions are not implemented.
3. Requirements analysis-design exploration session. The users involved in the initial requirements capture interview are invited to critique the concept demonstrator and interview the designer. The session is recorded for subsequent analysis.
4. Session analysis. Data collected are analyzed and conclusions are reported back to the users. This frequently leads to a further iteration of revising the concept demonstrator and another analysis session.

The end point of the method delivers the concept demonstrator, a set of analyzed DR diagrams expressing users' preferences for different design options, and specifications as text, diagrams, or more formal notations, depending on the designer's choice. In addition, video of the analysis sessions is available for requirements traceability analysis.

The walkthrough method employs scenario scripts that describe an imaginary work situation for the user and a typical key task. The session is started with an introduction and verbal summary of the situation described in the scenario narrative, for example, “Imagine you are in your office and a production order arrives ....” One developer operates the concept demonstrator while the explainer-rapporteur asks questions at key points in the demonstration script.

At key points in the sequence, a designed response to a requirement is illustrated. This is best explained by reference to the example used for the validation, which is covered later. Figure 6 illustrates a screen dump from a shipboard emergency management system. The user’s requirement is for timely and appropriate information to support decision making. The operational steps accompanying Figure 6 are

User: identify the hazard location
System: shows location of fire
User: sound alarm
User: find location of fire-fighting crews
System: displays crew information and location on the diagram
User: decide appropriate instructions to give to crew
System: displays a checklist of actions.

The key point in the task is how to instruct the emergency team on where to go and how to deal with the hazard, in this case a fire. The concept demonstrator illustrates one design option. Alternative solutions expressed in a DR format are illustrated in Figure 7. The user’s attention is drawn to the design options, in this case providing complete information for decision support. The first option displaying comprehensive information is illustrated with the demonstrator; this is followed by option 2, provision of more restricted but relevant information for the task, by identifying the team nearest the fire; and then the final option to give the emergency team autonomy and broadcast the location of the fire. The users are asked to rate each option and consider the trade-off criteria. The diagram also functions as a recording medium, since ranking of options, additional ideas, and notes can be scribbled on top of the diagram. Indeed, in many cases, discussion may promote redrawing the diagram.

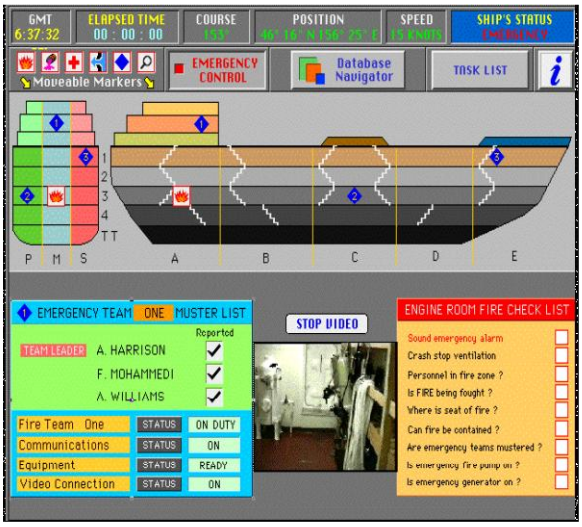

Figure 6. Concept demonstrator showing the “show emergency teams and hazard location” design option for the Muster emergency teams task.

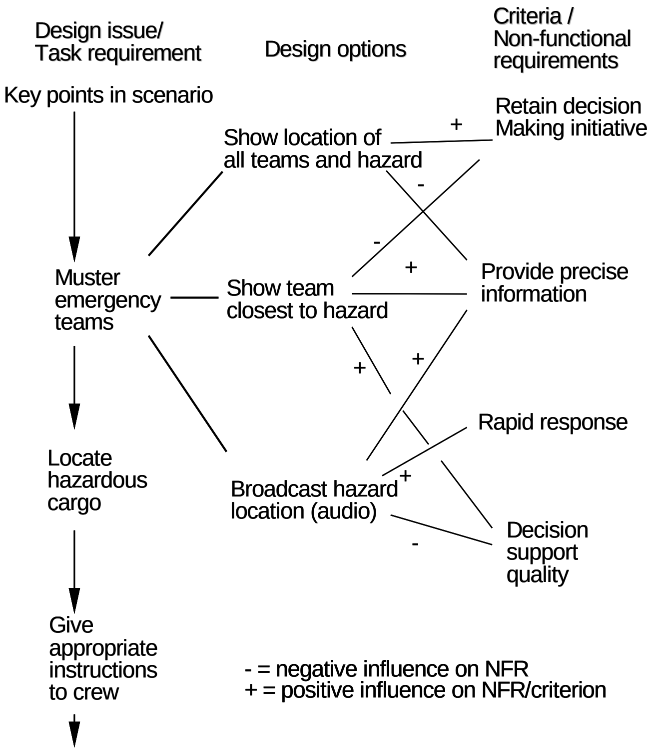

Figure 7. A design rationale diagram used as a key point in the concept demonstrator script.

The DR diagram is used as a shared artifact to promote discussion, and gesture is used where possible to illustrate differences between the options by pointing to the screen. One obvious problem is bias towards the option implemented in the demonstrator. This can be counteracted by using storyboard sketches of the other options and by more vigorous critiquing by the developers of the implemented version. In particular, use of the criteria is a powerful way of promoting critical thought. The motivation for using design rationale is to explore the possible solution space with the user. Rationale diagrams enable this to be done cost-effectively, since only one version of the demonstrator is produced. However, should additional resources be available, alternative versions of the artifact can be implemented and both versions illustrated at the key point.

Evaluation of SCRAM demonstrated that more detailed requirements and design feedback were captured using the method than with conventional requirements analysis techniques without multiple representations (Sutcliffe & Ryan, 1997). The method also stimulated creative reasoning about new design solutions through the process of critiquing the concept demonstrator and alternative solutions that were presented.

Creative Combinations

Although integrated representations show considerable promise in stimulating design reasoning, the creativity and cognition literature suggests that challenging content, shifting viewpoints, metaphor, and analogies also play important roles in creative reasoning (Cross, 2000, 2002; Karsenty, 1996). Selecting concrete examples and setting up contrasts may therefore be an important extension to DR and integration of representations (Buxton, 2007).

DR affords configuration of challenges and contrasts by its comparison of trade-offs between alternatives; however, to challenge thought requires a considerable shift in the traditional view of DR as a discussion forum for trade-offs to active design of rationale. This could be explored by creating unusual solutions or criteria that challenge the conventional assumptions about a design. Contrasts might be borrowed from value-based design (Friedman, 1997), sketching stories (Buxton, 2007), or extreme characters (Gaver et al., 2003) to produce challenges. Another example is the emergence of antipatterns (I. Alexander, 2002) as a means of stimulating counterintuitive reasoning. Combinations of representation and content could provide a design space for creative exploration, using probes based on analogies and metaphors that alter viewpoints on a design problem with cultural probes and unusual examples of design (Gaver et al., 2003) to stimulate thought.

Choice of how many and which representations to combine is a complex question for further research. I expect the answer may be "horses for courses," meaning, for wide-ranging creative design with green-field applications, sketches, storyboards, and scenarios (Buxton, 2007; Moggridge, 2006) may be the best choice, although as Nigel Cross (2002) noted, expert designers still reuse basic knowledge in the form of "first principles." Such knowledge could be passed on as DR or claims. In more constrained contexts, creative reasoning may be better supported with more detailed representations, models, and specifications (Kaindl, 1995; Paterno, 1999). Although multiple representations were effective in SCRAM (see previous section), there were limits; I did not integrate models with the other representations, since the management of artifacts, DR, and scenarios within one session was already complex. A future view on the representation creativity problem may be to evaluate how representations contribute to the "common ground" (Clark, 1996) between the parties in design conversations. Representations need to promote shared understanding between the parties according to their prior knowledge and the design problem in hand.

CONCLUSIONS

The argument advocated in this paper is that the constructive tension between different types of representation is productive since they have different affordances for abstract reasoning and detailed critiquing. Unfortunately, there are obstacles in the way of using multiple representations, even though many advocate them in HCI and software engineering (Mylopoulos et al., 1999; Paterno, 1999; Sutcliffe, 2000). In reality, it is difficult to get practitioners to accept complex representations, and even simple ones get misused and customized to individuals' needs. Take MacLean et al.'s (1991) QOC variant of DR as an example. This is a simple representation of a design question, alternative solutions, and evaluation criteria for the solutions. However, QOC has been difficult to introduce into new communities of practice (MacLean & McKerlie, 1995), and similar problems have been encountered with the gIBIS (Conklin & Begeman, 1988) version of DR (Buckingham Shum, 1996; Sutcliffe & Ryan, 1997). Carroll, in his more recent work, has simplified claims (Carroll, 2000; Rosson & Carroll, 2001), abandoning complex formatting. Claims are presented as simple design principles, in association with a motivating scenario and occasionally an artifact. In terms of process, Carroll advocates a more creative view of design, with scenarios playing roles of "thought prostheses" and challenges for design.

So is there a synthesis for the model-analytic and creative-exploration approaches to design? A partial answer is acknowledging the "horses for courses" argument. A differentiation between formal model-analytic and creative-exploratory design approaches will always be necessary for applications that range between safety critical, on the one hand, and those oriented toward entertainment, education, and general commerce on the other. So different combinations of design presentations have contributions to make in different phases of design and design contexts. Scenarios and prototypes can stimulate thought and provoke argument on detail, whereas models give the wider, more abstract context for design reasoning. DR can provide the link between the two, although its effectiveness in supporting decision making or just documenting the result is an open question. Finally, design representations enable knowledge to be reused effectively in a generalized form as models, claims, principles, and guidelines.

Combinations of representations using prototypes, scenarios, and claims are advocated in scenario-based design (Carroll, 2002), and these have been successfully applied in eScience applications (Thew et al., 2008; Thew et al., 2009), as well as in Carroll's development of collaboration tools for eCommunity applications (Carroll & Rosson, 1996). SCRAM integrated DR with scenarios and early prototypes and this proved to be an effective combination for critiquing designs and stimulating further design ideas (Sutcliffe & Ryan, 1997). Prototypes with scenarios and formatted question lists have been successfully applied in requirements analysis (Sutcliffe, Gault, & Maiden, 2005), which, while not using DR explicitly, did present issue lists and alternatives to provoke design reasoning. The scenario presenter tool evolved from earlier research on automated support for design reasoning with tools that produced question prompts linked to specific locations in a scenario or use case, thus giving more active support for design reasoning (Sutcliffe, Maiden, Minocha, & Manuel, 1998). A combination of scenarios, prototypes, and lightweight design representations appears to have evolved in several research strands suggesting that combining representations has some utility.

The more general question that requires considerable future research is how juxtaposing contrasts in the content can augment the combination of different representations. Challenges to reasoning from usual content are known to induce creative reasoning; however, how such content can be produced for specific situations is not clear. Seeding the design environment with stimulating content (Fischer, 1996) assumes considerable insight into future problems. Although premade solutions might inhibit creative reasoning if designers just take the easy option of reusing previous solutions, cognitive probes in the forms or personae, and extreme characters (Djajadiningrat et al., 2000), can stimulate thought. In future work I will use common ground (Clark, 1996) as a theoretical framework for exploring cognitive probes and DR, as well as a combination of representations to support creative design reasoning.

REFERENCES

Alexander, C., Ishikawa, S., & Silverstein, M. (1977). *A pattern language*. Oxford, UK: Oxford University Press.

Alexander, I. (2002). Initial industrial experience of misuse cases in trade-off analysis. In *Proceedings of RE-02 IEEE Joint International Conference on Requirements Engineering* (pp. 61–70). Los Alamitos CA: IEEE Computer Society Press.

Borchers, J. (2001). *A pattern approach to interaction design*. Chichester, UK: Wiley.

Buckingham Shum, S. (1996). Analyzing the usability of a design rationale notation. In T. P. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques and use* (pp. 185–215). Hillsdale NJ, USA: Lawrence Erlbaum Associates.

Buxton, W. (2007). *Sketching user experiences: Getting the design right and the right design*. Amsterdam: Elsevier.

Carroll, J. M. (Ed.). (1995). *Scenario-based design: Envisioning work and technology in system development*. New York: Wiley.

Carroll, J. M. (2000). *Making use: Scenario-based design of human-computer interactions*. Cambridge, MA, USA: MIT Press.

Carroll, J. M. (2002). Making use is more than a matter of task analysis. *Interacting with Computers, 14, 619*–627.

Carroll, J. M., & Rosson, M. B. (1992). Getting around the task-artifact framework: How to make claims and design by scenario. *ACM Transactions on Information Systems*, *10*, 181–212.

Carroll, J. M., & Rosson, M. B. (1996). Developing the Blacksburg Electronic Village. *Communications of the ACM*, *39*, 69–74.

Clark, H. H. (1996). *Understanding language*. Cambridge, UK: Cambridge University Press.

Cockburn, A. (2001). *Writing effective use cases*. Boston: Addison-Wesley.

Conklin, J., & Begeman, M. L. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *ACM Transactions on Office Information Systems*, *64*, 303–331.

Cross, N. (2000). *Engineering design methods: Strategies for product design*. Chichester, UK: Wiley.

Cross, N. (2002). Creative cognition in design: Processes of exceptional designers. In *Proceedings of the 4th Conference on Creativity and Cognition* (pp.14–19). New York: ACM Press.

Djajadiningrat, J. P., Gaver, W. W., & Frens, J. W. (2000). Interaction relabelling and extreme characters: Methods for exploring aesthetic interactions. In D. Boyarski & W. A. Kellogg (Eds.), *Conference Proceedings: DIS2000 Designing Interactive Systems: Processes, Practices Methods and Techniques* (pp. 66–71). New York: ACM Press.

Dowell, J., & Long, J. L. (1998). A conception of the cognitive engineering design problem. *Ergonomics*, *41*, 126–139.

Eason, K. D., Harker, S. D. P., & Olphert, C. W. (1996). Representing socio-technical systems options in the development of new forms of work organisation. *European Journal of Work and Organisational Psychology*, *5*, 399–420.

Fischer, G. (1996). Seeding, evolutionary growth and reseeding: Constructing, capturing and evolving knowledge in domain-oriented design environments. In A. G. Sutcliffe, D. Benyon, & F. Van Assche, (Eds.), *Domain knowledge for interactive system design; Proceedings: TC8/WG8.2 Conference on Domain Knowledge in Interactive System Design* (pp. 1–16). London: Chapman & Hall.

Friedman, B. (Ed.). (1997). *Human values and the design of computer technology*. Cambridge, UK: Cambridge University Press.

Gaver, W. W., Beaver, J., & Benford, S. (2003). Ambiguity as a resource for design. In V. Bellotti, T. Erickson, G. Cockton, & P. Korhonen (Eds.), *CHI 2003 Conference Proceedings: Conference on Human Factors in Computing Systems* (pp. 233–240). New York: ACM Press.

Gibson, J. J. (1986). *The ecological approach to visual perception*. Hillsdale, NJ, USA: Lawrence Erlbaum Associates.

Graham, L. (1996). Task scripts, use cases and scenarios in object oriented analysis. *Object-Oriented Systems*, *3*, 123–142.

Hampton, J. A. (1988). Disjunction in natural categories. *Memory and Cognition*, *16*, 579–591.

Jacobson, I., Christerson, M., Jonsson, P., & Overgaard, G. (1992). *Object-oriented software engineering: A use-case driven approach*. Reading, MA, USA: Addison Wesley.

Johnson-Laird, P. N., & Wason, P. C. (1983). *Thinking: Readings in cognitive science*. Cambridge, UK: Cambridge University Press.

Kaindl, H. (1995). An integration of scenarios with their purposes in task modelling. In G. M. Olson & S. Schuon (Eds.), *Designing Interactive Systems: DIS 95 Conference* (pp. 227–235). New York: ACM Press.

Karsenty, L. (1996). An empirical evaluation of design rationale documents. In *Proceedings of CHI'96 Conference: Human Factors in Computing Systems* (pp. 150–156). New York: ACM Press.

Kuutti, K. (1995). Workprocess: Scenarios as a preliminary vocabulary. In J. M. Carroll (Ed.), *Scenario-based design* (pp. 19–36). New York: Wiley.

Kyng, M. (1995). Creating contexts for design. In J. M. Carroll (Ed.), *Scenario-based design* (pp. 85–108). New York: Wiley.

Lakoff, G., & Johnson, M. (1999). *Philosophy in the flesh: The embodied mind and its challenge to western thought*. New York: Basic Books.

MacLean, A., & McKerlie, D. (1995). *Design space analysis and user-representations* (Technical Report EPC-1995-102). Cambridge, UK: Xerox Research Centre Europe.

MacLean, A., Young, R. M., Bellotti, V., & Moran, T. P. (1991). Questions, options and criteria: Elements of design space analysis. *Human-Computer Interaction*, *6*, 201–250.

Moggridge, B. (2006). *Designing interaction*. Cambridge, MA, USA: MIT Press.

Monk, A. G., & Wright, P. (1993). *Improving your human-computer interface: A practical technique*. Boston: Prentice Hall.

Mylopoulos, J., Chung, L., & Yu, E. (1999). From object-oriented to goal-oriented requirements analysis. *Communications of the ACM*, *42*, 31–37.

Norman, D. A. (1999). *The invisible computer: Why good products can fail, the personal computer is so complex, and information appliances are the solution*. Cambridge, MA, USA: MIT Press.

Paterno, F. (1999). *Model-based design and evaluation of interactive applications.* Berlin, Germany: Springer-Verlag.

Potts, C. (1999). ScenIC: A strategy for inquiry-driven requirements determination. In *Proceedings: 4th IEEE International Symposium on Requirements Engineering* (pp. 58–65). Los Alamitos CA, USA: IEEE Computer Society Press.

Potts, C., Takahashi, K., & Anton, A. I. (1994). Inquiry-based requirements analysis. *IEEE Software*, *11*, 21–32.

Rolland, C., Ben Achour, C., Cauvet, C., Ralyte, J., Sutcliffe, A. G., & Maiden, N. A. M. (1998). A proposal for a scenario classification framework. *Requirements Engineering*, *3*, 23-47.

Rosch, E., Mervis, C. B., Gray, W., Johnson, D., & Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology*, *7*, 573–605.

Rosson, M. B, & Carroll, J. M. (2001). *Usability engineering: Scenario-based development of human computer interaction*. San Francisco: Morgan Kaufmann.

Simon, H. A. (1973). The structure of ill-structured problems. *Artificial Intelligence*, *4*, 181–201.

Sutcliffe, A. G. (2000). Bridging the communications gap: Developing a lingua franca for software developers and users. In *Actes du XVIIIe Congres: INFORSID* (pp. 13–32). Toulouse, France: Inforsid.

Sutcliffe, A. G. (2002). *User-centred requirements engineering*. London: Springer.

Sutcliffe, A. G., & Carroll, J. M. (1998). Generalizing claims and reuse of HCI knowledge. In H. Johnson, L. Nigay, & C. Roast (Eds.), *People and computers XIII; Proceedings: BCS-HCI Conference* (pp. 159–176). Berlin, Germany: Springer-Verlag.

Sutcliffe, A. G., & Carroll, J. M. (1999). Designing claims for reuse in interactive systems design. *International Journal of Human-Computer Studies*, *50*, 213–241.

Sutcliffe, A. G., Gault, B., & Maiden, N. A. M. (2005). ISRE: Immersive scenario-based requirements engineering with virtual prototypes. *Requirements Engineering*, *10*, 95–111.

Sutcliffe, A. G., & Maiden, N. A. M. (1992). Analysing the novice analyst: Cognitive models in software engineering. *International Journal of Man-Machine Studies, 36,* 719–740.

Sutcliffe, A. G., Maiden, N. A. M., Minocha, S., & Manuel, D. (1998). Supporting scenario-based requirements engineering. *IEEE Transactions on Software Engineering, 24,* 1072–1088.

Sutcliffe, A. G., & Ryan, M. (1997). Assessing the usability and efficiency of design rationale. In S. Howard, J. Hammond, & G. Lindgaard (Eds.), *Proceedings: Human Computer Interaction INTERACT-97* (pp. 148–155). London: Chapman and Hall.

Sutcliffe, A. G., Thew, S., Venters, C., De Bruijn, O., NcNaught, J., Proctor, R., & Buchan, I. (2007). ADVISES project: Scenario-based requirements analysis for e-science applications. In *Proceedings of UK All Hands Conference on e-Science, Nottingham.* Available on CD-ROM.

Thew, S., Sutcliffe, A. G., De Bruijn, O., McNaught, J., Procter, R., Venters, C., & Buchan, I. (2008). Experience in e-science requirements engineering. In *Proceedings: 16th IEEE International Requirements Engineering Conference* (pp. 277–282). Los Alamitos CA, USA: IEEE Computer Society Press.

Thew, S., Sutcliffe, A. G., Procter, R., De Bruijn, O., McNaught, J., Venters, C., & Buchan, I. (2009). Requirements engineering for e-science: Experiences in epidemiology. *IEEE Software, 26*(1), 80-87.

Toulmin, S. (1958). *The uses of argument*. Cambridge, UK: Cambridge University Press.

Yu, E. S. K. (1993). Modelling organisations for information systems requirements engineering. In S. Fickas & A. C. W. Finkelstein (Eds.), *Proceedings: 1st International Symposium on Requirements Engineering* (pp. 34–41). Los Alamitos, CA, USA: IEEE Computer Society Press.

Author's Note

All correspondence should be addressed to:
Alistair Sutcliffe
Manchester Business School
University of Manchester
Booth Street East
Manchester M15 6PB, UK
ags@manchester.ac.uk

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

HUMAN TECHNOLOGY

An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889

www.humantechnology.jyu.fi Volume 6(1), May 2010, 55–70

PROMOTING GROUP CREATIVITY IN UPSTREAM REQUIREMENTS ENGINEERING

Rosalie J. Ocker
College of Information Sciences and Technology
The Pennsylvania State University, USA

Abstract*: The upstream stage of requirements engineering (RE) focuses primarily on determining high-level organizational requirements. Upstream RE provides perhaps the best opportunity to instill creativity into the design process, since it is where stakeholders figure out what to build. However, exactly how to incorporate creativity into current RE methods remains a fundamental concern. Negative social influences, such as those associated with status differentials, ingroup bias, and majority influence, can impede group creativity and otherwise negatively impact the upstream RE process. This paper discusses these issues. Two IBIS-based design rationale approaches are presented and suggestions for diminishing the potential for negative social influences are offered.*

Keywords: *Social influence, creativity, design rationale, requirements engineering.*

UPSTREAM REQUIREMENTS ENGINEERING

The upstream stage of requirements engineering (RE) focuses primarily on determining high-level organizational requirements. The process begins with an often ill-defined, unstructured problem and works towards a feasible problem definition and then to a set of high-level requirements. Determining upstream requirements is typically an intensive collaborative process of communication and negotiation (Holtzblatt & Beyer, 1995) among heterogeneous stakeholders, consisting of users, designers, project sponsors and other effected parties. Each stakeholder group brings its unique perspective to this process; thus, knowledge acquisition, sharing, and integration must be accomplished to develop a mutually shared understanding (Waltz, Elam, & Curtis, 1993).

Upstream RE provides perhaps the best opportunity to instill creativity into the design process (Couger, 1996) since it is where stakeholders figure out what to build. However, exactly how to incorporate creativity into current RE methods remains a fundamental concern since current methods rarely include processes to encourage creativity (Nguyen & Swatman, 2006). Furthermore, reaching a stage of shared understanding and eliciting high-level requirements can be laden with negative intergroup social processes, such as status differentials, ingroup bias, and majority influence at the expense of minority influence (Ocker, 2007a). These social influences can thwart creativity and otherwise negatively impact the upstream RE process (Ocker, 2005, 2007b).

URN:NBN:fi:jyu-20105241907

This paper seeks to offer insight into how creativity can be encouraged during upstream RE by addressing and diminishing negative social influences between (and within) stakeholder groups. This paper is organized as follows: The next section reviews individual and group-level creativity and discusses how social influences impact creativity. Then group support systems and electronic brainstorming are discussed within the context of social influences. Finally, the IBIS approach to design rationale is discussed. The paper ends with suggestions for integrating Group Support Systems (GSS) with anonymous electronic brainstorming and anonymous voting into two IBIS based DR approaches.

CREATIVITY AND SOCIAL INFLUENCES

Creativity is a complex interaction of person and situation that takes places at both the individual and group levels. Creativity at the individual level is a function of antecedent conditions (e.g., the home environment), personality, knowledge about the task, motivation, and cognitive style/abilities (Amabile, 1988, 1990; Barron & Harrington, 1981; Carrol, 1985; Guilford, 1977). Concerning styles/abilities, a substantial body of research has focused on the divergent production of ideas as the dominant cognitive link to creativity. Divergent thinking progresses away from a problem in a variety of different directions and involves breaking down barriers and restrictions on thoughts. Convergent thinking, on the other hand, involves progression towards a single answer (Thompson, 2003). The cognitive processes of fluency, flexibility, originality, and elaboration have been identified as essential to the divergent production of ideas (Guilford, 1984). Personality traits associated with creativity include independent thought and judgment, autonomy, persistence, self-confidence, intellectual honesty, and an internal locus of control (e.g., Amabile, 1988; Barron & Harrington, 1981; Woodman & Schoenfeldt, 1989).

Creativity at the team level is more likely to occur when the composition of the team includes "stimulating colleagues" (Parmeter & Gaber, 1971). Heterogeneous teams composed of individuals who bring a range of knowledge, ideas, and approaches to problem solving improve the teams' creative performance (Hoffman, 1959; Hoffman & Maier, 1961). Diversity in terms of areas of specialization and work responsibilities are especially relevant to enhanced team creativity.

West (1990) proposed that creative teams operate in an environment of participative safety and foster a climate for excellence. Collaboration that occurs in a nonjudgmental and supportive team atmosphere engenders a feeling of interpersonal safety among participants. West reasoned that this nonthreatening atmosphere promotes creativity because members are more likely to risk proposing new ideas.

A climate for excellence refers to a team atmosphere where a mutually shared concern for performance excellence pertaining to a vision or outcome thrives. A tolerance for diversity of opinion and constructive conflict are the hallmarks of this climate; opposing opinions are not only offered, but also are debated and critiqued by team members (King & Anderson, 1990).

Social Influences

The qualities and characteristics associated with group creativity are influenced by the social conditions and context in which the creative situation occurs. Group-level creativity is impacted by a number of factors that come into play when individuals collaborate. These include the member composition of the group, characteristics of the group, such as the degree of trust and cohesiveness, and the group collaboration context (e.g., degree of virtuality or physical distance between group members and means of communication; Woodman, Sawyer, & Griffin, 1993). Interaction between individuals and groups are impacted by social influences. Woodman et al. argue that social influences stemming from cross-level interactions between individuals and groups are critical to understanding the enhancers and inhibitors impacting creativity at the group level.

The pervasiveness of social influences within a group is described by Vinacke, Wilson, and Meredith (1964):

> In a very real sense, any interaction between or among persons can be viewed as a social influence process. It would be hard, certainly, to think of a social setting in which at least one person is not attempting to bring about some desired response in another. Even ordinary conversations have this characteristic. (p. 259)

A discussion of the social influences that have the potential for impact on a group's creativity is presented next. Specifically, status, social identity and ingroup bias, majority influence, and minority influence are presented. Table 1 contains a description of each social influence and its potential impact on creativity.

Status

A status characteristic is any characteristic that influences a group member's own or others' evaluations and beliefs about that group member. As delineated by Cohen and Zhou (1991),

> Status characteristics can be "diffuse" (i.e., hold over a wide range of situations and performances), or be "specific", (i.e., limited to a particular situation, or task). Status characteristics may be external to the interaction or may emerge in the course of task interaction; they may be explicitly relevant to the group task or they may become relevant in the course of interaction. Gender, race, and military rank are examples of diffuse characteristics that are external to the group interaction. Mathematical ability is a specific status characteristic that is explicitly relevant to solving mathematical problems and may become relevant to a whole range of verbal and nonverbal tasks. (p. 180)

Table 1. Social Influences and their Potential Effect on Creativity.

Social Influence	Description	Potential Creativity Effect
Status	Status hierarchies result in inequalities in interaction; higher status members have more influence than lower status members.	Reduces
Social Identity	Members categorize themselves into "us vs. them" subgroups. Subgroups develop separate identities leading to ingroup bias (increased interaction with and preferential behavior towards members of one's subgroup).	Reduces
Majority Influence	Majority opinion-holders influence minority opinion-holders who re-caste their views to conform to majority; promotes convergent thinking.	Reduces
Minority Influence	Minority opinion-holders influence majority opinion-holders; promotes independent and divergent thinking.	Increases

Status characteristics theory (SCT) suggests that individuals *combine* status information of group members to form expectations of their collective performance (see Wagner & Berger, 1993, 1997, for summaries). In this way, status hierarchies are formed within a given group, which result in inequalities in interaction such that higher status individuals initiate and receive more interaction and have more influence than lower status members. For low-status members to attain some level of influence, they must show more evidence of ability than high-status members (Biernat & Kobrynowicz, 1997).

SCT has relevance for the composition of a given group. It is not the status of the individuals within a group, per se, that organizes member interaction. Rather, it is the composition of the group with regard to the status differentials between members (Sell, Lovaglia, & Mannix, 1992). Moreover, the more divergence between the states of a status characteristic (e.g., a team of four males and a female is likely to be more divergent than a team of two males and a female), the more impact the status characteristic has on group interaction (Kanter, 1977; Moreland & Levine, 1992).

A structural approach also has been used to account for behavior due to status differences. The theory of proportional representation posits that the numerical representation of a status type (e.g., race, sex)—that is, the relative numbers of a given status indicator—influences interaction (Kanter, 1977). According to Kanter, those in the numerical majority control the group and its culture. Skewed groups are those whose membership has a preponderance of one status type over another. In skewed groups, a member from the nondominant category may experience feelings of isolation and powerlessness. This may lead to behavior by the nondominant members that tends towards passive and inhibited conduct.

Social Identity and Ingroup Bias

Social categorization theory (Tajfel, 1981) and social identity theory (SIT; Tajfel, 1978; Tajfel & Turner, 1986; Turner, 1981) suggest that people derive social identity primarily from membership in groups (not to be confused with team membership). For example, demographic differences can result in people categorizing themselves into "us vs. them" groupings. In such situations, positive social identity results when one can make favorable comparisons between the group to which one is a perceived member (i.e., the ingroup), as compared to other germane groups to which one is not a perceived member (i.e., the outgroups).

Decades of research indicates that subgroups form due to diversity in terms of demographic attributes (e.g., race, age, sex), psychological differences (e.g., beliefs), and affiliations (for a comprehensive review, see Williams & O'Reilly, 1998). In team dynamics, subgroups develop separate identities and exhibit ingroup bias—that is, increased interaction with and preferential behavior towards members of one's subgroup, reduced trust and team cohesiveness, and increased conflict between subgroups—which impairs team effectiveness and performance (e.g., Lott & Lott, 1965; O'Reilly, Caldwell, & Barnett, 1989; Smith et al., 1994).

A fault line divides a group's members according to one or more attributes (Lau & Murnighan, 1998), as depicted in Figure 1. The more attributes that are aligned along the same fault line, the stronger the fault line, and the resulting distinction between subgroups. For example, if a team is composed of male engineers and female marketing professionals, it has a stronger fault line than if the engineering and marketing groups were composed of both males and females. Thus, rather than the amount of diversity within a team, Lau and

Murnighan (1998) argue that it is the alignment or correlation (Cramton & Hinds, 2005) of member attributes that increases the strength of the division between subgroups.

The configuration of a team also has been shown to create a fault line. For example, when team members are spread across multiple locations, subgroups tend to form according to location, resulting in ingroup bias (Ocker, Huang, Benbunan-Fich, & Hiltz, in press; Panteli & Davison, 2005; Polzer, Crisp, Jarvenpaa, & Kim, 2006). The number of team locations can affect the degree of ingroup bias: For example, teams configured across two locations have been found to exhibit stronger ingroup dynamics compared to teams with three locations (Polzer et al., 2006). Additionally, when a team includes both colocated members and isolated members, ingroup dynamics can still prevail (Polzer et al., 2006). Bos, Shami, Olson, Cheshin, and Nan (2004) found that colocated members formed one subgroup, while the isolates banded together to form their own subgroup.

Majority Influence

Groups have a need for uniformity of opinion. Moscovici (1974) asserts that this is due to two primary reasons. First, since groups normally have a purpose, the group feels the need to move in a certain direction to achieve that purpose, which is much easier to accomplish when group members hold similar opinions. Second, groups have a need for a sense of social reality, which is achieved through the validation of one's own judgments and opinions by the other members of the group. To achieve uniformity, groups typically exclude deviance and are unwilling to compromise (Asch, 1951; Festinger, 1950, Sherif, 1935).

Majority influence is a type of social influence centered on conformity, which entails movement in beliefs and behavior toward the group. The act of conforming requires two parties: the majority group and the minority individual or subgroup. The majority has its own set of beliefs and definitions for acceptable behavior—in essence, its rules and norms. Cohesion within the group reaffirms the belief and acceptance of previously made decisions, and prohibits the acceptance or adherence to other norms. Conformity within the group serves to absorb any deviance by the minority, as deviance is seen as a threat to the majority. The function of conformity is successfully fulfilled when (a) the majority of the group has a well-defined set of norms, responses, and attitudes, and (b) the group exerts social pressure on the individual or subgroup that lacks well-established norms. Majority influence prevails when deviant individuals or subgroups recast their views or behaviors to conform to those of the group (Moscovici, 1974).

In general, research has found that what contributes most to conformity is the existence of unanimous agreement (e.g., Graham, 1962; Mouton, Blake, & Olmstead, 1956). Thus, yielding to the majority, although influenced by various factors such as size or shared power of the majority, is credited to the primary influence that a *perceived consistency* of the majority opinion has on

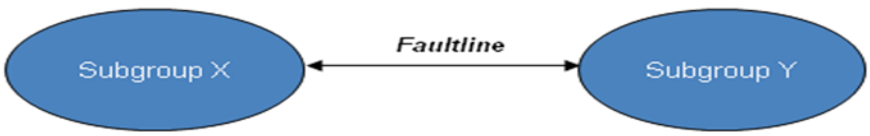

Figure 1. Subgroups divided by a fault line.

the minority. It is this attribute that is believed to cause the minority opinion holder to succumb to the majority opinion.

Normalization is the process whereby the "reciprocal influence of group members induces them to formulate or to accept a compromise" (Moscovici, 1974, p. 208). Individuals attempt to create an internal reference point—a norm or standard—when an external norm or standard is lacking (Sherif, 1935). In considering the case in which the majority of individuals do not have a well-defined norm or solution, Moscovici (1974) described the process of normalization:

> When a number of individuals are confronted by a set of objects or stimuli which they are supposed to judge or a problem which they are supposed to solve and there are no particular norms or rules to govern their behavior, each of the individuals is hesitant and relatively inconsistent. As soon as they begin discussing the situation or making their judgments, each of them becomes aware of the discrepancies among themselves. Since they have no motivation to increase their uncertainty by widening their differences, nor to provoke conflict between themselves, they generally follow the road of compromise. This compromise generally leads to the establishment of an average judgment or response. This is what is called the normalization process. (p. 224).

The development of a norm or standard is due to the need for stability that is provided by a frame of reference within which responses can be organized. Moscovici asserted that the normalization process occurs in order to avoid conflict and disagreement, and therefore is not necessarily a result of cooperation and mutual understanding within the group. Again, convergent, rather than divergent, thinking is prevalent.

Minority Influence

Minority influence describes the situation where minority opinion holders influence the opinions of the majority opinion holders. Thus, the situation is similar to that of majority influence, although in this case the minority exerts influence on the majority. Conflict and behavioral style are important aspects in the development of minority influence on a group. Moscovici and Nemeth (1974, p. 220) asserted that it is the behavioral style, that is, the "orchestration and patterning of the minority's behavior" that is at the root of the minority's influence. They explained,

> It is such behavioral styles that cause the majority to question its own position and consider the possibility that the minority may be correct. When such patterning leads to assumptions that the minority is consistent and certain of its position, that it is objective and unbiased in its judgments, then the minority can be effective. (p. 220)

Maass and Clark (1984) pointed out that,

> Since Sherif's (1935) and Asch's (1951) early work on conformity, it has become a social psychological truism that individuals tend to yield to a majority position even when that position is clearly incorrect. Conformity became a term nearly equivalent in meaning to social influence. It was not until 1969 that Moscovici and his coauthors pointed out that social influence is by no means limited to a one-direction dependency of the minority on the majority... a consistent minority is able to exert a remarkable degree

> of influence even when it is not equipped with such characteristics as power, status, competence (Hollander, 1964). (p. 428)

Nemeth (1986) made a connection between creativity and minority influence. In a series of studies, Nemeth and colleagues found that minority influence stimulated independent and divergent thought, so that issues and problems were considered from more perspectives. This resulted in group members detecting and exploring not only new solutions, but correct solutions. Nemeth explained the process as follows:

> When the influence source is a minority, the assumption that the minority is incorrect and the disinclination to publicly adopt its position lead to an initial dismissal of the minority viewpoint. However, with consistency and confidence on the minority's part over time, people are stimulated to understand such alternative views (e.g. "How can they be so wrong and yet so sure of themselves?"). As a result, they are stimulated to reappraise the entire situation, which involves a consideration of numerous alternatives, one of which is the position proposed by the minority. As such, the thought processes are marked by divergence and, hence, the potential for detecting novel solutions or decisions. (Nemeth, 1986, p. 26)

Nemeth thus offered a reconceptualization of minority influence. "The implications for creativity and decision making, both at the individual and group levels, become considerable" (Nemeth, 1986, p. 25).

Group Support Systems and Brainstorming

Can technology assist in overcoming the negative social influences that can thwart creativity in groups? A recent study indicates the usefulness of a group support system (GSS) in workgroups with distinct social identities (Lim & Guo, 2008). A GSS incorporates computer technology with communication and decision processes in support of group problem-solving and decision-making activities. Historically, GSSs were designed for same-time, same-place meetings where each meeting participant has his/her own computer monitor and keyboard (see Fjermestad & Hiltz, 1998, 2000, for in-depth reviews). In a "decision room" GSS, a meeting facilitator assists the group in all activities, from providing technical support to chairing the GSS session, and in moving the group through a planned agenda.

A GSS typically includes a toolkit to assist groups in performing their activities. A system might include a planning tool, an electronic brainstorming tool, and various voting tools. For example, GroupSystems,[1] arguably the most extensively used and studied GSS, includes a series of tools to support electronic brainstorming for groups. A comparison of traditional and electronic brainstorming provides a good example of how technology and process can alter the affects of social influences on the creative process.

Traditional Brainstorming

As originally devised, traditional brainstorming (face-to-face, no technology support) involves four rules designed to reduce negative social influences so as to increase idea generation and group creativity (Osborn, 1963). These rules are

- Focus on quantity: This rule is based on the axiom that quantity leads to quality. By increasing the number of ideas generated, it is assumed that there is a greater chance of generating a creative and effective solution.
- Withhold criticism: By withholding criticism, the expectation is that participants will be more likely to submit far-fetched, radical, and even possibly "silly" but potentially stimulating ideas to the group.
- Welcome unusual ideas: To encourage "out of the box" thinking, participants are encouraged to forego assumptions and look at the problem from new perspectives.
- Combine and improve ideas: Ideas can be merged to form new, "better" ideas, following the maxim that 1+1=3.

Osborn's claims that traditional face-to-face brainstorming groups produce more and better ideas than the same number of people working alone have been refuted time and again (see Mullen, Johnson, & Salas, 1991, for a review). Two explanations have been offered regarding this phenomenon (Diehl & Stroebe, 1987). *Production blocking* occurs when participants must wait to convey their ideas to the group because another member is speaking. During this waiting period, it is speculated that the silent participants forget their ideas or self-censor, resulting in the loss of a significant number of their ideas. *Evaluation apprehension* stems from a fear of disapproval by others (Cottrell, 1972) that results in participants holding back ideas. *Social loafing*, also known as free-riding, is the tendency of participants to put forth less effort in group settings than they would if working individually (Latane, 1981).

Electronic Brainstorming Using GSS

Using a GSS, electronic brainstorming (EBS) attempts to address the shortcomings of traditional brainstorming by blending a component of the nominal group technique (the ability to generate ideas without interruption) with a component of traditional brainstorming (sharing ideas with other participants). The typical EBS process includes steps for generating ideas, editing ideas, and evaluating ideas in a decision-room type GSS context (Gallupe & Cooper, 1993). Anonymous EBS is a variation that, as the name implies, eliminates the association between a participant and his/her ideas, edits, and evaluations.

When generating ideas, members type an idea into the GSS and hit enter, at which point the idea is disseminated to the group. Members receive others' ideas randomly. In terms of editing ideas, the GSS includes a sort feature that enables members to organize ideas by keywords, and then combine ideas or delete redundant ideas. The evaluation of ideas is typically accomplished by voting. In rank order voting, each participant can individually prioritize the idea list. The GSS then takes these individual rankings and creates a group ranking based on all members' rankings. Any number of votes can occur in an effort to reach consensus on the priority of ideas.

Parallel entry and anonymity are important in addressing the limitations of traditional brainstorming (Connolly, Jessup, & Valacich, 1990). Production blocking is addressed through parallel entry, the ability of participants to simultaneously enter and share ideas. In anonymous EBS, evaluation apprehension and social loafing are reduced because participants share and evaluate ideas without being identified, free from the fear of criticism (given a large-enough pool of participants). Thus, anonymous EBS separates ideas from the status of their proposers and, as such, promotes equalized power within the group. Furthermore, the opportunity for minority influence is potentially enhanced, not only by separating ideas from proposers, but also by increasing the opportunity for the minority to be "heard," since there is no opportunity for the majority opinion holders to shut out the minority.

Design Rationale: IBIS to Support Argumentation

Traditionally, design rationale (DR) has been applied to RE since it epitomizes the "wicked" types of problems addressed by Rittel and Weber (1973). DR provides a structure for focusing discussion among the different stakeholders within a project team. DR originates from two areas: early studies of design activity conducted by Kunz & Rittel (1970) and argumentation as developed by Toulmin (1958). However, Rittel was the first to advocate systematic documentation of DR as part of the design process.

Rittel's approach to design reasoning is based on argumentation, and thus is concerned with supporting debate and discussion. Rittel developed a method to represent (a) a network of issues (design questions); (b) selected and reflected answers; and (c) arguments for and against these answers. The outcome of his work was IBIS (Issue-Based Information System) that is a method, not a computerized information system, for supporting the reasoning process required in design and other wicked problems.

The objective of an IBIS discussion is for each of the stakeholders to try to understand the specific elements of each other's proposals. Initially, an unstructured problem area or topic is presented.

> About this topic and its subtopics a *discourse* develops. *Issues* are brought up and disputed because different positions are assumed. *Arguments* are constructed in defense or against the different positions until the issue is settled by convincing the opponents or decided by a formal decision procedure. (Kunz & Rittel, 1970, pp. 1-2, emphasis in original).

Thus, the discussion unfolds as one tries to persuade others of one's point of view.

The gIBIS was a software platform used to conduct research on using hypertext, GSS, and rhetorical models to facilitate and capture software system design decisions and their rationale. gIBIS was a prototype software tool for building and browsing IBIS networks (Conklin & Begeman, 1988; Conklin & Yakemovic, 1991). It provided a graphical interface and had a limited GSS capability, allowing several users to contribute to an IBIS discussion synchronously.

The IBIS method makes it "harder for discussants to make unconstructive rhetorical moves, such as 'argument by repetition' and name calling, and it supports other more constructive moves, such as seeking the central issue, asking questions as much as giving answers, and being specific about the supporting evidence of one's viewpoint" (Conklin & Begeman, 1988, p. 305). Especially relevant in addressing social influences, users of the IBIS method report that the structure that IBIS imposes on discussions served to expose "axe grinding, hand waving, and clever rhetoric" and that they valued the tendency for assumptions and definitions to be made explicit (Conklin & Begeman, 1988, p.323).

The semistructured nature of IBIS accounts for some of these advantages (Malone, Grant, Lai, Rao, & Rosenblitt, 1986). The IBIS structure does not place any constraint on the writer when it comes to expressibility. At the same time, the reader is provided with the recurrent structure in the textual material that aids both search and comprehension (Conklin & Begeman, 1988).

As with other DR, an IBIS-based approach can be primarily descriptive or prescriptive; some are a combination of both (Dutoit et al., 2006). Descriptive approaches aim to portray designers' thinking processes and emphasize the issue base as a history of the design process; they do not seek to modify designers' reasoning. In contrast, prescriptive approaches seek to improve the design process by improving the reasoning of designers. This is done through a prescribed process to be followed, as the issue-base structure is developed through debate and discussion.

Two Examples of IBIS-based DR Approaches

In this section, two examples of IBIS-based DR approaches are presented. Each approach is supported by computerized tools.

Wisdom Approach

Wisdom is both a prescriptive approach and a tool that is intended for use by project stakeholders during the early problem definition stage of RE (MacKenzie et al., 2005; Rooksby, Sommerville, & Pidd, 2006). Rather than leading to finalized requirements, the goal of Wisdom is to achieve a common understanding of the problem to be addressed before moving forward. The prescribed process, described below and highlighted in Table 2, incorporates two problem-structuring techniques: cognitive mapping and dialogue mapping using IBIS.

Brainstorming, the first step in the process, is used to encourage divergent thinking about relevant issues and concerns in broad terms (see previous discussion on Brainstorming for a description of the process steps). If a form of nominal group technique is desired, then each participant brainstorms individually by typing their ideas into their computer whereby only the facilitator and the system chauffeur can identify the contributor. Otherwise, a cooperative (nonanonymous) brainstorming technique is used.

Cognitive mapping (Tolman, 1948) is the second step in the process and provides a macro view of the problems. A cognitive map is a representation of how an individual views or thinks about a situation and, as such, can be viewed by others. It includes nodes that represent concepts and directional arcs that indicate linkages and causality between concepts. In the Wisdom approach, the facilitator creates a map that identifies participants' concerns and issues. The goal is for the group to identify key issues and to gain agreement, which normally

Table 2. Steps in the Wisdom Approach.

Step	Description
(1) Brainstorming	An anonymous or co-operative (non-anonymous) brainstorming technique is used to encourage divergent thinking about relevant issues and concerns in broad terms.
(2) Cognitive mapping	The facilitator creates a cognitive map that identifies participants' concerns and issues. The goal is for the group to identify key issues and to gain agreement, which requires debate and possible voting in the negotiation process.
(3) Dialogue mapping	Using IBIS notation, participants create a dialogue map for each key issue. Maps are linked using hypertext.

requires debate and negotiation, and possibly voting. The result of the cognitive mapping step is to reach agreement on and commitment to a way forward.

Dialogue mapping is the third step and is used after key issues have been identified during cognitive mapping. Using IBIS notation—specifically question, idea, and argument node types—participants create a dialogue map for each key issue, using hypertext to link them as appropriate. Dialogue mapping provides a micro view of the issues and promotes "rigorous discussion and analysis of individual issues" (Rooksby et al., 2006, p. 142)

The tool consists of an application that runs on the facilitator's laptop. The tool provides network connectivity that supports individual brainstorming during a meeting. The facilitator creates the map, which is projected onto a shared display. Thus, participants' use of the tool is indirect. A Web interface supports asynchronous work prior to the meeting.

The Wisdom designers (Rooksby et al., 2006) describe the importance of the facilitator:

> The effectiveness of a meeting is dependent on the skills of a neutral facilitator [9, 30]. The facilitator's objective is to foster procedural rationality, where stakeholders agree that sensible decisions have been made and commit to them. In practice, a facilitator ensures that a meeting remains focused, that the evolving cognitive map accurately reflects the ongoing discussion, that stakeholders get the opportunity to air their views and that the decision process is sensible. (p. 142-143).

WinWin Approach

WinWin is an example of a descriptive approach to design rationale. It is a "set of principles, practices, and tools, which enable a set of interdependent stakeholders to work out a mutually satisfactory 'win-win' set of shared commitments" (Boehm & Kitapci, 2006, p.78). WinWin is built on the spiral model, which combines the features of the prototyping and waterfall models (Boehm, 1988). With the intent of risk management, the spiral model is an incremental development methodology that consists of a series of cycles. Each spiral cycle consists of four phases: determine objectives, identify and resolve risks, develop and test, and plan the next iteration.

WinWin adds a negotiation process to the front end of each spiral cycle. The negotiation activities consist of (a) identifying "success-critical" stakeholders, (b) eliciting the stakeholders' "win" conditions, (c) negotiating mutually satisfying win-win conditions between stakeholders, and (d) monitoring and control of a win-win balance during development.

> Stakeholders express their goals as win conditions. If everyone concurs, the win conditions become agreements. When stakeholders do not concur, they identify their conflicted win conditions and register their conflicts as issues. In this case, stakeholders invent options for mutual gain and explore the option trade-offs. Options are iterated and turned into agreements when all stakeholders concur. (Boehm & Kitapci, 2006, p. 180)

The EasyWinWin tool embeds the WinWin negotiation process within a GSS (i.e., GroupSystems). As indicated in Table 3, the negotiation process includes steps where stakeholders (a) review and expand negotiation topics, (b) brainstorm, (c) converge on win conditions, (d) prioritize win conditions, (e) reveal issues and constraints, (f) identify issues and options, and (g) negotiate agreements.

Table 3. EasyWinWin Negotiations Steps.

Step	Description
Review & expand negotiation topics	Jointly build an outline of topics
Brainstorm	Share goals, perspectives, expectations
Converge on win conditions	Discuss ideas from brainstorming session to jointly develop list of win conditions
Prioritize win conditions	Vote on win conditions to determine priorities
Reveal issues and constraints	Surface and understand issues
Identify issues and options	Surface issues due to constraints, risks, uncertainties, and conflicting win conditions
Negotiate agreements	Establish mutual commitments by formulating win conditions

The designers (Boehm & Kitapci, 2006) state that,

> The focus on consensus leads to a higher acceptance of decisions and to an increased mutual understanding among the involved parties. The evaluation of the WinWin model shows that the use of an issue model of negotiation support enhances trust and shared understanding among shareholders, even in the presence of uncertainties and changing requirements. (p. 187)

CRITIQUE AND CONCLUSIONS

I conclude by critiquing the Wisdom and WinWin approaches in terms of addressing social influences and offer suggestions for enhancing the approaches in order to explicitly address social influences to promote creativity. Both Wisdom and WinWin recognize the importance of including key stakeholders in order to identify requirements and reach agreement on substantive issues before moving a project forward. Both include brainstorming to generate ideas, issue surfacing and representation as well as negotiation to reach agreement.

West (1990) proposed that creative teams operate in an environment of participative safety (a nonjudgmental and supportive team atmosphere) and foster a climate for excellence (exhibiting a tolerance for diversity of opinion and constructive conflict). However, the manner in which stakeholders are included in each approach tends toward an ideal view of group interaction (i.e., all participants' viewpoints are encouraged, and all participants feel equally valued and willing to share diverse or controversial viewpoints). The Wisdom and WinWin approaches both rely heavily on meeting facilitators to create and maintain a supportive atmosphere. However, expert facilitators can be scarce. In their absence, processes may not be carried out as their designers intended.

I suggest that a stakeholder analysis be conducted as an initial step. This analysis should result in an understanding of pre-existing relationships between stakeholder groups and identify existing or potentially conflicting interactions between key stakeholder groups. Given the outcome of the stakeholder analysis, and a tool such as EasyWinWin, anonymity can be incorporated throughout the remaining steps (e.g., anonymous EBS, anonymous voting, and anonymous issue surfacing and representation are all likely candidates). Thus, the potential for negative social influences (i.e., status differentials, social identity, and majority influence), all of which breed conformity as opposed to creativity, can be explicitly addressed.

Furthermore, by embedding steps that address social influences, less dependence is placed on the skills of a facilitator and the benevolence of stakeholder groups. The result is that negative social influences have the *potential* to be side-stepped. Thus, a group context conducive to a more egalitarian and participative exchange of ideas is promoted, which in turn, is more favorable to group creativity.

In conclusion, there is a need to instill awareness of the potential for negative social influences throughout the RE process. Negative social influences should be addressed in a deliberate manner in order to promote creativity in the requirements engineering process.

ENDNOTE

1. For more information, see GroupSystems.com

REFERENCES

Amabile, T. M. (1988). A model of creativity and innovation in organizations. *Research in Organizational Behavior, 10*, 123–167.

Amabile, T. M. (1990). Within you, without you: The social psychology of creativity, and beyond. In M. A. Runco & R. S. Albert (Eds.), *Theories of creativity* (pp. 61–91). Newbury Park, CA, USA: Sage.

Asch, S. E. (1951). Effects of group pressure upon the modification an distortion of judgments. In H. Guetzkow (Ed.), *Groups, leadership, and men* (pp. 177–190). Pittsburgh, PA, USA: Carnegie Press.

Barron, F. B., & Harrington, D. M. (1981). Creativity, intelligence, and personality. *Annual Review of Psychology*, *32*, 439–476.

Biernat, M., & Kobrynowicz, D. (1997). Gender and race-based standards of competence: Lower minimum standards but higher ability standards for devalued groups. *Journal of Personality and Social Psychology, 72*, 544–557.

Boehm, B. (1988). A spiral model of software development and enhancement. *Computer*, *21*(5), 61–72.

Boehm, B., & Kitapci, H. (2006). The WinWin approach: Using a requirements negotiation tool for rationale capture and use. In A. H. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 173–190). Berlin, Germany: Springer-Verlag.

Bos, N., Shami, N. S., Olson, J. S., Cheshin, A., & Nan, N. (2004). In-group/out-group effects in distributed teams: An experimental simulation. *Computer Supported Cooperative Work* (*CSCW)*, *6*, 429–436.

Carroll, J. B. (1993). *Human cognitive abilities*. New York, NY: Cambridge University Press.

Cohen, B. P., & Zhou, X. (1991). Status processes in groups. *American Sociological Review*, 56, 179–188.

Conklin, J., & Begeman, M. (1988). gIBIS, A tool for exploratory policy discussion. ACM *Transactions on Office Information Systems (TOOIS),* 6, 303-331. Conklin, J., & Yakemovic, K. C. (1991). Process-oriented design rationale. *Human-Computer Interaction*, 357–391.

Connolly, T., Routhieaux, R. L., & Schneider, S. K. (1993). On the effectiveness of group brainstorming. *Small Group Research*, *24*, 490–503.

Connolly, T., Jessup, L. M., & Valacich, J. S. (1990), Effects of anonymity and evaluative tone on idea generation in computer-mediated groups. *Management Science*, 36, 689–703.

Cottrell, N. B. (1972). Social facilitation. In C. McClintock (Ed.), *Experimental social psychology* (pp. 185–236). New York: Holt, Rinehart & Winston.

Couger, J. D. (1996). *Creativity & innovation in information systems organizations,* Danvers, MA, USA: Boyd & Fraser.

Cramton, C. D., & Hinds, P. J. (2005). Subgroup dynamics in internationally distributed teams: Ethnocentrism or cross-national learning. *Research in Organizational Behavior*, *26*, 231–263.

Diehl, M., & Stroebe, W. (1987). Productivity loss in brainstorming groups: Toward the solution of a riddle. *Journal of Personality and Social Psychology*, *53*, 497–509.

Dutoit, A. H. (2006). Rationale management for requirements engineering.and In A. H. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 132–136). Berlin, Germany: Springer-Verlag.

Dutoit, A.H., McCall, ., Mistrik, I, & Paech, B. (2006). Rationale management in software engineering: Concepts and Techniques. In A. H. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 1–48). Berlin, Germany: Springer-Verlag.

Festinger, L. (1950). Informal social communication, *Psychological Review*, *57*, 271–282.

Fischer, G. (1990). Cooperative knowledge-based design environments for the design, use and maintenance of software. In *Proceedings of the Software Symposium 90* (pp. 2–22). Kyoto, Japan.

Fjermestad, J., & Hiltz, S. (1998). An assessment of group support systems experimental research: methodology and results. *Journal of Management Information Systems*, *15*(3), 7–149.

Fjermestad, J., & Hiltz, S. (2000). Group support systems: A descriptive evaluation of case and field studies source**.** *Journal of Management Information Systems*, *17*(3), 115–159.

Gallupe, R. B., & Cooper, W. H. (1993). Brainstorming electronically. *Sloan Management Review, 35*, 27–36.

Graham, D. (1962). Experimental studies of social influence in simple judgment situations. *Journal of Social Psychology, 56*, 245–269.

Guilford, J. P. (1977). *Way beyond the IQ*. Amherst, MA, USA: Creative Education Foundation Press.

Guilford, J. P. (1984). Varieties of divergent production. *Journal of Creative Behavior*, *18*, 1–10.

Hoffman, L. R. (1959). Homogeneity of member personality and its effect on group problem-solving. *Journal of Abnormal Social Psychology*, *58*, 27–32.

Hoffman, L. R., & Maier, N. R. F. (1961). Quality and acceptance of problem solutions by members of homogeneous and heterogeneous groups. *Journal of Abnormal Social Psychology, 62*, 401–407.

Hollander, E. P. (1964). *Leader, groups, and influence.* New York: Oxford University Press.

Holtzblatt, K., & Beyer, H. R. (1995). Requirements gathering: The human factor. *Communications of the ACM*, *38*(5), 31–32.

Kanter, R. M. (1977). Some effects of proportions in group life: Skewed sex ratios and responses to numerical minority women. *American Journal of Sociology*, *82*, 965–990.

King N., & Anderson, N. (1990). Innovation in working groups. In M. West & J Farr (Eds.), *Innovation and creativity at work* (pp. 81–100). Chichester, UK: Wiley and Sons.

Kunz, W., & Rittel, H. (1970). *Issues as elements of information systems* (Working paper No. 131), available from the, Institute of Urban and Regional Development, University of California at Berkeley, http://iurd.berkeley.edu/sites/default/files/wp/131.pdf

Latane B. (1981). The psychology of social impact. *American Psychology*, *36*, 343–356.

Lau, D. C., & Murnighan, J. K. (1998). Demographic diversity and faultlines: The compositional dynamics of organizational groups. *Academy of Management Review*, *23*, 325–340.

Lim, J., & Guo, X. (2008). A study of group support systems and the intergroup setting. *Decision Support Systems*, *45*, 452-460.

Lott, A., & Lott, B. (1965). Group cohesiveness as interpersonal attraction: A review of relationships with antecedent and consequent variables. *Psychological Bulletin*, *64*, 259–309.

Maass, A., & Clark, R. D. (1984). Hidden impact of minorities: Fifteen years of minority influence research. *Psychological Bulletin*, *95*, 428–450.

Malone, T. W., Grant, K., Lai, K.-Y, Rao, R., & Rosenblitt, D. (1986). Semi-structured messages are surprisingly useful for computer-supported coordination. In D. Peterson (Ed.), *Proceedings of CSCW'86, MCC/ACM Conference on Computer-Supported Cooperative Work* (pp. 102–114). Austin, TX, USA: ACM Press.

MacKenzie, A., Pidd, M., Rooksby, J., Sommerville, I., Warren, I., & Westcombe, M. (2005).Wisdom: Decision support and paradigms of decision making. *European Journal of Operational Research*, *170*, 156–171.

Moreland, R. L., & Levine, J. M. (1992). The composition of groups. In E. J. Lawler, B. Markovsky, C. Ridgeway, & H. A. Walker (Eds.), *Advances in group processes (*Vol. 9; pp. 237–280). Greenwich, CT, USA: JAI.

Moscovici, S. (1974). Social influence, I: Conformity and social control. In C. Nemeth (Ed.), *Social psychology:Classic and contemporary integrations* (pp. 179–216). Chicago: Rand McNally.

Moscovici, S., & Nemeth, C. (1974). Social influence, II: Minority influence. In C. Nemeth (Ed.), *Social psychology: Classic and contemporary integrations* (pp. 217–249). Chicago: Rand McNally.
Mouton, J. S., Blake, R. R., & Olmstead, J. A. (1956). The relationship between frequency of yielding and the disclosure of personal identity. *Journal of Personality, 24*, 339–347.
Mullen, B., Johnson, C., & Salas, E. (1991). Productivity loss in brainstorming groups: A meta-analytic integration. *Basic and Applied Social Psychology*, *12*(1), 3–23.
Nemeth, C. J. (1986). Differential contributions of majority and minority influence, *Psychological Review*, *93*, 23–32.
Nguyen, L., & Swatman, P. A. (2006). Promoting and supporting requirements engineering in creativity. In A. H. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 209–230). Berlin, Germany: Springer-Verlag.
Ocker, R. J. (2005). Influences on ceativity in asynchronous virtual teams: A qualitative analysis of experimental teams. *IEEE Transactions on Professional Communication, 48*, 22–39.
Ocker, R. J. (2007a). A balancing act: The interplay of status effects on dominance in virtual teams. *IEEE Transactions on Professional Communication, 50*, 204–218.
Ocker, R. J. (2007b). Creativity in asynchronous virtual Teams: Putting the pieces together. In T. Torrres & S. MacGregor (Eds.), *Higher creativity for virtual teams: Developing platforms for co-creation* (pp. 26–47). Hershey, PA, USA: Idea Group.
Ocker, R. J., Huang, H., Benbunan-Fich, R., & Hiltz, S.R. (in press). Leadership dynamics in partially distributed teams: An exploratory study of the effects of configuration and Distance. *Group Decision and Negotiation*.
O'Reilly, C., Caldwell, D., & Barnett, W. (1989). Work group demography, social integration, and turnover. *Administrative Science Quarterly*, *34*, 21–37.
Osborn, A. F. (1963). *Applied imagination: Principles and procedures of creative problem solving* (3rd rev. ed.). New York: Charles Scribner's Sons.
Panteli, N., & Davison, R. M. (2005). The role of subgroups in the communication patterns of global virtual teams. *IEEE Transactions on Professional Communication*, *48*, 191–200.
Parmeter, S. M., & Gaber, J. D. (1971). Creative scientists rate creativity factors. *Research Management*, *14*, 65–70.
Polzer, J. T., Crisp, C. B., Jarvenpaa, S. L., & Kim, J. W. (2006). Extending the faultline concept to geographically dispersed teams: How colocated subgroups can impair group functioning. *Academy of Management Journal*. *49*, 679–692.
Rittel, H., & Webber, M. (1973). Dilemmas in a general theory of planning. *Policy Sciences*, *4*, 155–169.
Rooksby, J., Sommerville, I., & Pidd, M. (2006). A hybrid approach to upstream requirements: IBIS and cognitive mapping. In A. H. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 137–154). Berlin, Germany: Springer-Verlag.
Sell, J., Lovaglia, M. J., Mannix, E. A., Samuelson, C. D., & Wilson, R. K. (1992). Investigating conflict, power, and status within and among groups. *Small Group Research*, *35*, 44–72.
Sherif, M. (1935). *The psychology of social norms*. New York: Harper & Row.
Smith, J. B., Weiss, S. F., Ferguson, G. J., Bolter, J. D., Lansman, M. L., & Berad, D. V. (1986). WE: A writing environment for professionals (Tech. Rep. No. 86-025). University of North Carolina at Chapel Hill, USA, Department of Computer Science.
Smith, K., Smith, K., Olian, J., Sims, H., O'Bannon, D., & Scully, J. (1994). Top management team demography and process: The role of social integration and communication. *Administrative Science Quarterly*, *39*, 412–438.
Tajfel, H. (1978). *Differentiation between social groups: Studies in the social psychology of intergroup relations*. London: Academic Press.
Tajfel, H. (1981). *Human groups and social categories: Studies in social psychology*. Cambridge, UK: Cambridge University Press.
Tajfel, H., & Turner, J. C. (1986). The social identity theory of intergroup behaviour. In S. Worchel & W. G. Austin (Eds.), *Psychology of intergroup relations* (pp. 7–24). Chicago: Nelson.
Thompson, L. (2003). Improving the creativity of organizational work groups. *Academy of Management Executive*, *17*, 96–109.
Tolman E. C. (1948). Cognitive maps in rats and men. Psychological Review, *55*,189–208. .
Toulmin, S. E. (1958). *The uses of argument*. Cambridge, UK: Cambridge University Press.

Turner, J. C. (1981). The experimental social psychology of intergroup behaviour. In J. C. Turner & H. Giles (Eds.), *Intergroup Behaviour* (pp. 66–101). Oxford, UK: Basil Blackwell.

Vinacke, W. E., Wilson, W. R., & Meredith, G. M. (1964). *Dimensions of social Psychology*. Chicago: Scott, Foresman and Co.

Wagner, D., & Berger, J. (1993). Status characteristics theory: The growth of a program in theoretical research programs. In J. Berger & M. Zeldich, Jr. (Eds.), *Studies in the growth of theory* (pp. 23–63). Stanford, CA, USA: Stanford University Press.

Wagner, D., & Berger, J. (1997). Gender and interpersonal task behaviors: Status expectation accounts. *Sociological Perspectives, 40*, 1–32.

Walz, D. B., Elam, J. J., & Curtis, B. (1993). Inside a software design team: Knowledge acquisition, sharing and integration. *Communications of the ACM*, *36*(10), 63–77.

West, M. A. (1990). The social psychology of innovation in groups. In M. West & J. Farr (Eds.), *Innovation and creativity at work* (pp. 309–333). Chichester, UK: Wiley and Sons.

Williams, K. Y., & O'Reilly, C. A. (1998). Demography and diversity in organizations: A review of 40 years of research. *Research in Organizational Behavior*, *20*, 77–140.

Woodman, R., & Schoenfeldt, L. F. (1989). Individual differences in creativity: An interactionist perspective. In J. A. Glover & C. R. Reynolds (Eds.), *Handbook of creativity* (pp. 77–92). New York: Plenum Press.

Woodman, R., Sawyer, J., & Griffin, R. (1993). Toward a theory of organizational creativity. *Academy of Management Review, 18,* 292–321.

Authors' Note

All correspondences should be addressed to
Rosalie J. Ocker
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16801
rocker@ist.psu.edu

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

HUMAN TECHNOLOGY

An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889

www.humantechnology.jyu.fi Volume 6 (1), May 2010, 71–105

THE PRACTICE LEVEL IN PARTICIPATORY DESIGN RATIONALE: STUDYING PRACTITIONER MOVES AND CHOICES

Albert M. Selvin
Verizon, USA, and
Knowledge Media Institute
The Open University
Milton Keynes, UK

Simon J. Buckingham Shum
Knowledge Media Institute
The Open University
Milton Keynes, UK

Mark Aakhus
School of Communication & Information
Rutgers, The State University of New Jersey
New Brunswick, NJ, USA

Abstract: *Most research in design rationale focuses on specific tools, methods, models, or artifacts. There has been relatively little attention to the practice level of design rationale work: the human experience of working with the tools and methods to create rationale artifacts. This paper explores a particular juncture of creativity and design rationale that is found in the special case of helping groups of people construct representations of rationale within live meetings. Such work poses challenges and requires skills different from those of individuals working alone. We describe the role of practitioners who perform caretaking and facilitative functions in collaborative or participatory design rationale sessions, and present a set of analytical tools aimed at making the practice level more visible. We locate the analysis in a theoretical framework aimed at understanding the experiential dimensions of such practice, including sensemaking, narrative, aesthetics, ethics, and improvisation.*

Keywords: *knowledge media, sensemaking, improvisation, reflective practice, aesthetics, ethics, narrative, grounded theory, facilitation, visualization.*

INTRODUCTION

There are a variety of techniques used to foster creativity in design, such as brainstorming exercises and ideation workshops. Other articles in this special issue argue for or against the notion that design rationale techniques can spur creativity in the design process. In this paper we shift focus away from creativity as something that might be evoked through the collaborative creation of a design rationale artifact, and toward the ways in which creativity can

URN:NBN:fi:jyu-20105241908

manifest itself in the act of fostering creativity and engagement with such an artifact for others. These can be creative acts on the representation, creative ways of intervening in group process, or reframing participant utterances.

Creating representations of design rationale in collaborative groups requires a set of skills similar to other forms of participatory media practice. Understanding such practices calls for an empirical approach that can illuminate the sociotechnical, as well as aesthetic and ethical, considerations involved in evoking and representing information like design rationale, argumentation, and exploratory discussion within groups of people in live meetings. Our intent is to make this practice, with its particular conditions and challenges, visible and amenable to analysis.

While this approach can help with building better tools and methods for capturing design rationale, that is not our primary goal here. Rather, we aim to focus on the practice aspects of creating complex design rationale (DR) representations in groups. Our principal subject is not the participants in a collaborative DR session, although they are just as interesting in their own right. Rather, we are looking at the experience of people in the role of caretakers or facilitators of such events – those who have some responsibility for the functioning of the group and session as a whole. Collaborative DR practitioners craft expressive representations on the fly with groups of people. They invite participant engagement, employing techniques like analysis, modeling, dialogue mapping, creative exploration, and rationale capture as appropriate. Practitioners inhabit this role and respond to discontinuities with a wide variety of styles and modes of action. Surfacing and describing this variety are our interests here.

Good representations of design rationale do not come for free, and they often do not come easily. Proponents of DR tools and methods have long faced low adoption and even resistance to their approaches from many of their intended audiences (Buckingham Shum, 1996). Many researchers have explored this phenomenon, attributing it to factors such as the high cognitive overhead that the approaches seem to instill. For many would-be DR users, it requires considerable effort to move from customary forms of verbal and written argumentation, which seem to pour forth seamlessly, to the ostensibly more abstract forms of DR modeling, such as Rittel's Issue-Based Information System (IBIS; Kunz & Rittel, 1970; see Figure 1). Even initially enthusiastic prospective adoptees often run into a variety of difficulties as they try to build their first DR representations, finding the rhetorical moves unwieldy or struggling with the software tools to express and manage things as they would like.

Compounding these challenges by attempting to construct such representations in groups–with the additional interpersonal issues, group dynamics, and usual issues of trying to get things done in meetings—would seem to be a recipe for failure. And yet, successful practitioners of collaborative and participatory DR, issue-based exploration, and argument or dialog mapping do exist. A small but growing community of such practitioners has moved well past the "Can it be done?" phase, and these practitioners have successfully applied their approaches in a wide variety of professional, organizational, and research settings. For such practitioners, further improving their practice involves understanding and deepening the skills required. However, little in the research literature addresses such skills and practices directly, let alone research advanced enough to use them as the basis for developing a body of principles and guidelines, as other professional practices rely on. This paper aims at supplying some foundational considerations for helping foster increased attention to, and development of, such practices.

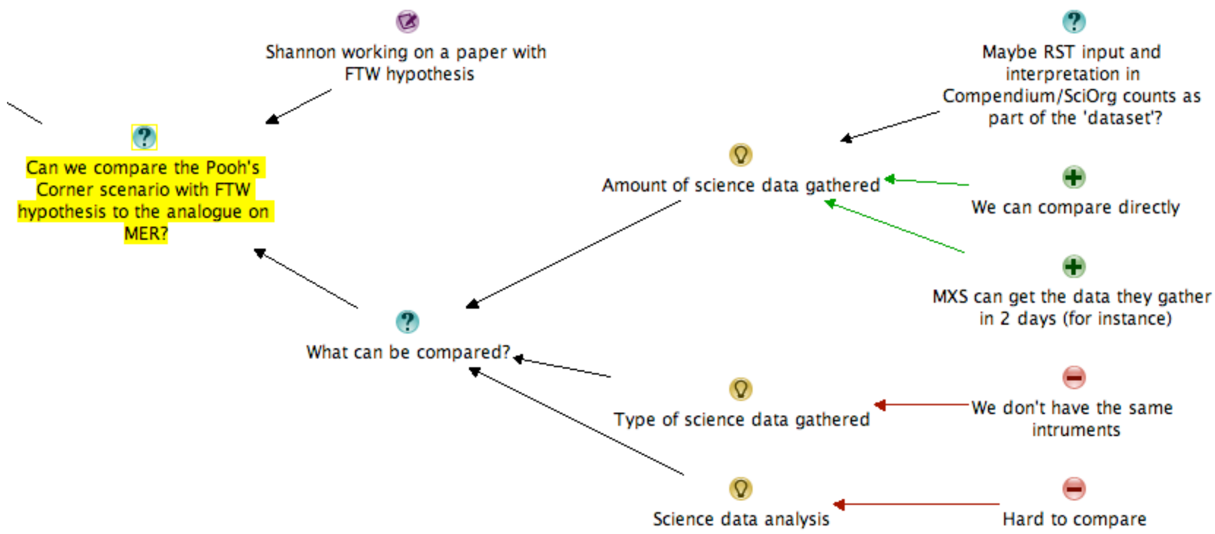

Figure 1. An example of an IBIS summary of a conversation, mapped during eScience field trials described in one of the case studies presented below. This can be contrasted with Figure 4, a much more constrained map largely generated by a software agent.

A note on terminology: There are many ways to refer to the practices we discuss in this paper, and the considerations described apply to other sorts of collaborative representations of knowledge besides design rationale. However for our purposes here we will use the abbreviation *PDR* in the rest of the paper to stand for participatory design rationale.

We authors have spent more than 40 collective years studying, developing, and working with PDR and argumentation approaches, both in individual and group settings. We have trained others to work with such methods, including classes specifically for practitioners intending to facilitate collaborative and PDR modeling sessions. As members of international communities of similar researchers, practitioners, and tool developers, we share an interest (in varying ways) in the practice dimensions of the approaches. Through these experiences, we have identified a number of considerations that appear to characterize the practice level of PDR.

In the balance of this paper, we describe these practice studies, explain our methodology, and provide illustrative examples. We also describe the theoretical framework that is taking shape against the background of repeated rounds of investigation and reflection. Key research questions include:

- (RQ1) What is the nature of the skills required to construct graphical knowledge representations in real-time, participatory settings?
- (RQ2) What are the kinds of choices practitioners face, especially at sensemaking moments within the course of conducting sessions?
- (RQ3) How does the context of the service being provided affect the choices a practitioner makes?

AN EXAMPLE OF PDR PRACTICE

What do we mean by the practice aspects of facilitating participatory design rationale? In this section we provide an illustrative example.[1]

A committee in a medium-sized public school district (approximately 20,000 students) in the Hudson River Valley region of New York State was tasked with analyzing the alternatives for school building capacity in the district, which has experienced declining enrollment. This highly contentious issue had come up many times before. The district's superintendent of schools was concerned that the discussion would be unproductive, due to tensions and unsurfaced assumptions between the various interest groups (school administrators, teachers, parents, taxpayers, etc.). Every school building has an active, vocal contingent of parents and teachers who have strong interests in keeping their own local school open. Equally strong and vocal are the many local taxpayers who feel that school taxes are already too high. To address this, the superintendent asked two outside practitioners with expertise in conducting PDR sessions to help run the meetings.

The practitioners convened a series of meetings in a library of one of the schools. A committee of twenty parents, teachers, community activists, and administrators met once a week to work through the alternatives. For each meeting, the practitioners prepared an agenda with a hypermedia issue mapping tool.[2] The agenda focused on various alternatives, policy matters, process considerations, and other issues.

The practitioners employed a variety of approaches. First, they facilitated a general discussion of the issues involved, using a conventional IBIS approach (representing discourse as issues, positions, pros, and cons) to capture and display the discussion as it proceeded. This involved rapid synthesis of what the meeting attendees were saying, thus creating nodes and links in the hypermedia tool that showed the relationship of statements to each other. They also validated the way they captured the statements by frequently asking the participants to look at the maps, asking "Does this capture what you said accurately?" Sometimes participants looked closely and provided detailed feedback (e.g, "Well, not really. What I was really trying to say was this…"). At other times, the heat of the discussion was such that it was difficult for the practitioners to intervene without running the risk of derailing the meeting's momentum. The practitioners had to make moment-to-moment decisions on how much to intervene, and in what ways.

Between meetings, the practitioners analyzed the maps from the general discussion. They looked for recurring themes and questions and, from these, created a template covering the major considerations that would guide choices between the alternatives (see Figure 2). They then facilitated several sessions using the template to structure conversation about each of the alternatives in turn. By the fourth session, the facilitators were able to induce the participants to conduct an analysis according to the template, while still capturing as much of the side discussion and issues as possible. Also between sessions, the district office distributed via mail all of the map output in text form to all the participants.

At the end of the process, the practitioners held a plenary session for the broader community to understand the final decision. The maps of rationale and templated analysis made the pros and cons for each alternative, as well as many of the comments and points of view, clear and explicit. Even though there was little consensus that the chosen alternative was the best one, the community members completing a postpresentation questionnaire agreed that the

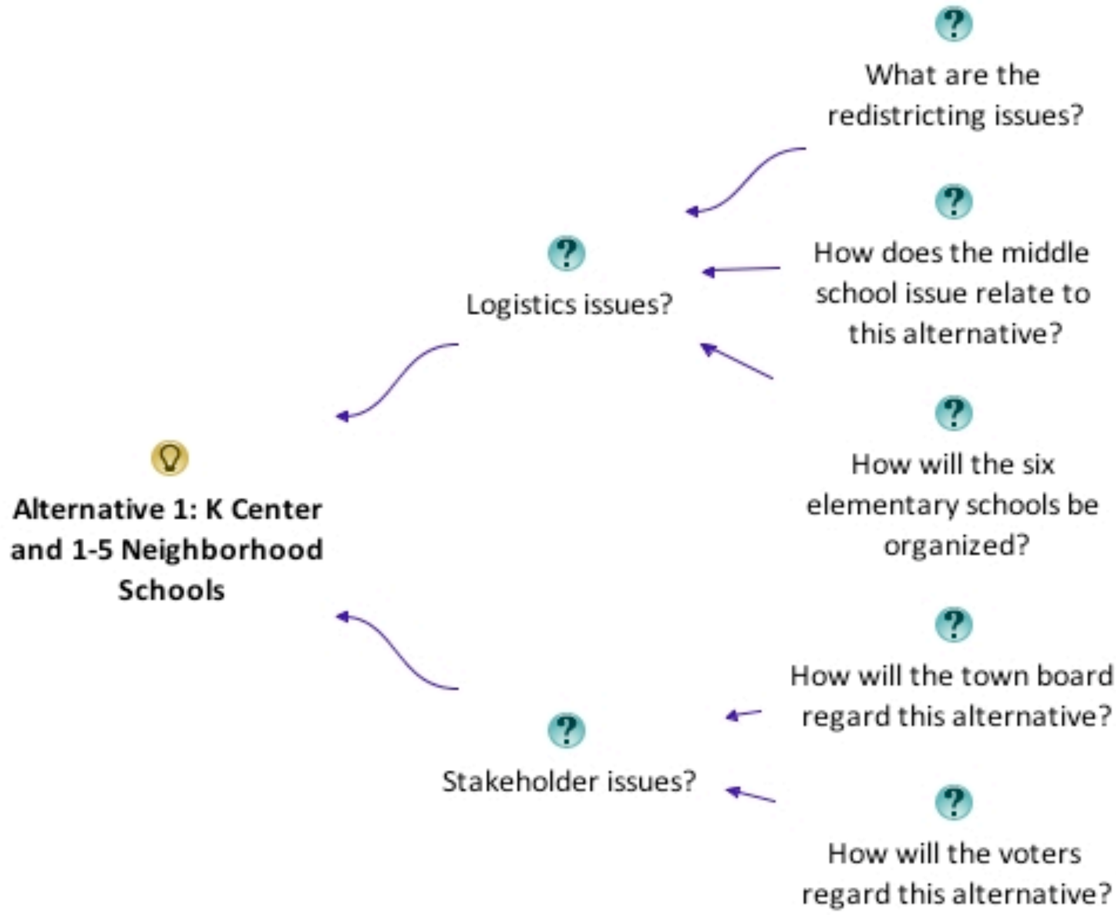

Figure 2. A portion of a meeting agenda, using a template to analyze alternatives.

process had been conducted in a fair manner, and that the discourse and competing points of view had been made more explicit and comprehensible than in previous years.

A FRAMEWORK FOR UNDERSTANDING PDR PRACTICE

Understanding practice like that described in the previous section requires taking into account a complex constellation of people, tools, representations, surroundings, and other factors. These have been summarized in the model shown in Figure 3.

The primary elements of the model are the people involved in creating the DR representation, and the representational artifact itself, as seen at the center of the diagram. The practitioner [a], which can be more than one person, orchestrates the participatory event and holds himself or herself responsible for its success. He/she is concerned with the quality and clarity of the representation and the participants' relationship to it. The practitioner takes primary responsibility for the form and content of the representation and the success of the session within its context [i]. As we saw in the example in the previous section, there can be

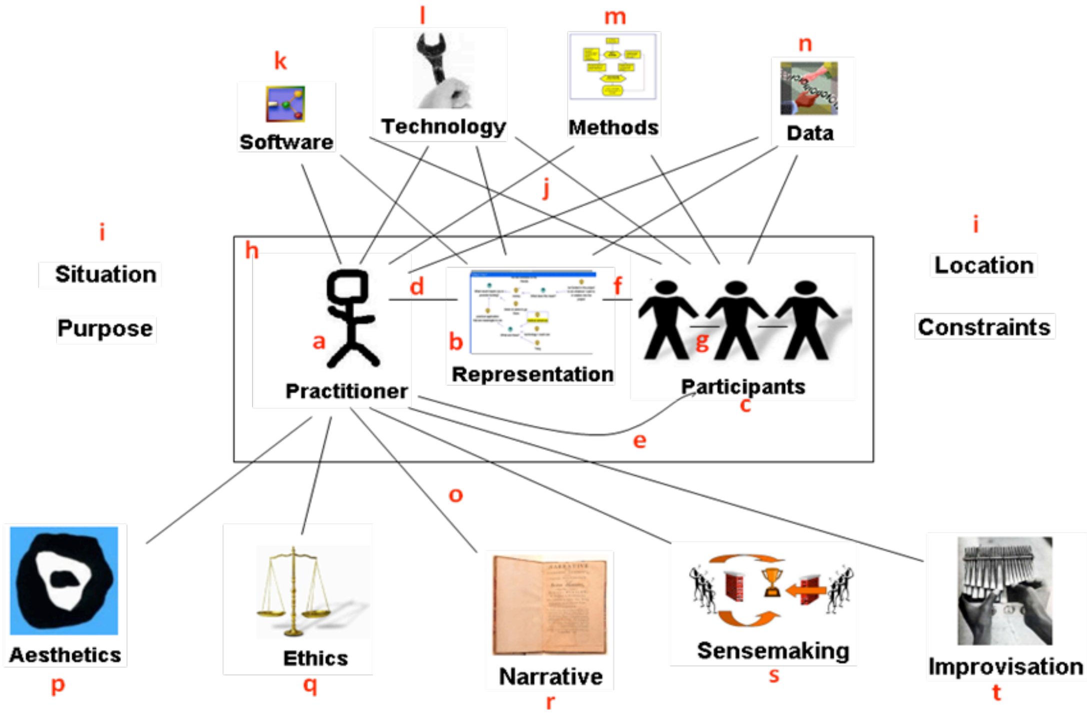

Figure 3. A framework for understanding participatory design rationale practice.

varying levels of intervention. The practitioners are not necessarily the ones with their hands on the equipment: Approaches where the participants themselves do the issue mapping directly are also possible (though often more difficult to carry out successfully). The practitioner interacts with the representation [b] as well as with the participants [c]. The nature of this interaction varies with the context and the specific role(s) that practitioners play in the activity system of the session. We follow McCarthy & Wright (2004) in emphasizing the particularity and situatedness of individual, as distinct from collective experiences of and responses to the tools and methods used in PDR sessions. As such, we look for the unique and creative appropriations practitioners can make, especially in uncertain situations. These often shift in the course of a project, such as the varying ways that the practitioners in the previous section engaged the participants with the representation. Over time they grew to understand both the needs of the different parties involved (parents, teachers, administrators, community members) and the kinds of attention each group was likely to pay to them as outside consultants with a limited franchise to change the accustomed (and contentious) group process. Through extensive "backstage" discussion and work with the materials, they evolved the PDR templates and engaged participants in the representation.

The representation [b] can be any sort of DR or other type of representation, ranging from paper-based argument diagrams drawn on an easel sheet to software-based discourse models, such as the hypermedia representations used in the example above and that we will discuss in our case studies below. There can be multiple types of representation used in a session, including notes and action items. The participants [c] are the people in the room (whether a real or virtual space) taking part in the session. Although the diagram depicts the participants as identical figures, in fact they are quite unique. Being aware of and appropriately dealing with the diversity of participant personalities, relationships, and interests is a key practitioner skill, as well as an ethical imperative. Line [d] symbolizes the interaction of practitioners with the representation, which consists of actions on it (such as creating or modifying it), considering it, planning what to do with it, or even ignoring it. As with that of practitioners, participant interaction with the representation is best understood in a situated manner. Each party in the school capacity example had a unique perspective on the proceedings, the representation, and the other participants. For example, community members whose main purpose for attending was to speak out for lower taxes had to be convinced that a facilitated process would serve their needs, while the administrators who had arranged for the consultants to take part had to balance their anxiousness about both the outcome and the credibility of the process as it played out.

Line [e] shows the interaction of practitioner with participants. This can take many forms, even in a single session, such as facilitative interventions (keeping matters on track, making sure everyone is heard), questions and discussion, and process checks. It is a two-way stream, as participants also interact with the practitioners in various ways. Line [f] is the interaction of the participants with the representation, which ranges from passive to active, from directly engaged with considering it and making changes to it to ignoring it or giving it occasional once-overs. Line [g] shows the interactions of participants with each other, from collegial to disputatious to side conversations.

The three primary elements (practitioner, representation, participants) are contained within box [h], symbolizing the boundaries of the session itself, such as a specific meeting. Some efforts may consist of a single session, where some comprise many sessions (which may include individual DR mapping sessions as well as collaborative ones). The session is in turn located within its surrounding context [i]. The context includes the overall project in which the DR activity is taking place, the specific locations where sessions are held (including whether they are face-to-face, virtual, or a combination); the situation that contains the session, such as the project of which the session is a part, the organizations involved, and the problem domain; the purpose of the session, and the constraints operating in the situation, such as time, budget, attention, or other resource limitations. In the school capacity example, the sessions were the individual meetings held in the library, while the context included the immediate school capacity project, but also elements such as the history of previous attempts to resolve the issue, the relationships of the various participants, and the constraints of producing a report within a limited timeframe.

The lines [j] show the relations between the primary elements and what can be called the set of enablers: software [k], technology [l], methods [m], and data [n]. Each enabler is connected to each primary element, because all interact with each. (Note that methods are not connected directly with the representation; methods are always filtered through a person's actions.) Practitioners use the chosen software [k] to operate on the representation; there can be multiple software packages in use (or none). Participants may also use the software. The software in turn runs on whatever technology platform [l] is in use, such as laptop computers. Technology also includes whatever display tools are being used, such as LCD projectors, virtual meeting or telepresence rooms, and voting keypads (non-computer technology such as flip charts, markers, and whiteboards also count). During sessions, specific methods [m] will be employed, whether formal methods such as IBIS argument mapping or data flow diagrams, or informal methods like brainstorming or round-robin discussion. All of these operate on and draw from the data [n], which is the subject matter for the session, the conversations and ideas put forth and captured during the session, and any supporting material, such as reference information.

Finally, we turn to the dimensions that inform an understanding of practice and the practitioner experience itself. Lines [o] show the aesthetics [p], ethics [q], narrative [r], sensemaking [s], and improvisation [t] associated with the work of the practitioner. These dimensions by themselves do not constitute creativity in the aesthetic and/or improvisational sense, but they help us see how creativity emerges when practitioners respond to breakdowns and anomalies in the course of PDR sessions. In the need to intervene in a session to restore its forward movement, practitioner creativity can result in choice, action, and materials seamlessly coming together to resolve the breakdown. Practitioner creativity can be seen in the ways they draw from these dimensions in the moment of action. Since these dimensions form the basis for the analyses described later in this paper, we expand on them in the following section.

DIMENSIONS OF PDR PRACTICE

It can be tempting to treat the work of a PDR practitioner as simply one of following established protocols, or unnecessary where it is assumed that meetings and participants can take care of themselves. Yet even when there are no so-called facilitators in a meeting, usually someone, however informally, takes on aspects of the role of ensuring that the meeting reaches its goals. If a knowledge construction task is to be done (as opposed to simply listening to someone else give a presentation), someone will often jump up and take notes on a flipchart or draw a diagram on a whiteboard. This is just as much what we mean by *practitioner* as a paid professional who comes in to run the process and generate the products of a meeting.

In either case, when people act as PDR practitioners in our sense of the word, they inherently make choices about how to proceed [q], give form to the visual and other representational products [p], help establish meanings, motives, and causality [r] and respond when something breaks the expected flow of events [s], often having to invent fresh and creative responses on the spot [t]. These aspects of PDR practice are summarized in Table 1 and described in the following subsections. Although we present them as separate entities here for the purpose of description and analysis, in fact in they commingle in the experience of practice, as will be seen in the illustrative example at the conclusion of this section.

Table 1. Dimensions of Participatory Design Rationale Practice.

Practice Dimension	Definition
Aesthetics [p]	How practitioners shape and craft the representation
Ethics [q]	How a practitioners' actions affect other people
Narrative [r]	Meaning and causality applied to the flow of events
Sensemaking [s]	The ways in which practitioners deal with situations of doubt or instability
Improvisation [t]	The spontaneous, creative moves that practitioners can make

Aesthetics [p]

All diagrammatic DR approaches have explicit and implicit rules about what constitutes a clear and expressive representation. People conversant with the approaches can quickly tell whether a particular artifact is a “good” example. This is the province of aesthetics.

Aesthetics has to do with what human beings, in the moments when they are imparting expressive form via some medium (Arnheim, 1967), are actually doing: pulling together aspects of experience into a new whole that itself provides a shaped experience (Dewey, 1934/2005). The aesthetic dimension of PDR practice is concerned with the shaping and crafting of DR representations in response to both immediate and context-specific imperatives (things that must be done to help achieve participant and project goals), as well as to implicit and explicit concepts of right form. Using the lens of aesthetics offers a unique perspective on the relationship of a practitioner to the participants, emphasizing process, collective and participatory expressive forms, even ethical and political concerns (Cohen, 1997). Understanding the aesthetic dimension of a collaborative practitioner’s work emphasizes how the encounter between participants, representations, and practitioner unfolds, the extent to which representation-building engages participants, and the ways in which participants are affected by the proceedings.

In explicitly incorporating the idea of aesthetics, we follow Dewey’s (1934/2005) argument that aesthetics is not an elite, esoteric, or rarefied concept, even though it is treated that way in common usage. Rather, it is to be understood as the high end of a continuum from prosaic experience; it is a paradigm for “true,” unalloyed experience. Aesthetics govern how we would experience any situation if the diluting, dulling, oppressive, or conflictual aspects were stripped away.

Our research investigates what distinguishes form-giving actions from other sorts, looking at the uniquely aesthetic characteristics of such actions in the work of a PDR practitioner. When working with groups, the boundaries of the world of experience are closely aligned with the situation in which they are operating – the people, goals, interests, and constraints of the project or team they are working with. Even within this bounded world, the dimensions and particulars of experience can be vast and diverse, so the problem – and hence the artfulness – of pulling them together into an “integrated structure of the whole” (Arnheim, 1967, p. 5).

For example, we look closely at how specific choices regarding form respond to the situation and express something of uniqueness (or fail to). Skilled practitioners can make choices in their actions on the representation that impart a complex of meanings and nuances.

Ethics [q]

The ethical dimension is concerned with the responsibilities of the practitioner to the other people involved, and to their various individual and collective needs, interests, goals, and sensibilities. In some situations, these responsibilities can be weighty in nature—for example, in situations of conflict or dispute, where every action and statement on the part of participants or practitioner holds the possibility of worsening the situation. In less fraught settings, consequences of action or inaction may be less severe, but can still have effects on the concerns of the participants or other stakeholders. Of particular concern are practitioner actions that affect the engagement of participants with each other, with the subject matter of their work, and with the nature and shaping of the representations. These often can take the form of questions: Should I do action *x* or action *y*? What effect will it have on these participants if I do *x*? Should I intervene in their conversational flow? or Should I expend the effort to capture everything that person A is saying at this moment, or is the time better spent in cleaning up the map or preparing for the next activity?

Aakhus (2001, p. 362) advocates research into the communicative actions of facilitators, so as to "advance the normative level of communication practice." He stresses that facilitators' work is not just a neutral enabler of participants' decision-making, or a simple unfolding of a priori processes, but rather contains many instrumental aspects in which practitioner choices directly affect participants and the course of events during sessions of their work. He also examines the "transparency work" performed by communication practitioners in an ethical light (2002). This work, the result of active crafting on the part of the facilitator, is often invisible in accounts of practice. Aakhus (2003) further critiques frameworks that deemphasize the ethical responsibilities of particular mediation and group facilitation practices, arguing that "objectivity" is an inaccurate way to frame practitioner actions. Other researchers also examine choices and dilemmas faced by group support systems (GSS) facilitators (e.g., Yoong & Gallupe, 2002). Facilitators do in fact intervene in their clients' situations. Schön (1983) argues for practitioners to take active and conscious ethical stances, recommending reflection-in-action as the means to achieve this. Our research identifies moments when practitioners make choices with such ethical implications. These often arise and pass quickly, such as the momentary shift in attention away from the participants that we see in the example presented below.

Narrative [r]

The narrative dimension concerns the connecting of diverse moments and statements over time, as well as the human experience of causality and consequences. Practitioner actions that have a narrative dimension – that serve to connect elements of the story being built in the DR representations for later telling and reading by others – contribute to the narrative shaping of both the effort itself and the representations that are the primary focus of their actions. Narrative is both a basic human developmental mechanism independent of any particular embodiment (Murray, n.d.) and an aesthetic form that can be represented in verbal, written, performed, or other forms. Narrative functions as a key human strategy for exploring and overcoming unexpected turns of events. Stories and story-making form a key psychological strategy for connecting disparate events. This is particularly so when there is a break or disruption from an expected course of events. "The function of the story is to find an intentional state that mitigates or at least makes comprehensible a deviation from a canonical cultural pattern" (Bruner, 1990, p. 49).

The skill of the storyteller lies in the artfulness and effectiveness with which he/she can craft an artifact that makes sense of the "breaches in the ordinariness of life" (Bruner, 1990, p. 95). Narrative is a central means by which we are able to glue together bits of experience to construct a new understanding. It is also a key part of human development, a way that we learn to construct and communicate understanding of events and environments. Further, narrative is an intentional form – things that are created, with varying degrees of skill, to serve various purposes. Approaches like scenario-based design employ narratives to capture both concrete detail and the inherent ambiguities in design situations, as well as to create communicative artifacts that can help bridge disciplinary differences (Rosson & Carroll, 2009).

McCarthy and Wright (2004) point out that, as individuals, our interactions with technology can be understood through the prism of roles like author, character, protagonist, and coproducer. We are always actively engaging with technology as individuals with our own aims, history, emotions, and creativity, as much as we are also embedded in a sociohistorical context or attempting to perform some kind of task or composite activity.

In our approach, narrative analysis provides a frame for understanding practitioner efforts to maintain the coherence of representations even in the face of interruptions and potential derailments within sessions. Narrative provides a way to understand what coherence means in the context of a particular session (e.g., What is the intended arc of events? How is that arc meaningful to the participants? What roles do the various parties play and how are those important within the surrounding situation?). As well as looking at this encompassing framing of a session, we also look at the ways breaches of the expected occur, and how the practitioner as protagonist reacts to these. Finally, we look at the narrative aspects of the DR representation itself and how changes to the representation relate to the other narrative levels at play in and around a session.

For example, in one of our case studies that took place at a small workshop, the following narrative elements provided key context: There was a pre-existing set of conditions that framed the event, supplying expected causality, reasons for people to be at the event, expected roles, and assumed meanings. Some of the relevant narrative aspects included the ostensible purpose of the workshop, the personal reasons each participant had for attending (e.g., what they hoped to gain from it), the expected trajectory of the facilitated session itself, and the practitioner's own expectation that she would be able to capture and represent the discussion as it unfolded. When the session started to unravel due to a drift in focus on the part of participants (as well as the surfacing of some metadiscussion, like "Why are we talking about this?"), this constituted a breach for which the relatively novice practitioner had no ready-made, unproblematic response.

Sensemaking [s]

Creating DR representations is in itself often a way to help negotiate and construct a shared understanding (Weick & Meader, 1993) of a situation or project as a whole. Within this larger frame, the act of representation itself engenders both negotiation as well as confusion, when the tools and discourse lose, if even momentarily, a clear sense of fit. In many design sessions, there are moments where forward progress is blocked because of unforeseen, uncontrolled, or otherwise problematic obstacles. Our research focuses on the sensemaking dimensions of the actions, and their consequences, that take place at such moments. They call for creative and skilled responses from whoever is playing a facilitative/representational role, since programmed or prescribed responses and rote actions are rarely sufficient in such situations.

Dervin's (1983) model of sensemaking posits that a person is always attempting to reach a goal, or set of goals. Goals themselves shift in priority and nature, in time and place. Some are explicit where others are tacit. The person moves toward these goals until an obstacle (a gap in Dervin's terminology) stops them. The obstacle impedes the person's progress and stymies efforts to continue. In order to resume progress, the person needs to design a movement (a bridge) around, through, over, or away from the obstacle. This can be as simple as asking someone for directions or help, or a complicated set of actions that may have a trial-and-error character. “As an individual moves through an experience, each moment is potentially a sense-making moment. The essence of that sense-making moment is assumed to be addressed by focusing on how the actor defined and dealt with the situation, the gap, the bridge, and the continuation of the journey after crossing the bridge” (Dervin, 1992, p. 69-70). These sensemaking actions can be understood as attempting to answer a set of tacit questions: What is stopping me? What can I do about it? Where can I look for assistance in choosing and taking an action? Weick and Meader (1993, p. 232) define sensemaking as the process of constructing “moderately consensual definitions that cohere long enough for people to be able to infer some idea of what they have, what they want, why they can’t get it, and why it may not be worth getting in the first place.”

Although in some ways sensemaking can be thought of as a perpetual, ongoing process (Weick, 1995), it is also something placed in sharp relief by encountering surprise, interruption, or “whenever an expectation is disconfirmed” (Weick, 1995, p. 14). Schön (1987, p. 19) characterizes such moments in professional practice as situations of “complexity, instability, and uncertainty,” laden with “indeterminacies and value conflicts.” Such moments are further defined by a “density of decision points” (Sawyer, 2003, p. 145). In professional practice, the moments where sensemaking comes to the fore can have the character of impasses (Aakhus, 2003) or dilemmatic situations (Tracy, 1989; see also Aakhus, 2001).

PDR practice can include many such moments. Our research looks at the particular character of practitioner sensemaking at those moments, especially as it is expressed through moves on the representations, explorations of and changes to them, and interactions with participants about them (Selvin & Buckingham Shum, 2008, 2009). We consider in what ways DR representations, and the practitioners’ interactions with them, contain both a source of obstacles and impasses, and a means of resolving or addressing them. In part, we focus on such moments because it is often where practitioner skill and creativity are most clearly manifested. In the example at the end of this section, we see a sensemaking trigger occur when participants discover that the geospatial data they had expected to see was missing from the artifact they were examining. We will see a further example described later in the paper.

Improvisation [t]

As discussed in the previous subsection, practitioners often encounter moments where they must deal with the unexpected events in the course of a PDR session. While some aspects of participatory DR practice follow predetermined patterns and draw on techniques and methods planned in advance, skilled practitioners often find themselves switching to alternative sensemaking strategies, or even improvising. It is the degree of creativity employed at this point that distinguishes the *improvisational* dimension of action from other sorts of sensemaking activities. Improvisation can be discerned in the freshness and innovativeness of the response to an event that triggers sensemaking.

Improvisation is difficult to control for, or measure in, laboratory or outcome-based studies of software tool use. Some research into meeting behavior, such as the use of GSS technologies, tends to regularize the practices surrounding the technology, analogous to similar moves to "script" teacher-student interactions (Sawyer, 2004) and otherwise de-skill or de-emphasize the creative aspects of many sorts of professional practices (Schön, 1983). Yet improvisation is central to understanding what truly occurs in real-world software use situations, especially where there are creative, unpredictable elements at play, such as constructing a representation of design rationale with a group of people in live conversation.

Sawyer (1999) discerns three levels at which to understand improvisation: individual (improvisation on the part of particular actors), group (improvised interactions within a bounded, particular situation), and cultural ("the pre-existing structures available to performers — these often emerge over historical time, from broader cultural processes"; p. 202). The cultural level supplies the elements of a practitioner's repertoire (Schön, 1983), the collection of pre-existing techniques and concepts (whether learned in school or from work or other experiences) that contain what the practitioner draws from, combines, and invokes in the heat of an encounter. Practitioners of exceptional skill often possess repertoires of great range and variety (Schön, 1983) that they are capable of combining in innovative, expressive, and subtle ways. This kind of characterization is particularly apt when a practitioner is confronted with a situation of confusion or uncertainty, where he/she can no longer continue on with a single pre-existing method or technique (though a return to it later is possible) and must make rapid decisions about what actions to take and ways to inflect those actions, or risk losing the coherence of the session, thus jeopardizing its goals.

Maintaining an awareness of the emergent aspects of a situation, however, does not mean that all is left to chance. Sawyer (2004, p. 12) emphasizes the concept of "disciplined improvisation," which juxtaposes improvisational aspects of practice (dialogue, sensemaking responses, spontaneous and creative acts) with "overall task and participation structures," such as "scripts, scaffolds, and activity formats." Skilled practitioners are able to navigate judiciously between moments when they can rely on pre-existing structure and scripted actions, and moments calling for fresh responses and combinations. In a PDR session, improvisation can take many forms, such as sudden shifts in stance or tool strategy. Often these are mini-improvisations that occur and conclude rapidly, unplanned and not referred to verbally in the course of other sorts of actions. This is seen in the example below, which discusses a sensemaking trigger, an improvised response, and the aesthetic dimensions of the response.

An Example

By way of illustrating some of the phenomena discussed above, here we present a highly abbreviated portion of an analysis of one of the episodes from a case study.

Figure 4 shows the result of an episode of improvisation on the part of an expert practitioner that took place between 61m27s and 63m12s of a 2h15m session. In the context of NASA field trials (Clancey et al., 2005; Sierhuis & Buckingham Shum, 2008), a distributed team was working through auto-generated maps of science data associated with a robotic rover trial. The team suddenly realized that some of the expected data (geographic waypoints) were missing from the map.

As soon as he heard the participants commenting in surprise about this ("What waypoint is this?"), which constituted the sensemaking trigger, the practitioner spontaneously launched a search for potential sources of the missing information, opening and inspecting the contents of several other maps.

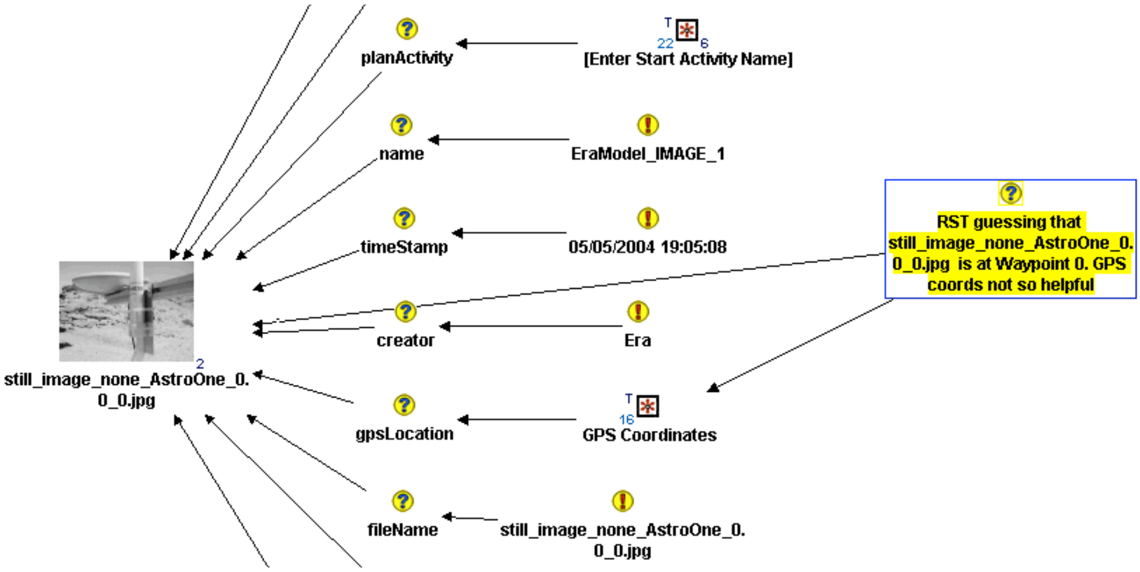

Figure 4. Portion of a screenshot from Mobile Agents project, showing an improvised response to a sensemaking trigger.

After determining that he could not find the waypoint information either, and while listening to the participants discuss their own attempts to locate the data in other records, the practitioner returned to the initial map and created a question node (highlighted in yellow in Figure 4) capturing the preceding deliberation from the participants. This was impromptu, not directed by the participants and not in response to any particular coda in the conversation. The practitioner determined that the group would not be able to get any more information to inform the waypoint determination than what they had just said.

The practitioner made several aesthetic choices during this event. He chose an area in the white space to the right of the imported science data nodes, implying or emphasizing by this choice that the new node is a comment on the science data rather than an addition to it: It is outside of the pre-existing imported science data. He also chose to link the node to the main image node, drawing the link across all the other nodes in the view, which serves to make it more dramatic, and possibly effective, emphasizing the disruptive quality of the missing information and the effect it had on the session. He makes a textual aesthetic choice in his use of the gerund *guessing* to imply the unfolding, transitive nature of the comment in the node. If he had used the past tense (“RST guessed”), it would not have conveyed the same process sense of the moment. He also chose to link it to the GPS Coordinates map node, indicating that the node is also commenting on the data contained in that map.

Summary

In our approach, we see the concepts of narrative, sensemaking, and improvisation as providing complementary frames for interpreting instances of practice. Narrative theory provides ways of looking at the container, purpose, intention, and gives the context for the breaches that occur. Sensemaking looks particularly at the breaches and the ways in which actions and representations respond. Improvisation within the context of sensemaking is where we can often most clearly see practitioners demonstrate relative levels of skill and artistry. All of these have both aesthetic and ethical dimensions.

These dimensions are not usually explicit in our source data. Caught up in the proceedings as they are, it is not often that a practitioner or participant in a PDR sessions will directly comment on the narrative framing or aesthetic shaping at work. We have had to develop a number of tools to help us discern and analyze how the dimensions of our framework are manifested in instance of actual practice. These are described in the following section.

STUDYING THE PRACTICE LEVEL

In this section we describe how we analyzed the ways in which the above dimensions play out in situations of actual PDR practice. As befitting exploratory work in an underresearched domain, we have employed qualitative research techniques to identify themes and hypotheses through close analysis of video and screen recordings of PDR sessions. Qualitative approaches, such as grounded theory (Strauss & Corbin, 1990), are generally regarded as appropriate when a field or phenomenon is in its early stages, and when research problems and theoretical issues are not yet well defined. In addition, many of the considerations that the practitioners we are studying must deal with are emergent in character, responding to the unexpected events and anomalies that intrude on even the most carefully planned sessions. Indeed, sensemaking considerations form the core of our analysis here, since being able to resolve the anomalies they encounter is a key success factor for the practitioners we study. The ability to diagnose and repair breakdowns by drawing on a pre-existing “repertoire of expectations, images, and techniques” (Schön, 1983, p. 60), as well as fresh creative responses in a near instantaneous fashion, is the hallmark of successful professionals of many kinds, and is no less the case for the practitioners we study.

Source Data and General Approach

We have studied both experienced and relatively inexperienced practitioners. These include several in-depth microanalyses of long PDR sessions, looking at how highly skilled practitioners encounter and solve sensemaking challenges in the course of working with their participants (Selvin, 2008). The settings were in situ sessions, often several hours long, held as part of larger projects, where the tasks carried out emerged from the highly contextual needs of those projects (such as a NASA remote science team looking at geological data during virtual meetings over a week-long field trial). We also conducted experiments where teams of mostly novice practitioners planned and carried out a facilitated session for their peers on the theme of space travel. In both types of studies, our primary data are video and screen recordings of the sessions. We analyzed these recordings using a grounded theory approach (Strauss & Corbin, 1990), paying special attention to participant and practitioner verbal statements, practitioner actions, and “moves” on the DR representation itself (changes done to the representation, such as adding a node or editing label text).

The analysis focused on characterizing the choices made by the informants in their preparation period (what they were trying to achieve, how they organized the base materials using the software, their intended flow of events, the roles they assigned, the software aspects they intended to leverage) and in their enacting these during group sessions. Using critical incident analysis (Tripp, 1993), we then selected moments where practitioners were faced with some kind of anomaly in the course of a session. We looked at the specific practitioner moves and choices that determined the outcome of the sensemaking moment, focusing on the aesthetic, ethical, improvisational, and narrative aspects of those moves and how these contributed to the ways in which participants engaged with the representation, with special emphasis on the character of the real-time shaping of the representation. Through repeated viewings and application of a number of analytical instruments (described below), we built up explanatory concepts, categories, and properties, focusing on the engagement of both practitioner and participants with the hypermedia representation.

Iterative Development of Analytical Tools

The five tools described below emerged from repeated rounds of analysis and reflection. In each, we started from the data (the recordings of PDR practice) and identified patterns and concepts that appeared to recur in the moves and statements contained in the video recordings. Early on, we concentrated on the move-by-move level and developed a fine-grained instrument with a number of categories derived from open and axial coding on the contextual meaning of each move and statement in a session. We identified sensemaking triggers in those sessions, moments where something disturbed the expected flow of events and forced the practitioner to do something different, often requiring creative improvisation to resolve the episode and return the session to its intended track. We then wrote narrative descriptions of these episodes, starting with the sensemaking trigger, describing the actions within the episode, and explaining how the episode was brought to closure.

While this approach produced a large amount of richly described data, several limitations became apparent. First, it was extremely time-consuming to apply the 18 analytical categories to each move and statement of a 2-hour session, which might contain over 1,300 moves and statements. A grid analysis of a single 2h15m session required almost 24,000 cells in a spreadsheet. Second, important aspects of the context itself seemed to recede as we concentrated on the individual moves. Without losing our focus on the meaning of individual moves, we needed a way to frame those moves that could more clearly connect them to their context, especially in ways that allowed us to identify the aesthetic and ethical dimensions informing the moves. This led us to develop two further instruments. The first provided a way to characterize the aesthetic "shaping" that both was intended (planned) and actually occurred during a session. The second was a distillation of the more finely-grained concepts and categories from the grid analysis that allowed us to characterize broader timeslots in a session with a more manageable set of three criteria derived from our open and axial coding. Both of these processes gave us the means to frame the episodes covered in the other analyses in the context of the session as a whole, in such a way as to highlight our dimensions of interest at all three levels of granularity (session, timeslot, and move).

Finally, we wanted a way to connect the results of these analyses more explicitly to the dimensions of our theoretical framework. This led us to create the "framing" tool. Its categories and questions are derived from the framework, conceived as an ideal, normative model for how a practitioner should act in a PDR situation. This allowed us to compare what actually happened in a session to an ideal model, so as to highlight how practitioner choices moved either

Table 2. Summary of Tools for Analyzing Participatory Design Rationale Practice.

Analytical Tool	Description
Shaping form	Characterizing the representational character of the whole session to delineate the intended and actual shaping that took place
CEU analysis	Mapping the coherence, engagement, and usefulness (CEU) dimensions of timeslots within the session. Aids in identifying sensemaking episodes
Narrative description	Rich description of a sensemaking episode, including dialogue and descriptions of events
Grid analysis	Micro-moment moves and choices during the episode
Framing analysis	Characterizing the practitioner actions during the episode in aesthetic, ethical, and experiential terms

closer or farther away from ideal behavior. The full set of analysis tools is summarized in Table 2 and described further in the following sections.[3]

Analytical Process

For each of the PDR sessions we analyzed, we employed the analytical instruments described above in the sequence represented in Figure 5. By applying this set of tools, we aimed at achieving both qualitative triangulation (Fortner & Christians, 1981) and increasing theoretical sensitivity (Strauss & Corbin, 1990) by looking at the data through multiple lenses.

We started by viewing the video recordings several times and creating a transcript of the entire session. Then, for each of the tools, we viewed the recordings again with the specific lens provided by that tool, which are described below.

Shaping Form

The shaping form comprises a set of questions asked about the session as a whole. It aimed at characterizing the representational character of the session. We described what kinds of roles participants and practitioners played in the shaping of the representation, both as a result of planning and intention, and in response to whatever exigencies actually occurred during the session.

The questions included a characterization of the overall ecosystem of the session (the surrounding context, purpose of the session, types of participants), as well as a number of questions designed to put focus on the interaction of people with the representation. Table 3 relates the questions to the dimensions of the framework.

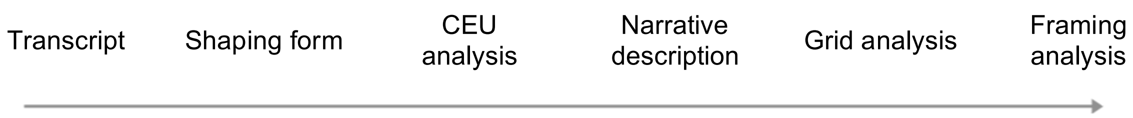

Figure 5. Analysis sequence.

Table 3. Relation of Shaping Form Questions to Dimensions of Participatory Design Rationale Practice.

Shaping Form Question	**Relation to Framework**
What shaping was intended (how the session was planned to work, what shaping the planners intended to occur, and how it would be accomplished)?	Shaping itself is largely the province of aesthetics [p], the construction of meaningful form. This question refers to the planned or intended sorts of shaping (which may or may not have occurred in the actual session).
What was the level and quality of participant and practitioner engagement (with maps, subject matter, process, environment)?	This question concerns the relationships of participants, practitioners, and representation to each other [framework elements d, e, f, g], as well as to the surrounding context and resources [i, j].
What types of shaping actually occurred during the session?	Means to report what sorts of aesthetic shaping [p] took place in the actual session.
If the intended shaping went awry, why did that occur? What blocks an intended shaping? How are the blocks resolved or avoided?	Identifies what sensemaking [s] triggers may have occurred, placing them in the context of the overall narrative trajectory of the session [r]. Explores the degree of improvisation [t] in resolving or avoiding obstacles to progress.
Who did the shaping, for what reasons? What contributions to the shaping occurred?	Maps the shaping actions [p] onto the way their performers related to the representation [d, f].
How were decisions about shaping made? What kinds of decisions were they? Who made them, on what basis? How were these decisions taken up into the representation itself (if they are)? Which are ignored or dropped? Why?	Looks at the choice making involved in both shaping actions and participant inclusion or exclusion in those actions. Often the clearest way to discern the situational ethics [q] of the practitioners.

The result takes the form of a narrative document, (e.g. Figure 6).

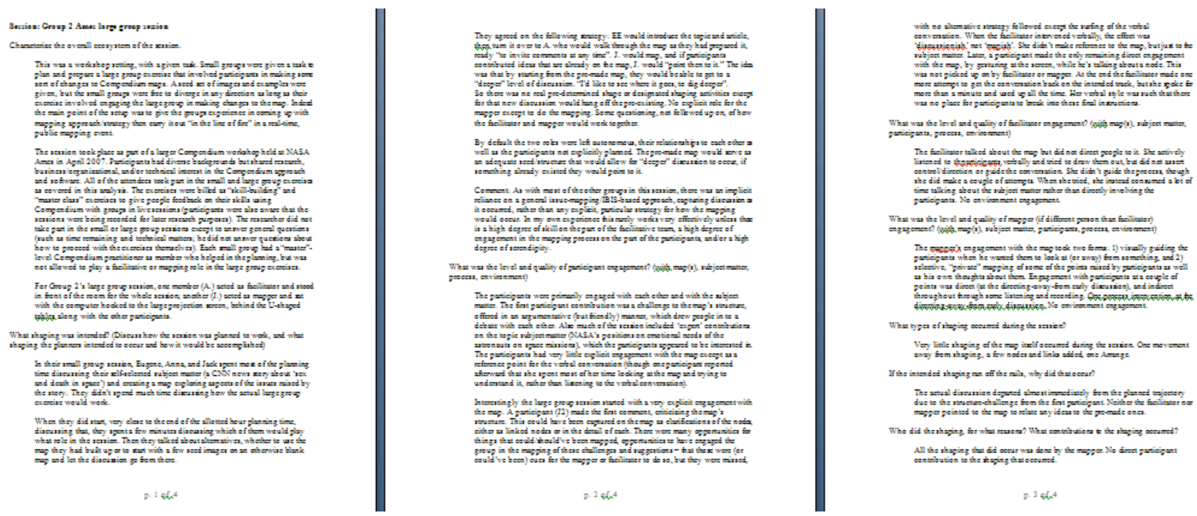

Figure 6. Example of a completed shaping form.

With the overall character of the representational role described, we now use the CEU tool to zoom into a lower level of detail to characterize the session as it unfolds over time.

Coherence, Engagement, and Usefulness (CEU) Analysis

In this analysis, we coded the CEU dimensions of each timeslot to build up a signature (in the sense of a distinctive pattern that indicates the character) for the session. When visualized as a grid, this provides a gestalt view, showing the extent to which the representational artifact being maintained by the practitioner was co-constructed by participants, in a way that seemed to add value.

Coherence involves keeping the information display, and the interaction of participants with it as well as with each other, understandable, clear, evocative, and organized. At any moment, the meaning and organization of the visual and textual elements of the display should be clear to participants (as well as practitioners). *Engagement* refers to the relationship of participants to artifacts in sessions involving any sort of representation, whether a whiteboard, easel sheet, or software projected in front of the real or virtual room. The value of the representation is directly related to the degree that the participants are engaged with it – whether they are looking at it, talking about it, referring to it, and involved in its construction or reshaping. *Usefulness* refers to the extent to which the representation appears to be adding value for the participants and helping to fulfill the predetermined or emergent goals of the session. It is the responsibility of the practitioners to make sure that the representation is a useful part of the proceedings.

We divided the video and screen recordings into 30-second timeslots. For each timeslot, we rated how the session had fared in that timeslot in terms of the CEU of the relationship of the participants to the hypermedia display. There are three ratings: High (three points), indicating a high or strong degree of engagement, coherence, and usefulness; Medium (two points), indicating a medium or average degree of the three criteria; and Low (one point), indicating that there was a low degree during that timeslot. Table 4 provides a set of examples illustrating how each rating is derived from the video data. The way we arrived at each rating was derived from the specifics of the session and timeslot itself, and thus vary in what we looked for and were able to discern in the data. Some ratings were assigned based on participant comments or observations of practitioner actions, while others by examining the representational artifact itself at that moment in time in the context of the current participant statements or actions.

For example, the DR representation in a specific timeslot might display a high degree of clarity and “readability”; all the content is legibly presented and laid out, and is faithful to the statements, tone, and purpose of the meeting (at least of its current activity). Thus we would rate both Coherence and Usefulness as High (3 points each). However, at that moment the participants are caught up in a side topic and are not paying attention to the representation, therefore we would rate Engagement as Low (1 point).

By assigning a color to each rating in the spreadsheet, we generate *heat maps* that provide a gestalt visualization of the whole session in terms of the three criteria. Figure 7 shows a comparison of CEU heat maps from six different sessions. Such heat maps make it possible to identify the overall tenor of the session, and to point out where sensemaking moments, or breakdowns, may have occurred—typically when the 3s (High ratings, green shading) drop to

Table 4. CEU Ratings and Exemplars.

Criteria	Low	Medium	High
Coherence	The representation is unclear or bears little fidelity to the current focus of interest; e.g., a participant remarks that "I don't see what we're talking about" on the map.	Moderate level of coherence, e.g., some confusion about the meaning of the way various nodes on the map are tagged, but generally the representation is clear enough to follow.	The representation is a clear reflection of the discussion or exercise, in form, content and organization. All participant contributions have clear places to be entered and linked on the map.
Engagement	The participants are paying little or no attention to the map; e.g., some participants are having a side conversation with no reference to the map.	An example is when participants start to make side conversation while practitioners are in the midst of making a complicated change to the map, rendering it temporarily less than clear.	Participants are looking at, talking about, and appearing to care about what is on a map; e.g., a participant validates that the way the practitioner has captured his/her input on the map is accurate.
Usefulness	The representation is not acting as a tool toward the realization of the session's purpose; e.g., the map is no longer keeping up with either the intended exercise or the emergent conversation.	This is evident when it is partially, but not completely, clear to the participants how the map will help them complete the exercise.	Indicates that the representation is integral to the achievement of the session's purpose; e.g., the structure put in place for the exercise is working efficiently; participants understand the sequence of events and actions.

2s (yellow) or 1s (red), indicating that the representational artifact seemed to add little or no value at that moment. When a session has High ratings throughout, it can indicate that the preparation and execution of the session (design and realization) were both well thought out in advance and handled in practice. In such sessions, possible breakdowns are avoided, often through the expertise of a practitioner.

Figure 7 also shows an overview of the sensemaking character of six of the sessions studied. This visualization shows that three of the Ames sessions contain a fair amount of red cells, indicating Low ratings for one or more of the CEU elements (possibly reflecting the relatively novice level of most of these sessions' practitioners). These are moments in the session when the session went somewhat awry in terms of the practitioners' intentions for having the group co-construct the representation. These would be prime locations to look for the sensemaking triggers (what set off the drop in the ratings), as well as what the practitioners and/or participants did to restore the session to better functioning. We can also see that the remaining Ames session as well as the two Rutgers sessions had few or no drops, indicating that the practitioners and participants experienced relatively unproblematic going.

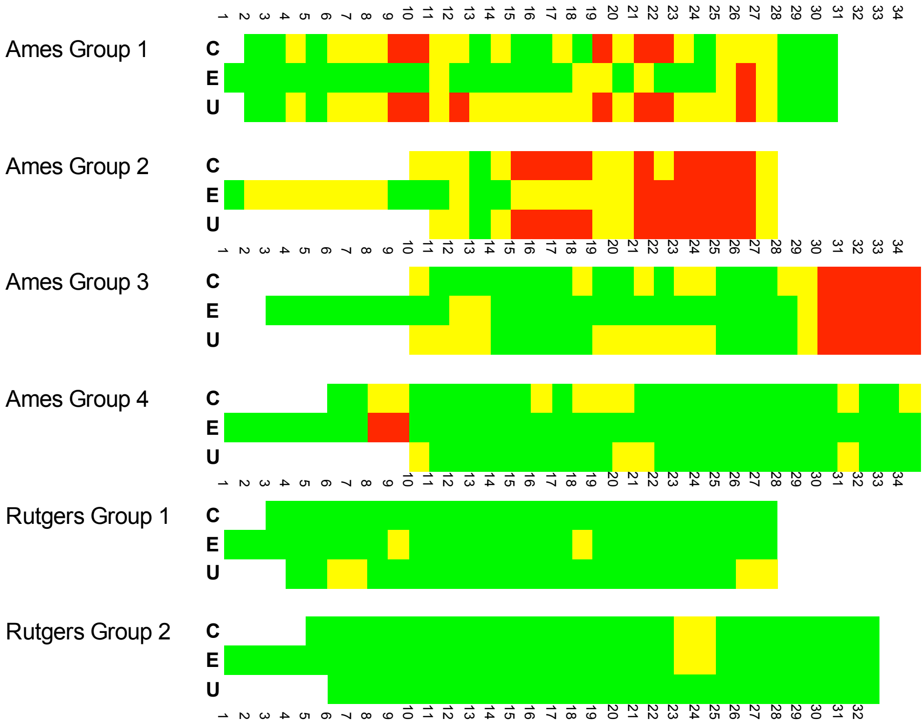

Figure 7. Heat maps from CEU analyses.

In fact those sessions proceeded very close to plan, whereas the Ames groups 1, 2, and 3 all experienced sensemaking challenges.

Other researchers (e.g., Yoong & Gallupe, 2002) apply to electronic meetings similar coherence and engagement constructs as the ones we invoke here. The main difference is one of granularity of analysis. Our primary interest is closer to the brushstroke level – understanding the meaning of the individual practitioner move, when set in context – than the whole-meeting level at which other researchers apply ideas of CEU. At this stage we are not attempting to find inherent relations or patterns among the three criteria, although that may be an outcome of future work.

After reviewing the shaping form and CEU analysis for a session, we selected a particular sensemaking episode for closer analysis. This new analysis started with a narrative description.

Narrative Description

The narrative description provides a rich delineation of a sensemaking episode within a session. For this, we identified a starting and ending point for the episode, from the point of the sensemaking trigger (an event or anomaly that initiates some sensemaking behavior) to its resolution or culmination. Sometimes there was no resolution per se, for example, when the practitioners were not able to bring a session back on track after a breakdown. For our purposes, this can happen when participants cease engaging with the representation and just talk to each other without any reference to the representation, as the excerpt below describes:

> In the second episode, the session does not recover from a resurgence of more abstract topics. Here, the sensemaking trigger comes in the midst of a coherent discussion of how to tag the two “surface type” comparison nodes that had come out of the previous few moments. At 13:52, participant E sees an opportunity to ask his recurring (abstract n the sense that it is commenting on the software itself, rather than on the subject matter of the exercise) question again, in a different form: “Well that’s—so that’s a question … .so in this tagging exercise are we allowed to have alternative or opposing views?” J jumps right in, echoing this kind of question: “And if you have opposing views how do you do it there in the tag?” At first, E’s question is absorbed in the discussion of how to tag the nodes, but then another participant, R, moves the discussion solidly in the abstract direction: “So far, I’m afraid, that we have introduced tags in such a way that you can’t question a tag.” Unlike the previous episode, however, this time no one jumps in to revert the discussion back to direct engagement with the map. Instead, spurred largely by K, the discussion moves to the relatively abstracted topic of how to think about tags in general.

Our analysis could not capture all of the narrative threads that perhaps were at work in a session. For example, we did not attempt to describe the individual “stories” (professional interests, emotional experiences, project trajectories) that each participant brought to a session, except when such information might have helped to shed light on the narrative framing or construction at work in the practitioner’s actions in a sensemaking episode.

Writing out a narrative description in this manner focused the analysis on the place each move or choice has in the way the sensemaking episode unfolds. We drilled down into even a finer level of detail with the grid analysis.

Grid Analysis

In the grid analysis for each sensemaking episode, we analyzed each practitioner/participant statement or representational move according to a number of criteria. This provided a fine-grained understanding of various dimensions of each move, such as the degree and kind of participant engagement with the representation at that moment; the engagement of the practitioner with the participants (e.g., acting in direct response to direction from a participant, or working off to the side to clean up some aspect of the map, or preparing for an upcoming event); the aspects of the setting on which practitioners were focused for that move (participants, maps, text, subject matter, surroundings, or process), and other factors. Mapping each move on the grid required careful consideration about what that move meant in the context of both the session as a whole and within the particular sensemaking episode, sensitizing the analysis in terms of the meaning to both participants and practitioners. Table 5 shows a portion of the taxonomy of concepts used in the grid analysis, derived from open and axial coding through repeated analyses of several long sessions.

The example grid analysis section shown in Figure 8 illustrates six practitioner moves: two verbal statements (at 14m47s and 14m51s) and four actions on the representation, at 14m46s, 14m48s, 14m51s (at the same time as a verbal statement) and 14m59s. Four of these moves were done with simultaneous focus on participants (engaged in conversation with them), maps (working on the form of the map), text (working with the text of the map’s icons), and the subject matter of the session, while one (the Link move at 14:59) is a shaping move on the map itself.

Table 5. Move-by-Move Analysis Schema for Grid Analysis.

Aspect	Description
Move Type	Assigns each practitioner move to a type in a taxonomy of moves in the *Compendium* software tool (e.g., Node Move-Arranging, Navigate-Map Open, etc.), or Verbal move types (Statement/Announcement, Acknowledgement, Query, Helpful Comment, Exclamation)
Participant Engagement with Representation	Characterizes the degree to which participants are paying attention to the representation during the move. Possible values: Active, Direct, Delinked, Partial, and Unclear. The Active value, which refers to moments when participants are directing the practitioner to perform particular actions on the representation, has the subtypes Text, Validation, Navigation, and Structure
Practitioner Response/ Engagement Mode	Characterizes the degree to which the practitioner is engaged with the participants during the move. Possible values: Direct, Semi-Direct, Indirect, Delinked. Delinked refers to moves when practitioner attention is focused completely on manipulation of the representation, not interacting or responding to the participants
Practitioner Focus	Characterizes what the participant is paying attention to and/or working with during the move. Can be (and often is) multiple. Values: Participants, Maps, Text, Subject Matter, Surroundings, Process

Time										Focus					
Start	End	Participant Statement	Practitioner Statement	Participant Engagement with Map	Screenshot	Practitioner Action	Move Type	Mini-Project	Response / Engagement Mode	Participants	Maps	Text	Subject matter	Surroundings	Process
1m56s long, from C14:32 – 16:28 (timeslots 21 – 26)															
14:32:00	14:48:00	E: Y'know and part of that actually is, is more sort of close interaction like with the people... like I love meeting astronauts, y'know .. and I still like meeting astronauts hearing about, hearing their stories and all that. [V27:42, C14:48]		Direct											
14:46:00				Direct		Creates an Idea node	Node Create-Answer			x	x	x	x		
14:48:00	14:59:00			Direct		Types "close interaction with with astronaust [sic] and rocket scientists" in the label field	Label Edit			x	x	x	x		
14:47:00	14:49:00		D: So close interaction with astronauts ...	Direct			Verbal Narration			x	x	x	x		
14:49:00	14:50:00	E: Or with the rocket scientists.		Direct											
14:51:00	14:59:00		D: [inaudible] ...or close interaction with those involved. Astronauts and rocket scientists, yeah. [V27:51]	Direct		Validation as the mapper (L.) typed the phrase "and rocket scientists", confirming that she had captured the point accurately.	Acknowledgement			x	x	x	x		
14:57:00	14:59:00	Bj: That old saying, it doesn't take a rocket scientist to...		Indirect											
14:59:00	15:01:00			Indirect		Links it to "What would inspire you to promote	Link				x				
14:59:00	15:00:00	E: Sometimes it does.		Indirect											

Figure 8. Excerpt from a grid analysis.

The grid analysis required very close inspection and increased sensitivity to nuances of the data. However, the process clearly demonstrated how much is going on when a skilled PDR practitioner is at work, supporting a team with the digital artifacts and rationale it needs as their deliberations unfold. Moreover, the grid analysis set the stage for characterizing practitioner actions and choices according to a set of criteria derived from the dimensions discussed earlier. We call this the *framing analysis*.

Framing Analysis

The framing analysis characterizes practitioner actions during the session in aesthetic, ethical, and experiential terms. It looks at how the practice and context interweave, and in what ways the aesthetic and ethical dimensions of the practice intertwine (McCarthy & Wright, 2004). We use this as the basis for a normative or ideal model against which we can hold up situations of practice (Aakhus, 2007; Aakhus & Jackson, 2005). Such a model could be used as a diagnostic tool to analyze what factors are preventing a situation from achieving its potential, or at least to characterize a practice situation in potentially useful ways.

The model used in framing analysis provides a set of components, elements, and exploratory questions to help determine how a context of service, the unique set of people, and the goals, constraints, situation, and subject matter can inform the shaping the practitioner performs on the representational object(s), and vice versa. Understanding and characterizing this has both normative (notions of what practice in such settings should be) and descriptive (how do we look at and characterize situated practice in service) aspects (Aakhus & Jackson, 2005).

The model contains three columns.[4] The first (leftmost) column shows the major categories or components of the practitioner's stance—his/her orientation toward various aspects of the situation or practice setting: the practitioner's towards him/herself and his/her own actions, towards the participants, and towards the situation as a whole. The middle column breaks down each stance into elements, each of which is explicitly related to the body of theory it arose from (largely from Bruner, 1990; Dewey, 1934/2005; Schön, 1983, 1987; and McCarthy & Wright, 2004). These elements constitute an ideal model of practitioner stance; that is, the model specifies the preferred conduct of a PDR practitioner as maintaining a dialogic orientation, fostering a heightened degree of connection between participants, the setting, purpose, and representation, and so on. The elements in turn generate descriptive (characterizing) or normative (evaluating) questions that can help guide the analysis of a particular setting, found in the rightmost column. The rightmost two columns of Component A of the framing model, which addresses the practitioner's own involvement in the situation, are shown in Table 6.

Considering the questions put forward in the framing model involved examining and reflecting on the analytical artifacts produced thus far. Since the framing analysis came last in the analysis sequence, by that time the analyst was very familiar with the specific occurrences in the video recording of the session, and particularly with the nuances of the behavior demonstrated by the practitioners during sensemaking episodes.

For example, in our Ames Group 2 case, we saw the following responses for component A.5, mediated objects and other interventions should preserve openness and dialogicity:

> *How do the actions of the practitioners inhibit openness and dialogicity?:* The prepared map appeared (and was said by participants afterward to be) too complex/involved for participants to engage with, although the mapping of the "needs" section

did seem to invite dialogue (unfortunately shut off by the mapper). The mapper's verbal intervention served to inhibit the nascent discussion about how to map the "needs" section.

In this case the practitioners needed either to be flexible in how the session would proceed, and evolve the map accordingly (with its extensive prestructuring that the participants were not paying attention to), or to intervene again to bring the session back to the course that they had intended. They could have brought the attention of the group to the portion of the map that contained the desired area of focus and created an effective way for the group to engage with it. As it happened, they stood by and waited to see if the conversation would come back to the intended course of its own accord (rarely an effective strategy).

Table 6. Component A of the Framing Analysis Model.

Element	Descriptive and normative questions
(A.1) Imposing their own coherence and values on a situation	What coherence is the practitioner imposing on the situation? What values is the practitioner imposing on the situation? In what ways are these congruent (or not) with those of the participants?
(A.2) Constructing narratives to account for how the situation arrived at the current pass; causes and breaches in canonicity	What is the narrative the practitioner is using to construct the situation? What is its degree of internal consistency? How evocative and inclusive is it? How useful is it?
(A.3) Eliminating prejudices, preconceptions, and personal desires in their work	What prejudices may be active? What preconceptions may be active? What personal desires or goals may be active?
(A.4) Personal authenticity in the practice setting	In what ways is the practitioner acting in an authentic manner (vs. received, affected, etc.)?
(A.5) Mediated objects and other interventions should preserve openness and dialogicity	How do the representations the practitioner constructs or modifies foster openness and dialogicity? How do they inhibit them?
(A.6) Artifacts should be clear, expressive, and helpful	How clear are the artifacts produced/modified by the practitioner? How expressive are they? How helpful are they within the context of practice?
(A.7) Perseverance in the face of checks and resistance	What checks to forward progress does the practitioner encounter? What resistance from participants, materials, etc. occurs? How does the practitioner respond in the face of these?
(A.8) Clear and focused communication	How clear is the practitioner's verbal communication? In what ways does the practitioner maintain focus on the aspects of importance in the situation?

CASE STUDY

The previous section of this paper introduced the various lenses we have been developing to make sense of PDR practice, illustrated with examples taken from a range of contexts. We now bring these together around a single design session, presenting brief examples of several of the above analyses to show how they provide different kinds of insight.

Setting

The setting for the session was a workshop for people interested in the Compendium[5] software tool for mapping multimedia information and design rationale (e.g., as IBIS maps), held at the NASA Ames Research Center, in May 2007. (Compendium is the descendant of the Graphical IBIS [gIBIS] design rationale prototype for mapping IBIS, questions-options-criteria [QOC], and other argumentation structures; Buckingham Shum et al., 2006; Conklin & Begeman, 1988). Half of one of the two days was given to a segment where less experienced practitioners could plan and facilitate a PDR session and get feedback from more senior practitioners. We divided the informants into groups of three to four and gave them the same general assignment and set of materials. We intended the practice task to be one that required neither expertise with real time use of the software, nor in the subject matter, so that the preparation and practice session could occur within a couple of hours without any advance knowledge on the part of the informants. We chose space travel as the subject matter (reasoning that it was a topic of general interest with which participants could be expected to have at least passing familiarity). We provided a set of 127 images inside Compendium that could be used in the exercise. Informants were informed that the sessions would be recorded for research purposes. They were given advance access to the task materials if they wanted to review them before the workshop.

Each group was given about 90 minutes to prepare (see Figure 9). Some groups included a more experienced practitioner who was allowed to help design and prepare the exercise but not to play an active part during the large group exercise itself.

Figure 9. Informants working on their materials for the large group exercise.

After the preparation period, each group took turns introducing and conducting their session with the larger group of participants. Typically each group had one person acting as the mapper (hands on the keyboard/mouse to control the Compendium hypermedia knowledge mapping software) and one as facilitator (guiding the discussion from in front of the room). Each group had 15 minutes to conduct their session, followed by a debrief discussion in which they also received feedback from the larger group.

In the following section we describe what one of the informant groups (Ames Group 1) encountered in their large group session. The process will be viewed through the lenses of the tools presented above.

Shaping Form Analysis

The shaping form analysis of Ames Group 1's large group session described how the practitioner team intended the session to proceed as an IBIS discussion of two central questions, for which they had also supplied seed answers. They also intended that each participant's contribution would be tagged with the participants' names, and that the participants would choose an image to correspond with their answer. There was no set outcome, just discussion mapping augmented with the tags and pictures. Both the facilitator and the mapper stayed directly engaged throughout the session. The mapper tried hard to capture all of the discussion on the map and to perform the ancillary tagging task. Both made interventions in the group process, slowing down the discussion at various times and asking for clarification. Both spoke directly to participants and appeared to be trying to get breadth and depth into the discussion as well as to let it and the map evolve. There were some environment issues having to do with how to use the software for elements like font size, which provided brief distractions. The session did experience some breakdowns, mainly when the mapper fell behind in creating a separate map to handle a rather abstract question that came from one of the participants (who himself was trying to understand why other participants kept steering the discussion away from the intended direction). The mapper was trying to perform a series of operations to do this, but new participant contributions came in while she was doing that and she fell behind.

CEU Analysis

Figure 10 shows the full-session CEU heat map for Ames Group 1. It is apparent from the heat map that timeslots 9-12, 19-22, and 26 contain some sort of anomaly or event that caused the coherence and usefulness scores to drop to the Low level.

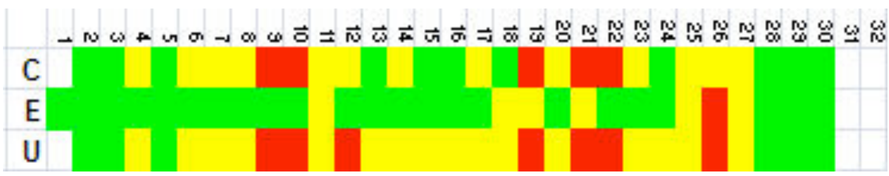

Figure 10. Heat map from Ames Group 1.

Figure 11 shows a fuller picture of the analytical grid used to develop the CEU ratings for timeslots 19–22 (and the recovery in timeslots 23–24). Here we see a narrative description of the events in each 30-second timeslot, the CEU ratings, and explanations of why each rating was given for each timeslot.

The CEU analysis pictured here provides context for finer-grained analysis of what happened in timeslots 17 through 23, the trajectory of a complete sensemaking episode, starting with a trigger and ending with the resolution.

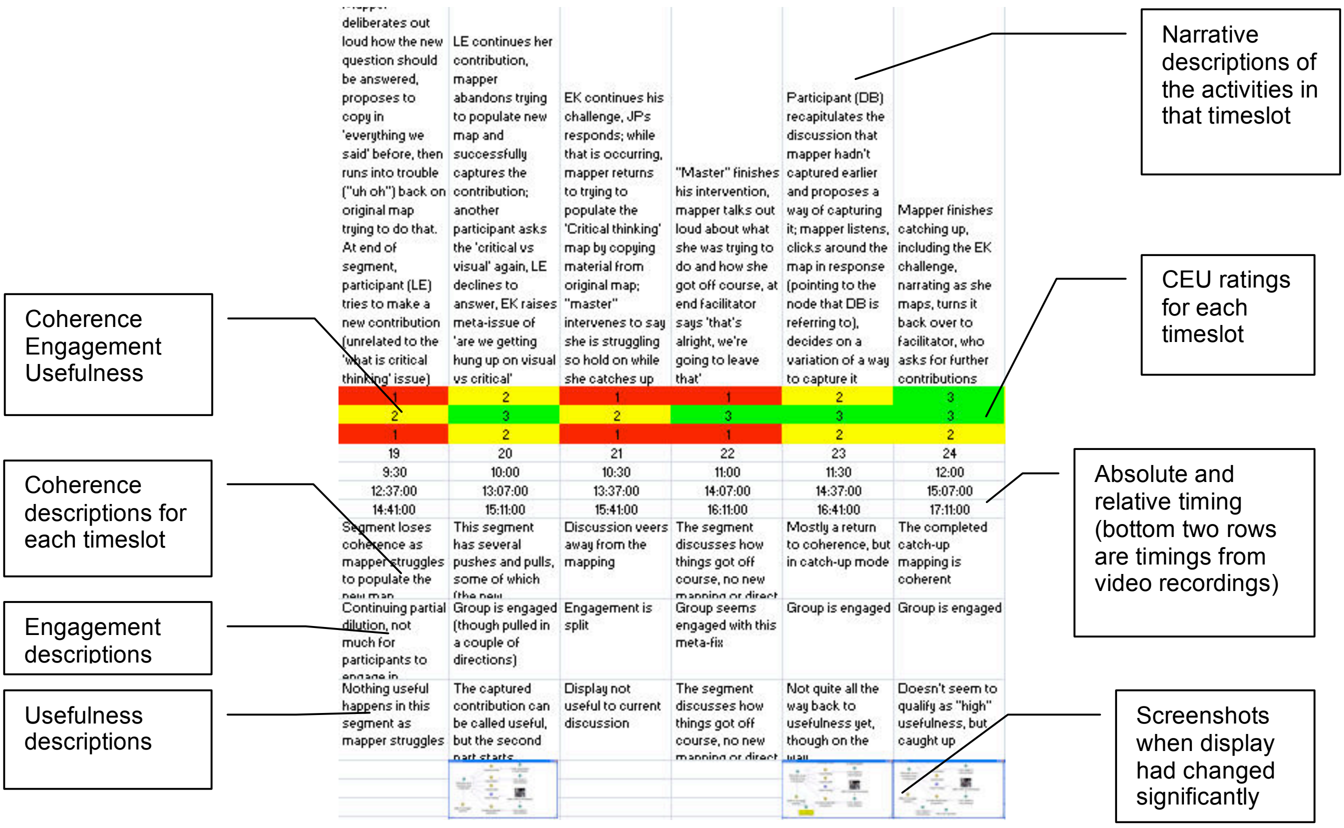

Figure 11. CEU ratings for timeslots 19-24.

Narrative Description of the Sensemaking Episode

The instance occurred for about 2.5 minutes of the 24 minute session, starting at 13:36 (timeslot 17) and lasting until 16:58 (timeslot 23). The session had proceeded more or less as planned until, at 13:36, one participant (P1) began to challenge some of the contributions to the overall discussion, questioning why some participants kept asking if others' contributions counted as "critical thinking" or "visual thinking" (illustrated on the screenshot in Figure 12).

The challenge did not fit into the expected flow of events, and the mapper, who up to that point had been able to capture participant contributions within the map quite fluidly, lost her way. This constituted the sensemaking trigger. Trying to make the structure of the representation

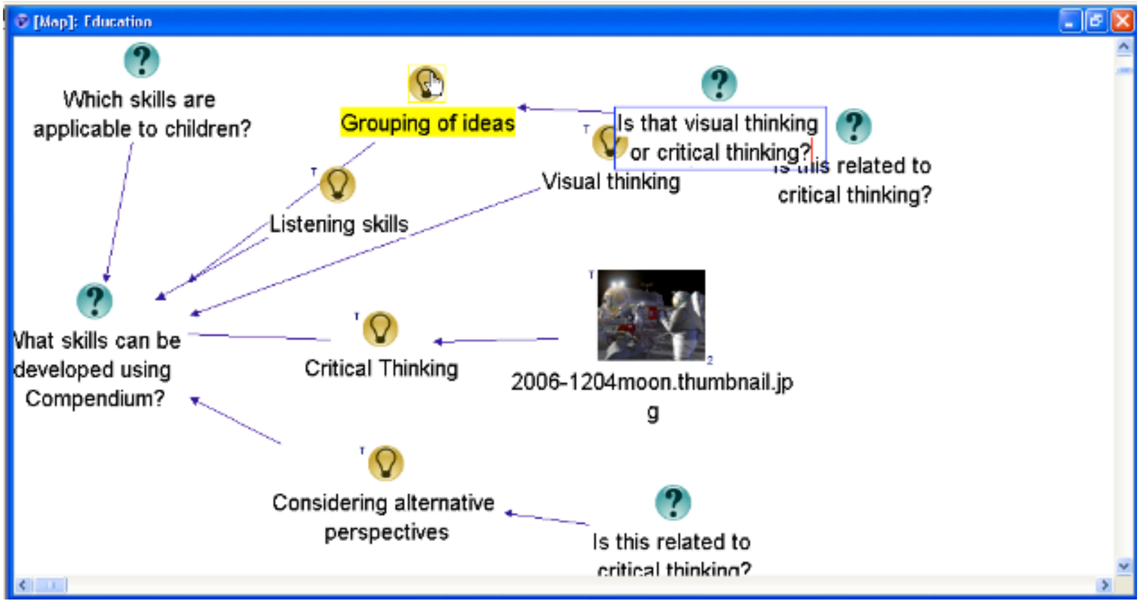

Figure 12. Ames Group 1 session: Map at 13m36s.

match the conversation when it veers from the expected course is a frequent challenge in PDR. In this case, this can be seen in Figure 13 in the three question nodes on the right half of the image. Often the planned structure does not seem to contain or fit what people are saying. She began trying to map P1's challenge at 13:49. At 14:42 she was in the midst of doing this when another participant (P2) made a new verbal contribution that did not reference the challenge.

A third participant, P3, asked if P2's comment counted as critical thinking or visual thinking, prompting a further challenge from P1. The mapper was able to capture P2's 14:42 contribution on the fly, but could not map either P3's question or P1's new challenge. In the course of this, the mapper got so far behind in mapping P1's challenge that she became stymied.

She faced two overlapping dilemmas. Firstly the participants' issue about how to frame the conversation itself, and secondly her own attempt to regain her momentum and resume making coherent additions to the map. The argument about critical versus visual thinking itself (and whether it was a fit subject for the session) can be seen as the collision of several competing narratives, some of which preceded the event, such as competing ideas for how such discussions should proceed. In this case, after some further back and forth among the participants, a fourth participant (P4) contributed a possible solution. After some negotiation about how much time was left in the session, the mapper asked the room for help in deciding what should be put onto the map. A fifth participant (P5) provided a helpful summary and suggestion for how to represent the discussion.

From that point until the end of the episode at 16:58, the mapper executed a rapid series of moves on the map, which enabled her to bring the map up to the point where it corresponded to the summary provided by participant P5, and to announce at 16:58, "I'm caught up."

In the excerpt, we see improvised actions that draw on practitioner (as well as participant) repertoires. Up to the point of the breach in timeslot 19, the mapper had followed a straightforward, preplanned dialogue mapping approach in her work on the knowledge map. When things went wrong, this had to be (temporarily) abandoned. With the help of several of the participants, the mapper was able to recast the situation, which helped her launch a rapid series of actions on the map to bring it back to a point where forward progress, and the dialogue mapping technique, could resume.

Framing Analysis

In the framing analysis, we relate aspects of the events in the episode to our theoretical framework, such as these aspects that emerged from the framing analysis for Ames Group 1:

> *What coherence is the practitioner imposing on the situation?*
>
> There were two types of imposed coherence: the practitioners' expressed desire for a "clean" discussion map, and attempts to keep the display coherent in the face of divergent and somewhat problematic contributions (in the sense of being hard to fit in or tending to pull the discussion off the intended course). There was a concerted attempt at coming up with coherent structures on the fly to encompass both the *primary contributions* (the answers to the seed questions) and the *meta contributions* about visual versus critical thinking, such as the participant statements at 13:36 and 14:51.
>
> *What values is the practitioner imposing on the situation? In what ways are these congruent (or not) with those of the participants?*
>
> There was a value of inclusiveness, of trying to map everything offered, but also a willingness to set some possible directions aside in the interest of cleanly mapping at least some of the contributions. The mapper had to choose between following all of the possible threads – particularly the pull toward the metaquestions about critical thinking – that some of the participants wanted to pursue versus focusing on cleanly mapping a few. Of necessity some participants' interests got shorter shrift as a result, such as a participant comment about "seeing bigger questions" at 14:42.
>
> *What is the narrative the practitioner is using to construct the situation?*
>
> The practitioners intended that a "clean" discussion would emerge "naturally" from the seed questions. A breach occurred when the actual discussion did not follow the intended form cleanly. The mapper was smoothly capturing the discussion as it happened, but then divergent input came in which required operations that (a) she was not quick enough at doing, or (b) did not seem to fit coherently.
>
> *How evocative and inclusive is it?*
>
> The preplanned, intended narrative set up a canonicity of a cleanly unfolding discussion, in which participants could provide tagged answers with images in response to the clear questions. However the answers started spawning a metadiscussion that broke down, and the practitioners were not able to be completely inclusive of all the contributions.
>
> *How do the representations the practitioner constructs or modifies foster openness and dialogicity?*

> The seed questions were appropriately open-ended, which lent itself to dialogue (e.g., "What skills can be developed using Compendium?"). The question nodes added during the session were partially so, though some were phrased as yes-or-no questions, which are less open (e.g., "Is this related to critical thinking?"). These were mostly done quickly in response to the metaquestions that were difficult to handle by their nature (e.g., "Is this visual or critical thinking?" or "Why do we keep getting hung up on this question?"). However, by deciding (after some back-and-forth) to link these questions to each answer they pertained to, the practitioners were preserving the opportunity to deal with the metaquestions later, recording them in answer to the concerns of some of the participants, while still keeping the intended discussion course going. A similar dynamic was gained by following the suggestion to open a separate map to address the metaquestion "What is critical thinking?" The possibility of exploring that question later remained open.

Summary

We see in these excerpts that the practitioner's actions can be characterized along multiple aesthetic and ethical dimensions. She had to make representational and process choices, which ultimately had consequences for which participant interests would be reflected in what ways. She had to temporarily abandon forward movement so that she could catch up, and reach out for help and suggestions for how to proceed. This proved successful, as she was able to get back on track.

SUMMARY AND DISCUSSION

In this paper we have described research that looks at the practice level of PDR: the wielding of DR tools in service to groups of people in collaborative, real-time settings. We can now revisit the research questions proposed earlier:

- (RQ1) What is the nature of the skills required to construct graphical knowledge representations in real-time, participatory settings?
- (RQ2) What are the kinds of choices practitioners face, especially at sensemaking moments in the course of conducting sessions?
- (RQ3) How does the context of the service being provided affect the choices a practitioner makes?

Rather than evaluating the PDR tools or methods themselves for RQ1, we took them as a given and focused instead on the human activity of creating the representations, especially on the skills needed and obstacles encountered in keeping DR artifacts coherent, engaging, and useful. For RQ2, we proposed a theoretical framework that has proven helpful in characterizing practitioner choices at sensemaking moments, and described the analytical tools that helped us examine video recordings of practice in light of the framework. For RQ3, we presented brief examples from some of our case studies describing instances of practitioner creativity and improvisation, often occurring as short "flashes," and seen examples of practitioners making aesthetic and ethical choices in the course of managing the multithreaded activities of a PDR session, including discussion of how the context and situation of practice informs the choices and moves practitioners make.

How does this kind of analysis advance matters? We propose that by looking closely at how actual sessions unfold, and treating their exigencies with the kind of attention given to other forms of professional practice, we elevate the activity of facilitating collaborative representation-making in groups (whether DR, or any other visual language) into a worthy research subject in its own right. This can provide a way for practitioners to deepen their understanding of their work by giving them a variety of means to analyze and reflect on their own practice, and can contribute to development of practitioner guidelines such as those common to other professional practices, such as the coherence guidelines for GSS facilitators proposed by Yoong and Gallupe (2002). The various analytical tools we have developed can contribute toward a methodology for characterizing the aesthetic and ethical dimensions of participatory media practice. These can lead to development of better approaches to practitioner education, thus "helping a student break into manageable parts what had at first appeared to be a seamless flow of movement" (Schön, 1987, p. 112). For example, the framing analysis could be used as a diagnostic tool to analyze what factors are preventing a situation from achieving its potential, or at least to characterize a practice situation in potentially useful ways.

The framework and analysis tools also shed light on the fostering of creativity in design meetings, particularly when rationale is being captured. As we have seen, people taking on the practitioner's role play a key part when teams encounter breakdowns and anomalies in the course of PDR sessions. Such moments, in small and sometimes large ways, can make the difference between success and failure of a design session. Failure can cause frustration and setbacks to a design effort, whereas success in swiftly resolving a breakdown frees up a team to bring their creativity to bear on the design problems rather than on "fixing" the meeting. We have seen how practitioner creativity can emerge when he/she intervenes in a session to restore its forward movement. At its best, practitioner creativity can result in choice, action, and materials seamlessly coming together to resolve the breakdown.

A potential contribution of this research is the development of a typology of dimensions of PDR practice, situations that a practitioner could face and the types of possible actions, such as the partial example in Table 7. Practitioners could use such a table to reflect on what did or did not happen in a particular session, considering the pros and cons of the different approaches given the context.

PDR practice is worthy of investigation in its own right and the methods outlined here provide a practical means and theoretical basis for doing so. These methods also point out how practitioners, tool builders, and consumers of PDR services can better understand how the micro, often tacit, dimensions of a practice shape the form and content of the product – namely, the rationale captured. Hence, aesthetics [p], ethics [q], narrative [r], sensemaking [s],

Table 7. Some Common Practice Situations and Example Actions.

Situation	Possible Actions
Participant topics or statements that do not fit the planned structure	Intervene in the conversation to bring it back to intended topic Evolve the structure on the fly Engage participants in direct reflection on the structure
Too much information and input coming too fast	Ask participants to slow down; be willing to intervene firmly if needed Capture as much as possible in background, wait for an opportunity to ask participants what was missed

and improvisation [t] are consequential for how matters are represented. With the approach described here, we can go much further in understanding how.

In future work we will draw on the foundational considerations outlined in this paper to develop concrete activities for practitioner education, as well as lessons for supporting tool design. We follow McCarthy & Wright's (2004, p. 62) argument that restoring the "continuity between aesthetic and prosaic experience" can reveal untapped and unexplored dimensions of the human experience of technology, for which more conventional approaches fail to provide tools for understanding. Using felt experience and an aesthetic viewpoint for technology use, they argue, would open up new possibilities for tool design. We will explore what general lessons, heuristics, and guidelines for practice can be drawn from the cases, and develop ways to help practitioners apply them to instances of practice. As a first step, we conducted a session at a gathering of graphic facilitators in August 2009.[6] Participants evaluated an instance of their own practice (a very different approach than PDR) using the CEU constructs, and reported that it helped them reflect in new ways about their actions.

Taking the practice level seriously means looking closely at what it takes to make sessions run well: how different practitioners overcome the experiential challenges involved in bringing a group of people through such an effort successfully. Partly because we have lived these challenges ourselves in many different contexts, and partly because we strongly believe in the benefits and potential of the approaches, PDR for us is already a professional practice deserving of careful study as an ongoing phenomenon rather than a start-up experiment, moving the question from "Can it work?" to "How can it work better?" Better tools and methods alone may help inculcate a broader interest in the practice level, but developing knowledge and expertise in the practice level can help bring about wider and more effective use of the tools and methods.

ENDNOTES

1. This case is drawn from an actual project.
2. For the purposes of this paper and project, we illustrated with hypermedia knowledge mapping software to capture the design rationale, but the same considerations apply to other sorts of DR approaches and tools.
3. Analysis artifacts from these studies are available on-line at http://people.kmi.open.ac.uk/selvin/analysis
4. A full version of the framing analysis model with discussion and citations is available at http://people.kmi.open.ac.uk/selvin/analysis/framing.pdf
5. Compendium Institute: http://compendium.open.ac.uk.institute
6. Voices of Visual Practice, 14th Annual International Forum of Visual Practitioners Conference, Montreal, Canada, August 5–7, 2009; http://www.ifvp.org

REFERENCES

Aakhus, M. (2001). Technocratic and design stances toward communication expertise: How GDSS facilitators understand their work. *Journal of Applied Communication Research*, *29*, 341–371.

Aakhus, M. (2002). *Design practice and transparency work in the technological facilitation of collaborative decision making*. Unpublished manuscript.

Aakhus, M. (2003). Neither naïve nor critical reconstruction: Dispute mediators, impasse, and the design of argumentation. *Argumentation*, *17*, 265–290.

Aakhus, M. (2007). Conversations for reflection: Augmenting transitions and transformations in expertise. In C. R. McInerney & R. E. Day (Eds.), *Re-thinking knowledge management: From knowledge objects to knowledge processes* (pp. 1–20). Dordrecht, The Netherlands: Springer-Verlag.

Aakhus, M., & Jackson, S. (2005). Technology, interaction, and design. In K. Fitch & R. Sanders (Eds.), *Handbook of language and social interaction* (pp. 411–436). Mahwah, NJ, USA: Lawrence Erlbaum.

Arnheim, R. (1967). Art and visual perception: A psychology of the creative eye. Berkeley, CA; USA: University of California Press.

Bruner, J. (1990). *Acts of meaning*. Cambridge, MA, USA: Harvard University Press.

Buckingham Shum, S. (1996). Analyzing the usability of a design rationale notation. In T. Moran & J. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 185–215). Mahwah, NJ, USA: Lawrence Erlbaum.

Buckingham Shum, S., Selvin, A., Sierhuis, M., Conklin, J., Haley, C., & Nuseibeh, B. (2006). Hypermedia support for argumentation-based rationale: 15 years on from gIBIS and QOC. In A. Dutoit, R. McCall, I. Mistrik, & B. Paech (Eds.), *Rationale management in software engineering* (pp. 111–132). Berlin, Germany: Springer-Verlag. Also available at http://oro.open.ac.uk/3032

Clancey, W. J., Sierhuis, M., Alena, R., Berrios, D., Dowding, J., Graham, J. S., Tyree, K. S., Hirsh, R. L., Garry, W. B., Semple, A., Buckingham Shum, S. J., Shadbolt, N., & Rupert, S. (2005, January-February). Automating CapCom Using Mobile Agents and Robotic Assistants. Paper presented at the American Institute of Aeronautics and Astronautics 1st Space Exploration Conference, Orlando, FL, USA. Available from: AIAA Meeting Papers on Disc [CD-ROM], Reston, VA, and as Advanced Knowledge Technologies ePrint 375: http://eprints.aktors.org/375, pp. 1–41.

Cohen, C. (1997). *A poetics of reconciliation: The aesthetic mediation of conflict.* Unpublished doctoral dissertation, University of New Hampshire, USA. Available on-line at www.brandeis.edu/ethics/coexistence_initiative/research_and_scholarship/reconciliation.pdf

Conklin, J., & Begeman, M. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *ACM Transactions on Information Systems, 6*, 303–331.

Dervin, B. (1983, May). An overview of sense-making research: Concepts, methods, and results to date. Paper presented at the annual meeting of the International Communication Association, Dallas, TX, USA.

Dervin, B. (1992). From the mind's eye of the user: The sense-making qualitative-quantitative methodology. In J. D. Glazier & R. R. Powell (Eds.), *Qualitative research in information management* (pp. 61–84). Englewood, CO, USA: Libraries Unlimited.

Dewey, J. (2005). *Art as experience*. New York: The Berkeley Publishing Group. (Originally published in 1934)

Fortner, R., & Christians, C. (1981). Separating wheat from chaff in qualitative studies. In G. Stempel & B. Westley (Eds.), *Research methods in mass communication* (2nd ed.; pp. 375–387). Englewood Cliffs, NJ, USA: Prentice Hall.

Kunz, W., & Rittel, H. (1970). Issues as elements of information systems (Working Paper No. 131). Studiengruppe fur Systemforschung, Heidelberg, Germany.

McCarthy, J., & Wright, P. (2004). *Technology as experience.* Cambridge, MA, USA: MIT Press.

Murray, K. (n.d.). Narrative partitioning: The ins and outs of identity construction. Retrieved May 3, 2006 from http://home.mira.net/~kmurray/psych/in&out.html

Rosson, M. B., & Carroll, J. (2009). Scenario-based design. In A. Sears & J. Jacko (Eds.), *Human-computer interaction: Development process* (pp. 1032–1050). Boca Raton, FL, USA: CRC Press/Taylor & Francis.

Sawyer, K. (1999). Improvised conversations: Music, collaboration and development. *Psychology of Music*, *27*, 192–205.

Sawyer, K. (2003). Group creativity: music, theater, collaboration. Mahwah, NJ, USA: Lawrence Erlbaum.

Sawyer, K. (2004). Creative teaching: Collaborative discourse as disciplined improvisation. *Educational Researcher*, *33*(2), 12–20.

Schön, D. (1983). *The reflective practitioner: How professionals think in action*. London: Basic Books.

Schön, D. (1987). *Educating the reflective practitioner: Toward a new design for teaching and learning in the professions*. San Francisco: Jossey-Bass.

Selvin, A. (2008). Performing knowledge art: Understanding collaborative cartography. In A. Okada, S. Buckingham Shum, & T. Sherborne (Eds.), *Knowledge cartography: Software tools and mapping techniques* (pp. 223–247). London: Springer-Verlag.

Selvin, A., & Buckingham Shum, S. (2008, April). Narrative, sensemaking, and improvisation in participatory hypermedia construction. Paper presented at the Sensemaking Workshop, CHI 2008: ACM Conference on Computer-Human Interaction, Florence, Italy. Available on-line at http://oro.open.ac.uk/19039

Selvin, A., & Buckingham Shum, S. (2009, April). Coherence, engagement, and usefulness as sensemaking criteria in participatory media practice. Paper presented at the Sensemaking Workshop, ACM Computer-Human Interaction (CHI) Conference, Boston, MA, USA. Available on-line at http://oro.open.ac.uk/12910

Sierhuis, M., & Buckingham Shum, S. (2008). Human–agent knowledge cartography for e-Science: NASA field trials at the Mars Desert Research Station. In A. Okada, S. Buckingham Shum, & T. Sherborne (Eds.), *Knowledge cartography: Software tools and mapping techniques* (pp. 287–305). London: Springer-Verlag.

Strauss, A., & Corbin, J. (1990). *Basics of qualitative research: Grounded theory procedures and techniques*. Newbury Park, CA, USA: Sage.

Tracy, K. (1989). Conversational dilemmas and the naturalistic experiment. In B. Dervin, L. Grossberg, B. O'Keefe, & E. Wartella (Eds.), *Rethinking communication: Volume 2. Paradigm examples* (pp. 411–423). Newbury Park, CA, USA: Sage.

Tripp, D. (1993). *Critical incidents in teaching: Developing professional judgment.* New York: Routledge.

Weick, K. (1995). *Sensemaking in organizations*. Thousand Oaks, CA, USA: Sage.

Weick, K., & Meader, D. (1993). Sensemaking and group support systems. In L. Jessup & J. Valacich (Eds.), *Group support systems: New perspectives* (pp. 230–252). New York: Macmillan.

Yoong, P., & Gallupe, B. (2002). Coherence in face-to-face electronic meetings: A hidden factor in facilitation success. *Group Facilitation: A Research and Applications Journal, 4*, 12–21. Available also on-line at http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.125.4454&rep=rep1&type=pdf#page=15

Authors' Note

Some of the data analysed in this article were gathered from eScience/robotics field trials funded by NASA (*Mobile Agents* project) and the UK EPSRC (*Collaborative Advanced Knowledge Technologies in the Grid* project), to whom we are indebted.

All correspondence should be addressed to
Albert M. Selvin
Verizon Information Technology
Knowledge Media Institute
Open University
Milton Keynes UK
alselvin@gmail.com

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

HUMAN TECHNOLOGY

An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889

www.humantechnology.jyu.fi Volume 6 (1), May 2010, 106–128

USING RATIONALE TO ASSIST STUDENT COGNITIVE AND INTELLECTUAL DEVELOPMENT

Janet E. Burge
Department of Computer Science and Software Engineering
Miami University, Oxford, Ohio
USA

Bo Brinkman
Department of Computer Science and Software Engineering
Miami University, Oxford, Ohio
USA

Abstract: *One of the questions posed at the National Science Foundation (NSF)-sponsored workshop on Creativity and Rationale in Software Design was on the role of rationale in supporting idea generation in the classroom. College students often struggle with problems where more than one possible solution exists. Part of the difficulty lies in the need for students to progress through different levels of development cognitively and intellectually before they can tackle creative problem solving. Argumentation-based rationale provides a natural mechanism for representing problems, candidate solutions, criteria, and arguments relating those criteria to the candidate solutions. Explicitly expressing rationale for their work encourages students to reflect on why they made their choices, and to actively consider multiple alternatives. We report on an experiment performed during a Data Structures course where students captured rationale.*

Keywords: *design rationale, creativity, student cognitive development.*

RATIONALE AS A METHOD FOR BUILDING CREATIVITY AND COGNITIVE MATURITY

One of the orienting questions for the National Science Foundation (NSF)-sponsored workshop on Creativity and Rationale in Software Design in 2009 was, “How can design rationale be used in the classroom to motivate and instruct students about reflection, idea generation, and evaluation?” (Daughtry, Burge, Carroll, & Potts, 2009). At the heart of this question is an implicit claim about creativity, that is, “creativity” in software design seems to involve not just idea generation itself, but also the iterative process that moves the designer to reflect, evaluate, and generate more ideas multiple times before committing to a final design.

Carroll’s (2009) workshop manifesto, “The Essential Tension of Creativity and Rationale in Software Design,” emphasized this by pointing to liminality as a key aspect of the creative design process. The manifesto described liminality as “Thinking and acting on the border between two contrasting concepts or rules, such as rapid switching between convergent and divergent modes of thinking.”

URN:NBN:fi:jyu-20105241909

We see a direct link between a student's cognitive development and the ability to engage in creative processes. Perry (1970) identified nine positions of development starting with duality, where answers exist for everything and where they can be right or wrong, into multiplicity, where all answers are valid, into relativism, where they begin to evaluate solutions based on the context, and continuing through several levels of commitment, where students can begin to integrate knowledge and make their own choices based on that information.

Students in the first two years of college tend to display dualistic and multiplistic tendencies. Though Perry pointed out that most college students are not pure Stage 1 dualists, few students in his study reached even the lowest levels of commitment until their junior year (p. 155). Similarly, Marcia B. Baxter Magolda reported (1992, p. 71) that more than 80% of juniors are "transitional knowers," (those that recognize relativism in some knowledge domains, but are still dualistic in others), and more than 40% of sophomores were still mostly dualistic.

This understanding of the epistemic styles of our students should inform our thinking about teaching design. Dualistic cognition is inherently opposed to the liminal state of mind that is so characteristic of creativity in design. We believe that students who come to a design problem with the attitude that there is a "right answer" to be discovered by analysis will commit to a design without engaging in reflection or iteration. They will commit too early, before they have a chance to be creative. Students in multiplicity or the early stages of relativism may be unable to distinguish between the good designs and poor designs that emerge in their thinking. Some evidence to support these claims can be found in the work of Atman, Cardella, Turns, & Adams (2005), who showed that senior engineering students spend 2 to 3 times more time on a design problem than freshman engineering students. This also correlates highly with the quality of the final solution, though their results do not directly address our claim that these effects are due, in part, to student epistemic styles.

We propose that requiring students to generate design rationale prior to implementing their solutions is a mechanism for encouraging reflection and delaying commitment to their initial design choices. Design rationale, the reasons behind decisions made while designing, is a way to represent design alternatives and the deliberation that produced them. In a sense, the rationale can be considered a language of design (Dym, Agogino, Eris, Frey, & Liefer, 2005), much like sketching (which captures structural aspects of design) or mathematics (which expresses constraints the design must conform to). In the case of rationale, this is a language that captures the design intent and its relationship to the design. The ability to analyze and evaluate design alternatives in terms of their success at achieving design goals (intent) requires higher order thinking skills.

In response to our orienting question, we claim that design rationale help to motivate and instruct students in the creative process by putting off the moment of commitment to a design. The time spent in the liminal phase of design, iterating from idea to evaluation and back, can be lengthened by the use of design rationale. A prospective design rationale (that is, design rationale built before implementation, as a method of exploring possible designs) serves as a way of documenting the designer's process of design.

This lengthening of the time the student spends in ambiguity and reflection should also lead to cognitive development, by forcing the student to experience the kinds of reflection and switching between modes of thought that are characteristic of higher levels of cognition.

In the rest of this paper we explore more fully the following two questions:

1) What are the links between creativity in design and cognitive level, and how can rationale assist in developing a student's capacities for each?
2) How can we assess whether or not use of rationale has had the intended effect?

In the balance of this first section we explore the first question. First we describe the motivation for teaching creativity in software development, and then expand on our proposition that "liminality" links software design with cognitive development. Next we discuss the use of design rationale as a pedagogical tool for encouraging cognitive development through reflection, and describe some prior applications of rationale to education.

Later in the paper we describe an experimental assignment we designed based on our ideas, and provide some initial assessments of our approach. In the final two sections we outline areas for future work and other ways that design rationale may be used to stimulate student cognitive development.

Creativity in Software Development

Software development is, at its core, a creative enterprise. Given a problem, there are many possible solutions. For some practitioners, this is what attracts them to the field—software development as an exercise in creative design. For others, especially as college students, the multitude of solutions, where there is often no clear "right" answer, can be a source of frustration. With the many demands on their time, both curricular and extracurricular, there is significant pressure to find the, or a, correct solution in as little time as possible. The skill of being able to understand just enough about the material to come up with an answer serves them well in some of their earlier courses, where a program is correct if it produces the correct set of outputs given a set of inputs. But they run into difficulty in their later courses, where solutions need to be analyzed on multiple dimensions. These difficulties are exacerbated in courses such as Software Engineering and Human–Computer Interaction, where the system design is influenced not only by the technology available but by how people intend to use it.

It is essential that computer scientists, and computer science students, think creatively in order to successfully develop software. Glass (1995) described several aspects of software development where creativity is critical: determining how to translate the customer/business needs into a problem that the software can solve; resolving stakeholder conflicts; designing solutions to new and complex problems; determining test cases; and enhancing existing systems to meet needs that were not initially anticipated by the customer or the developers. A student convinced of a single right answer is likely to either insist that the stakeholder(s) provide this answer (when the stakeholders may not be approaching the problem with an awareness of what is possible with the technology available) or insist that their solution is the only one, or the best one, even if it may not be acceptable to the client.

Liminality, Creativity, and Cognitive Development

The workshop manifesto (Carroll, 2009) emphasized three major characteristics of creativity in software design: playfulness, empathy, and liminality. We have chosen to focus on the liminal aspects of creativity because it seems to be the most natural fit for freshman- and sophomore-level courses.

To be sure, many instructors have great luck incorporating playful or empathic approaches in their coursework; many such assignments are presented every year at the SIGCSE (Special Interest Group on Computer Science Education) conference. But, for most students, Data Structures is the first required course that explicitly teaches a set of mathematical tools that can be used to compare one solution to another. Here we are speaking of the use of asymptotic analysis to compare the time and space requirements of data structures. When applied to simple problems, like sorting, such analyses seem definitive: For example, "Randomized Quicksort is more efficient than Insertion Sort." But when designing a data structure for a realistic problem, it is often the case that some operations can only be made fast if other operations are made slow, or if excessive amounts of memory are used, or if auxiliary data structures are used for bookkeeping. This means that, as a data structure is designed, there are many opportunities to shift focus from one operation to another, and to shift from analysis to idea generation and back. The manifesto links this "rapid switching between convergent and divergent modes of thinking" to creativity.

A concrete example will help clarify our point. In the experimental assignment described more fully below, students were asked to design a list-like data structure that needed to support dequeue operations (adding and removing items at the ends) as well as searching by key. One student, in his initial thinking, considered only arrays and linked-lists as possible designs, and selected linked-lists because they support dequeue operations in constant time. Upon evaluation of the designs, however, he discovered that search would be very slow, and so he returned to idea generation and added hash tables as a third design option. Upon evaluating hash tables he discovered that the dequeue operations would be tricky to implement, and returned again to idea generation.

Inspired by this example, we propose that a Data Structures course is a natural place to look for the contrasting concepts that give rise to liminal mental states. In Data Structures we teach the theory of algorithm running times, but also how to actually determine algorithm performance through experiments to confirm (or not!) the theory. We teach the canonical data structures, but we also give students problems for which the canonical data structures are a poor fit. We present the material of the class using diagrams and pseudo-code, but require students to actually write working programs using a real language.

We do not wish to define *creativity* only in terms of liminality, but we feel that much of what is creative about the work of students really arises when they are able to synthesize seemingly incompatible ideas from two apparently opposing or unrelated ways of thinking. In reference to the manifesto (Carroll, 2009), we claim that students are best able to "pursue surprise and unexpected outcomes" when they actively embrace and explore the "border between contrasting ideas."

We believe the ability to embrace liminal states and cognitive development are directly linked. Many useful theories of cognitive development might inform this discussion. We have already described the key aspects of Perry's (1970) model, which undergirds much of our thinking in these early sections. In our final section, we also use Bloom's Taxonomy (Bloom, 1956), which we found helpful in identifying other pedagogical applications of design rationale. The evidence of Perry (1970, p. 55–56) and Baxter Magolda (1992, p. 71) suggests that our Data Structures students (who are mostly sophomore computer science majors and junior engineering majors) will still be in transition towards relativism. Baxter Magolda's study showed that more than 40% of sophomores were still noticeably absolute in their thinking, and that very few juniors (less than 10%) are independent thinkers. In Perry's study juniors were rated as being in "commitment" (levels 7, 8 or 9) only about 50% of the time, and for sophomores it was less than 10%.

Students stuck in a dualistic way of thinking are unlikely to discover creative solutions, because they will be satisfied as soon as they identify any "correct" solution. The traps for students in multiplicity or naïve relativism are subtler. At this level, the student is aware that there are many viable solutions, but tends to assume that all are equally good. This can again block creativity because the student chooses a solution somewhat arbitrarily. When students are "stuck" at the lower levels of cognitive development, we suspect that the solution chosen is likely to be routine, familiar, or arbitrary, rather than innovative and creative.

So we propose that there is a link between comfort with liminal mental states and cognitive maturity, and that design activities that cause students to experience rapid switching between contrasting ideas help students to build up both cognitive and creative maturity.

Rationale, Reflection, and Liminality

In the experimental assignment sequence presented in the next section, we used prospective design rationale to encourage student creativity in an individual design task. As mentioned above, a prospective design rationale is one that is created before the design is implemented, as part of the design process. Contrast this with retrospective design rationale, which are written after the design is chosen, and may serve only to document the chosen design. Prospective design rationale fosters both creativity and cognitive development by encouraging, and capturing evidence of, reflection.

Reflection serves an important purpose in both education and in practice. In education, many researchers have proposed a link between reflection and cognitive/epistemic level. Dewey (1933, p. 9) defined reflective thinking as "active, persistent, and careful consideration of any belief or supposed form of knowledge in the light of the grounds that support it and the further conclusions to which it tends." Reflection guides the learning process as evidence is examined and conclusions drawn. Dewey's claim that reflective thinking is necessary when it is not possible to come up with "certain solutions" was the reason why King and Kitchener (2002) chose reflection as the basis for their model of student epistemological development. The reflective judgment model (King & Kitchener, 1994) defined seven stages of student epistemological development, broken into three categories: prereflective thinking, quasi-reflective thinking, and reflective thinking.

Schön, in his book *The Reflective Practitioner* (1983), described the need for professionals to move beyond technical rationality, where problem solving is the application of theory, to processes that allow for uncertainty and conflict. He described "knowing-in-action," where practitioners act based on tacit knowledge, and "reflection-in-action," where practitioners reflect on what they are doing as they do it.

Fischer, Lemke, McCall, & Morch (1991) described how design rationale supports reflection by capturing the designer's knowledge about the situation. Similarly, the iteration between idea generation (divergent thinking) and design selection (convergent thinking) is a reflective process. Design rationale supports both the capture of the alternatives and their exploration by supporting the evaluation of the more promising alternatives and any additional decisions required during their elaboration. In the illustrative example above we saw a student using the process of building design rationale as an opportunity for critical reflection.

We should note, as an aside, that in this study we focus only on the individual design projects, not on teamwork. Though we greatly appreciate the role that rationale can play in capturing and transferring knowledge in a team setting, we believe that the capture of rationale is beneficial even for one individual engaged in an individual design project.

So we claim that design rationale can be used to encourage critical reflection about software design problems. Further, we claim that such critical reflection, if embraced by the student, is likely to lead to greater creativity. Critical reflection and creativity are certainly not the same thing; rather, critical reflection tends to provide grist for creative energies to act upon. Incorrect assumptions tend to act as roadblocks for creativity, but critical reflection helps us to challenge these assumptions. We naturally tend to select designs similar to older successful designs with which we are already comfortable, but critical reflection can cause us to reject familiar solutions that are actually inappropriate.

Prior Work on Rationale in Education

Moran and Carroll's (1996) book included two approaches to using rationale in education. The first was to provide rationale in the form of templates to assist with user interface (UI) design (Casaday, 1996). The templates help designers to "ask the right questions" and assist designers with the process by guiding them toward a solution. Carey, McKerlie, & Wilson (1996, p. 375) built a library of "exemplary user-interface designs" along with their rationale so those examples could be used to teach UI design. Other work using rationale in UI design includes using design space analysis (DSA; MacLean, Young, Bellotti, & Moran, 1991) as part of the FLUID (framework for learning user interface design) interactive media system (van Aalst, van der Mast, & Carey, 1995). The work proposed here uses a more general approach (not one aimed at a specific type of design) and supports additional manipulation and evaluation of design criteria, as well as using rationale to assist with the definition and documentation of new designs.

Several software engineering textbooks either teach rationale (Bruegge & Dutoit, 2004) or use rationale as explanation for design case studies (Fox, 2006). Rationale is also present in the form of "consequences" in the ubiquitous Gang of Four (GoF) design patterns book (Gamma Helm, Johnson, & Vlissides, 1995), used both as a reference and as a supplemental textbook.

EXPLORING RATIONALE IN A DATA STRUCTURES COURSE

In the previous section we claimed that careful use of design rationale by dualistic and multiplistic thinkers should lead them to increases in creativity and cognitive maturity. This theory has implications for how one structures "design" projects for lower-level courses. In this section we will present a first attempt at such an assignment for a Data Structures course, and contrast it with the kinds of design assignments we had used in the past.

Our theory also requires some justification through evidence. We have some initial results based on our evaluation of the work produced by students for our experimental assignment using design rationale. While the experiment was by no means a controlled experiment, nor was it designed to validate our theory (it was, instead, designed to help the students learn), we still are able to report on some tantalizing results that point the direction for future work.

SEURAT and Pugh's *Total Design* in Data Structures

In the Data Structures course, we chose to use two different methods for capturing prospective design rationale. The first was the rationale management system SEURAT and the second was based on examples from Pugh's (1991) *Total Design*. In order for the reader to understand how we think rationale should be used in undergraduate courses, we must first describe what data these two types of design rationale capture, and how they support decision making.

Let us start with some general observations. Problem solving can be broken into four stages: problem definition and analysis, idea generation, idea evaluation and selection, and implementation of the selected idea (VanGundy, 1981). Rationale can support some idea generation techniques, such as brainstorming, by representing alternatives as generated, and attribute listing, a technique developed by Crawford (VanGundy, 1981), where attributes listed would be alternatives. Rationale captured in the form of argumentation is especially useful, however, during the evaluation and selection stage by capturing criteria, their relationship to the alternatives, and supporting evaluation. Some of the techniques described by VanGundy (1981) that could be supported by rationale are (a) the advantage–disadvantage approach, enumerating the advantages/disadvantages of each alternative with respect to a predefined set of criteria; (b) the Battelle method (Hamilton, 1974; VanGundy, 1981), dividing criteria into culling, rating, and scoring in order to narrow the field of alternatives; and (c) reverse brainstorming (VanGundy, 1981; Whiting, 1958), which is brainstorming on the disadvantages of each alternative. Rationale systems that perform evaluation, such as the software engineering using rationale (SEURAT) system (Burge & Brown 2004), can be considered a type of weighting system (VanGundy, 1981), by allowing weights to be assigned to the criteria and using those weights in evaluation.

In this work, we use argumentation-based rationale to capture the idea generation, idea evaluation, and selection stages of problem solving. We used two methods for representing rationale, SEURAT (for one experimental group) and written documents proposed in Pugh's total design methodology (for the other). Both methods require students to list many alternative designs, develop criteria by which to evaluate the designs, perform the evaluation, and select a solution. Both methods, furthermore, require argumentation to back up both the evaluation criteria and the final decision. SEURAT adds the additional capabilities of expressing the rationale in a hierarchical format, showing decisions and subdecisions, as well as providing the capability to calculate a numerical evaluation of the support for each alternative.

Software Engineering Using RATionale (SEURAT)

The SEURAT system (Burge & Brown, 2004) is a rationale management system (RMS) originally developed to assist with software maintenance by providing ways that the rationale could be used beyond just its presentation. SEURAT captures rationale as structured argumentation (decision problems, decision alternatives, and arguments) and uses both the structure of the rationale (syntax) and the content (semantics) to inference over the rationale to detect incompleteness (of the rationale) and inconsistency (of the design). The arguments in SEURAT can refer to system requirements, desired qualities, assumptions made, and relationships between alternatives. Figure 1 shows some rationale captured in the SEURAT Rationale Explorer. SEURAT stores the rationale in a relational database, allowing the rationales to be shared between multiple users during collaborative decision-making.

Figure 1's example shows three decisions, taken from the rationale for a conference room scheduling system. The decisions are displayed using a diamond-shaped icon containing a double-headed arrow. The second decision, "How do we know what the conference rooms are?" has a warning icon overlaid on it. This is because the alternative selected, "ascii file giving a list of room names," is not as well supported as the other candidate alternative, "serialized vector of room objects." The third decision, "How is the user associated with the meeting," has an error icon because none of the proposed alternatives has been selected yet.

The students who used SEURAT in the experiment were given a tutorial on how rationale are entered into SEURAT and how they could use SEURAT's ability to evaluate alternatives to assist them in their decision-making. The students were instructed to enter the functional and nonfunctional requirements that applied to the problem they were solving and then to enter the decisions, alternatives, and arguments. They were instructed to use their requirements in arguments, rather than utilizing the other types of arguments supported by SEURAT.

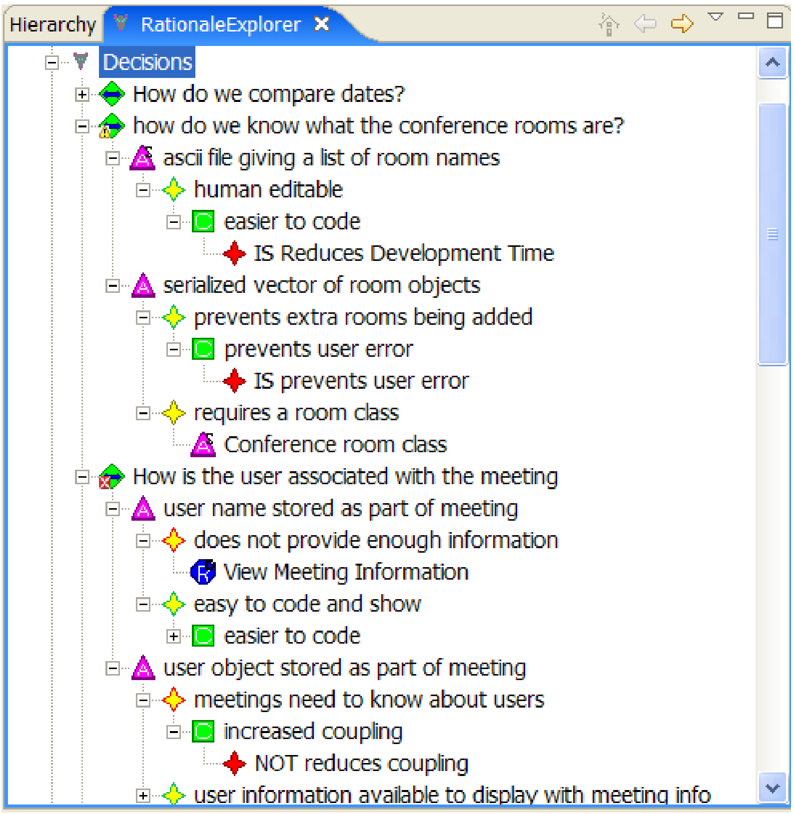

Figure 1. An image of the SEURAT Rationale Explorer, showing the hierarchical view of a structured design rationale.

Pugh's *Total Design*

The other section of the class used a set of design documents based on Stuart Pugh's (1991) book *Total Design*. It is not proper to say that we used "his" documents, because Pugh goes to great lengths to show many different types of documents that might be useful. In particular, we took advantage of three main parts of his approach:

1. A product design specification (PDS) listed the criteria by which designs should be judged. Each student constructed his or her own list of criteria, resulting in a bulleted list with argumentation that was very similar to the SEURAT group, but created in a word processor instead of the SEURAT RMS.
2. A Pugh Matrix was used for idea evaluation. The student created a two-dimensional table in a word processor. Each column corresponded to one of the designs, and each row to one of the criteria from the design specification. The student selected one design to be the "baseline" design, and then each design was compared to the baseline in each criterion. A plus symbol was entered in the table if the design in question was superior to the baseline on that criterion, a minus symbol if it was worse. We also instructed students to enter their arguments in support of their evaluations in each cell.
3. A short essay summarized the idea selection phase and provided arguments in favor of the student's solution, based on the evaluation in the Pugh Matrix. Note that Pugh was very clear that one should not just count up the number of plusses and minuses and then select based on the numerical answer that results. We instructed students to use their evaluation matrix to support their selection process, but to also use common sense.

Pugh's process has many commonalities with SEURAT. Design criteria are explicitly listed, and require argumentation. Designs are evaluated based on the criteria, and arguments supporting those evaluations are captured. The Pugh process supports a quasi-quantitative approach to selecting the final design.

There are major differences as well. On the negative side, SEURAT requires students to learn a new tool (the Rationale Explorer, which is a plug-in for the Eclipse Integrated Development Environment) instead of using a familiar tool (Microsoft Word). On the positive side, SEURAT forces students to be more careful about linking criteria with design decisions. In the non-SEURAT group, some students used arguments in their Pugh Matrix that had no relationship to their criteria, something that is much harder to do in SEURAT. SEURAT also naturally leads one to represent subdecisions in a hierarchy under the main decisions, much like an outline. The Pugh Matrix places all decisions at the same level.

Design in Data Structures Before Design Rationale

For several years we have had design projects in Data Structures similar to the one described here. In the past, however, students simply submitted a retrospective design justification. These documents took a variety of forms, but none of them were particularly formal, and only in very rare cases did the students compose them before implementing their solution. We found the quality of the resulting programs written by the students to be quite disappointing. In particular, there was some anecdotal evidence that students would not consider all of the important design criteria at the start of the project, but focus on only a few. Theses students tended to select familiar or canonical data structures because they never discovered the trade-offs involved in the real problem until *after* the solution was implemented.

Our feeling was that by introducing design rationale, and specifically prospective design rationale, into the Data Structures course we could cause the students to delay committing to a solution, and give them more opportunities to fully understand the problem.

Data Structures Class Experiment

Our main goal was to explore whether or not rationale could be of benefit to students in 2nd-year computer science coursework. As noted in the manifesto (Carroll, 2009), it could be that use of rationale would "limit creativity by anchoring thought"; it could also be that rationale would be viewed as busywork, or that time spent building the rationale would take away time from honest reflection and other creative activities. We expected, however, that students would actually spend more time in reflection if they had to build a full rationale than if they simply had to write a brief essay explaining their choice.

We designed a classroom activity in which students needed to design a solution to a data structures problem based on their understanding of the performance characteristics of various common data structures. The assignment was broken into 5 steps (see Table 1), each of which had its own delivered artifact.

The fifth step of the process is also meant to test whether students overcommit to their chosen solution, and refuse to change when criteria change.

The experimental subjects were 38 students (34 male, 4 female) in an undergraduate course on data structures and data abstraction (most students were in their 2nd or 3rd year of college). The difference between the two experimental groups was in Step 3. The first experimental group (henceforth the "SEURAT group") constructed design rationale using the SEURAT system. The second experimental group (henceforth the "Matrix group") used a version of Pugh's total design methodology (Pugh, 1991, Section 4.8).

Table 1. Description of the Main Problem and Stages of the Assignment Used in the Experiment.

Problem:	Design a list-like data structure that supports the following operations: Adding and deleting at the head and tail of the list, searching to find the index of the first data item matching a search term, and retrieving an item based on its index in the list.
Step 1:	List the criteria that you want your solution to adhere to. For example, do you want to have constant time searching? Do you want to try to minimize time spent coding?
Step 2:	Make a list of possible alternative implementation strategies. For example, a linked list would support all the operations, though not very efficiently. A hash table, on the other hand, can be made to be very efficient, but most students would find implementing it to be too challenging.
Step 3:	Create a design rationale expressing the tradeoffs between various alternatives in terms of how well they meet your criteria.
Step 4:	Select one of your alternatives, and implement it.
Step 5:	Write a paragraph explaining which alternative you would have selected if the "most important" criterion was removed.

We collected three artifacts from each participant: Their rationale (generated in Steps 1–3), their computer program (generated in Step 4), and their paragraph explaining their response to changing criteria (generated in Step 5). Note that, for the SEURAT group, the rationale could be fully captured in SEURAT, but the rationale for the Matrix group consisted of a list of evaluation criteria (with argumentation), a list of possible designs, and an evaluative matrix (henceforth the "Pugh Matrix"). Table 2 provides the metrics used to evaluate both sets of rationale in terms of rationale quality, and Table 3 provides the metrics used to evaluate the ideation skills demonstrated.

This first set of metrics, R1–R4, is meant to judge student success on the assignment in terms of their mastery of course objectives, as defined by the instructor. A score of 3 points or 2 points indicates that the student met instructor expectations, 1 or 0 indicates failing to meet expectations.

R1, R2, R3, R5 and R6 evaluate the rationale. R4 evaluates the response to changing criteria, and R7 evaluates the computer program code.

Examples of Student Artifacts and Reflections

In order to make this discussion more concrete, we present some small examples of student work. We will show some examples of creative designs from the experiment, as well as some examples of student argumentation.

What Kind of Creativity is Expected/Possible in Data Structures?

First, we wanted to provide some examples of creative solutions to the design problem. Recall that the student needed to design a list-like data structure that supports adding to the head and tail, looking up items by index, and searching for the first occurrence of a particular item.

Table 2. Data Structures Assignment Learning Metrics.

Metric	Excellent/High (3 pts)	Good/Medium (2 pts)	Poor/Low (1 pt)
R1: Are all relevant alternatives identified and provided?	The student provides all the relevant alternatives	The student provides most of the relevant alternatives	The student only produces one alternative
R2: Are the criteria appropriately mapped to the alternatives?	The student maps all the criteria to the correct alternatives	The student maps most of the criteria to the correct alternatives	The student does not successfully map criteria to alternatives
R3: Did the student select an alternative based on the rationale?	The student selects an alternative based on the level of support	The student selects some alternatives based on the level of support	The student did not appear to have reasons for making the selection.
R4: Did the student change the decision after the criteria change?	The student looks at differences in support levels and changes the decisions	The student sometimes fails to change the decision but instead stays with the initial plan	The student did not acknowledge the effect of changing criteria

Note: Each student received a score between 3 (for excellent) and 0 (for incomplete).

Table 3. Data Structures Assignment Ideation Metrics.

R5: Completeness	For each alternative in the following list, the student receives 1 point: Array, Linked List, Vector (or Array-List), Skip List, Hash Table, and Binary Search Tree. These are all of the data structures studied in the class (to that point) that would have been reasonable alternatives for the assignment. This scale is meant to measure the *quantity* of a student's candidate solutions (Shah, Smith, & Vargas-Hernandez, 2003). The instructor made a list of all canonical data structures that would have been useful in the assignment, and awarded one point for each. Students did not receive multiple points for minor variations on each data structure, so this scale does not count absolute quantity, but the quantity of "different enough" design candidates.
R6: Creativity	For each alternative in the following list, the student receives either 1 point or 0.5 points: Skip List (1 point), Binary Search Tree (1 point), Linked Lists with multi-item nodes (1 point), Extra pointers to speed up list traversal in a linked list (0.5 points), Pre-allocation of nodes for a linked list (0.5 points). These are all of the ideas that students came up with that did not come directly from lecture. Significant ideas received 1 point, and less useful ideas 0.5 points. This scale is meant to measure the *novelty* of student solutions, and our approach is very similar to that of Shah et al. (2003). In this case the instructor took a list of all design alternatives submitted by students, and eliminated those that appeared in most or all student submissions. The instructor then assigned point values to the remaining novel solutions based on how different the solution approach was from the non-novel approaches.
R7: Contest rank	Student solutions were ranked based on three speed tests. These three tests were given to students as part of the assignment description. As part of their analysis, they had to decide how heavily to weight these speed criteria, compared to other criteria such as ease of coding. This scale is meant to measure the *quality* of student solutions. Students received an ordinal ranking in each speed test, and then final rankings were based on a standard sum of ordinals. So, for example, a student that received 1^{st} in two tests and 3^{rd} in the last (sum of ordinals is 5) would beat a student that placed 2^{nd} in all three tests (sum of ordinals would be 6).

We have already given the start of an example in the Liminality subsection above. Our problem allows for a very wide variety of valid approaches. The most comfortable approaches would have been to use an array (the main data structure used in previous classes) or a linked list (which they had used on the previous assignment). Students had also seen the approach of leaving spare space at both ends of an array to cut down on the time needed for adding and removing items, and hash tables. None of these approaches were optimal for all operations: Linked lists are slow for searching and indexing, arrays are slow for searching, and a naïve use of hash tables would result in either fast searching or indexing (depending on whether one uses the value or the array index as the hash key), but not both. Also, many students considered their own programming abilities when selecting a design, and so leaned toward array- or linked-list-based solutions because these solutions tend to be easier to read, easier to program, and easier to debug.

The most creative students found synergistic combinations of the canonical approaches.

- One student combined arrays with linked-lists to get a solution that had faster indexing, but which was similar enough to the linked-list he had previously written that he felt confident he could complete it correctly. He changed his linked-list nodes to contain arrays of length 1000, which made his index-based lookups several hundred times faster than students with a regular linked list.

- Several students discovered that they could achieve better performance by keeping two separate data structures, one for searching by value and another for index-based lookups. One student had an array and a hash table, and another had two hash tables. One student attempted to combine a binary search tree (something he learned in high school) with an array.
- Two students kept auxiliary pointers to the middle of their linked-lists to speed up index-based lookup. They (and others) had considered skip-lists as a potential design, but eliminated them as an option for being too complicated.

On what grounds do we call these creative solutions? These students all found ways to combine apparently contrasting approaches. This is a form of liminal thinking, and also suggests that they returned more than once to idea generation. There is plenty of room for creativity in data structures classes, because even relatively small problems tend to fit the canonical data structures poorly.

Excerpts from Student Rationale and Argumentation

We also want to present a few concrete examples of student use of rationale, and reflections on rationale in our Data Structures course. Our goal here is to briefly indicate to the reader the type of argumentation and rationale that students produced. One should not try to make any general conclusions from the three anecdotes presented here, but instead we feel this should provide a bit of clarity in our discussion of assessment below.

Several students explicitly commented on the way that using rationale affected their performance on the assignment. One student from the SEURAT group said,

> *… I kind of went in biased towards a Doubly Linked list with a Hash Table … [but] the Hash-backed Array list still came out on top. This is because, while Doubly Linked list has a faster add/remove time, the Array-list has a much faster lookup by index time. … The design rational helped me visualize. Without this tool, I might not have fully realized that problem until it was too late.*

Interestingly, this student's Rationale (see Figure 2) did not take advantage of SEURAT's hierarchical decision-making capabilities, but did make use of its evaluation affordances.

A student in the Matrix group similarly noted that,

> *The analysis part of this report helped me pick this option. Doing the analysis allowed me to compare different options with each other to see the advantages and disadvantages of each. In the end the requirements that I found most important to deciding which option to go with included having a very fast way to search through the data structure, and having am* [sic] *option that was relatively easy to code.*

Though the student's argumentation is very brief (see Figure 3), it captures key differences between the various options. The instructor is able to see that the student thought through all the criteria, and had reasons (even if incorrect or naïve) behind the choices made.

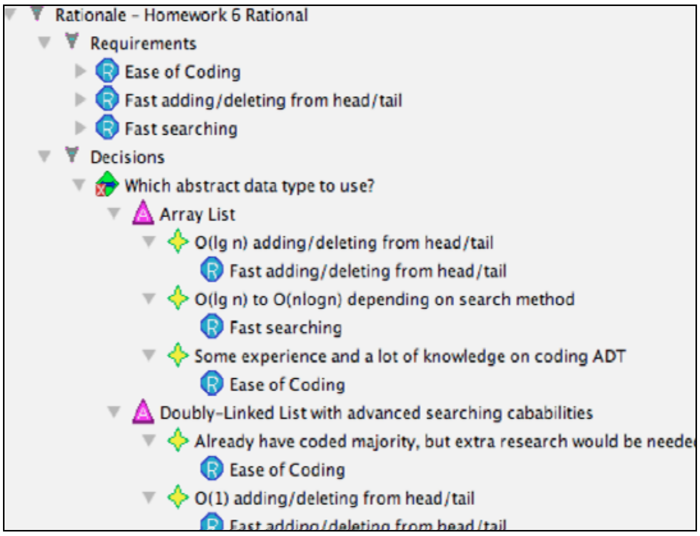

Figure 2. Excerpt from a student's SEURAT rationale.

Of course, there were some students critical of the use of design rationale. One student in particular commented about SEURAT that,

> *My design rational* [sic] *helped me … but in the end I don't think I will agree with it. … I could have figured this process easier by just writing this all out on paper, … I looked at my decisions and realized what was most important to me in this project, learning about the data structures. (I didn't put this in the calculations, so maybe they would be different...).*

What the student is saying here is that he chose not to include his most important decision criteria in his rationale and, as a result, the design rationale did not support his eventual decision. Several students struggled because they believed that only technical criteria should be present in design rationale. For a homework project in an undergraduate course, however, the controlling criteria may be completely nontechnical, just as in industry. Most students in the course seemed to understand this, and were willing to include nontechnical concerns (like time available to code or educational goals) in their list of criteria.

Assessing the Results of the Data Structure Design Assignment

The main purpose of this assignment was to stimulate student creativity and critical reflection through the use of design rationale. It will be quite clear to the reader that this was not an experiment designed to validate our theory, but rather a first attempt to put our ideas into practice.

Requirements	Option 4	Option 5
	Array-backed list with doubling approach	*Array-backed list with double and dictionary ADT*
The program is simple to code	Harder than the linked lists	The hardest option to code
The program is commented well	Easy to comment	Easy to comment
addToHead is fast	addToHead is nLog(n) time	addToHead is nlog(n) time
addToTail is fast	addToTail is nlog(n) time	addToTail is nlog(n) time
removeFromHead is fast	removeFromHead is nlog(n)	removeFromHead is nlog(n)
removeFromTail is fast	removeFromTail is nlog(n)	removeFromTail is nlog(n)
Search is very fast	Search is linear	search is constant
The program is robust	Harder than linked lists	Hardest to make robust
The list scales well	Methods do not scale as well as option 3	Search scales well, others methods don't
Finding nth node	time constant	time constant

Figure 3. Excerpt from a student's Pugh Matrix.

Nevertheless, we provide some assessment of the results of the assignment. We provide some evidence that using a RMS (as opposed to simply written rationale) did not negatively impact student performance. We also tried to gauge the amount of creativity exhibited by students in the project.

Experimental Results: Student Success and Rationale

As described above, we evaluated student designs using seven rubrics. The first four rubrics (R1–R4) assessed student success in assignment tasks, and were initially rated on a scale from 0 to 3 independently by the two authors. In cases of disagreement, the authors consulted and reached a consensus rating. Table 4 shows the results for each of the standard scale metrics.

The students showed success as measured by metrics R1, R3, and R4, but were less successful with mapping criteria to alternatives. Lacking a control group, we cannot prove that using design rationale helped students develop their criteria and alternatives, but we can observe that almost all met instructor expectations on these tasks, indicating that the design rationale were not an impediment to the intended learning. From the weak scores on R2 (mapping criteria to alternatives), it appears that students were weakest in the analysis phase of the assignment, which is not surprising for students at this level.

Experimental Results: Comparison of SEURAT to Matrix Results

The use of Pugh-style matrices is well established in engineering design. We wished to evaluate whether or not using SEURAT in undergraduate classes was a supportable approach. In order to do this, we considered seven hypotheses that compare student performance when

Table 4. Results for Student Success Metrics.

		SEURAT	Matrix	Average of both groups
R1: Are all relevant alternatives identified and provided?	%Excellent/High	**68**	**53**	**61**
	%Good/Medium	32	37	34
	%Poor/Low	0	10	5
R2: Are the criteria appropriately mapped to the alternatives?	%Excellent/High	47	29	38
	%Good/Medium	41	35	38
	%Poor/Low	12	35	24
R3: Do the students select an alternative based on their rationale?	%Excellent/High	**83**	**79**	**81**
	%Good/Medium	6	11	8
	%Poor/Low	11	11	11
R4: Do the students change their decision after the criteria change?	%Excellent/High	**83**	**74**	**78**
	%Good/Medium	6	5	5
	%Poor/Low	11	21	16

Note: N = 38, and some columns sum to 99% or 101% due to rounding.

using SEURAT to student performance when using the Pugh Matrix method. If students using Pugh matrices did substantially better than students using SEURAT, we might conclude that SEURAT should not be used with younger students. However, this turned out not to be the case: Students using SEURAT performed as well as, or better than, students using Pugh matrices in most tasks.

We used a two-sided Mann-Whitney U test to compare experimental groups (see Table 5). Because we have no reason to believe that any of our rubrics would correspond to a normal distribution, we felt that it would be unsound to use, for example, a *t*-test, because it requires the sampled data to be independent and normally distributed. The Mann-Whitney test does not suffer from this defect because it works by first ranking all samples, and then evaluating the likelihood of there being a marked difference in rank sum between the two experimental groups. More information about Mann-Whitney U tests (also known as Wilcoxon rank sum tests) may be found in statistics textbooks, such as Rice (1995, pp. 402-410). We set our threshold for significance at the $\alpha = 0.1$ level.

Table 5. Comparison of SEURAT Users to Matrix Users.

Alternative hypothesis	N_S	N_M	S_S	S_M	Test result
Students using SEURAT are more likely to present all the relevant alternatives than those using the Pugh method. (R1)	19	19	336	405	Null hypothesis accepted, $\alpha \approx 0.32$
Students using SEURAT are more likely to correctly map criteria to alternatives than those using the Pugh method. (R2)	19	19	322	419	Null hypothesis accepted, $\alpha \approx 0.16$
Students using SEURAT are more likely to select an alternative based on their rationale than those using the Pugh method. (R3)	19	19	369.5	371.5	Null hypothesis accepted, $\alpha \approx 0.98$
Students using SEURAT are more likely to change their selected alterative after criteria change than those using the Pugh method. (R4)	18	19	343.5	359.5	Null hypothesis accepted, $\alpha \approx 0.60$
Students using SEURAT will have a more complete set of alternatives than those using the Pugh method. (R5)	19	19	322.5	418.5	Null hypothesis accepted, $\alpha \approx 0.16$
Students using SEURAT will have a more creative set of alternatives than those using the Pugh method. (R6)	19	19	386	355	Null hypothesis accepted. (Note : this shows a negative correlation) $\alpha \approx 0.66$
Students using SEURAT will do better on instructor-defined performance criteria than those using the Pugh method. (R7)	17	18	246.5	383.5	Null hypothesis rejected, $\alpha \approx 0.05$*

Notes: N_s is the number of samples in the SEURAT group, and S_S is their scaled rank sum. N_M is the number in the Matrix group, and S_M is their scaled rank sum. α is a numerical approximation of the probability of rejecting the null hypothesis when it should be accepted, based on a two-sided Mann-Whitney U test. Note that since the lowest rank is best (1st place is better than 38th place), the *smaller* scaled rank sum indicates better performance.
* Statistically significant

We must take some care in interpreting these results. In particular, look at our result for R7. This measures the speed of the student's solution: The instructor gave students some speed-related criteria at the start. Students were free, however, to reject these criteria and instead focus on criteria such as ease of coding, ease of debugging, re-use of code, and other similar criteria that are contrary to high scores in R7.

One major threat to the validity of this assessment is due to the way the experimental groups were assigned; the SEURAT group comprised all students from one section of the course, while the Matrix group composed the other section. The SEURAT group was stronger than the Matrix group as measured by homework grades on assignments other than the experimental assignment. It is possible that higher ability levels of the SEURAT group masked difficulties with using SEURAT that would have been identified if experimental groups were allocated in a more careful way. Furthermore, some have suggested (e.g., Amabile, 1983) that technical expertise is a key factor that enables creativity. This would mean that the SEURAT group might be expected to be more creative than the Matrix group, on these sorts of tasks, simply due to their increased technical proficiency.

To try to correct for this problem we computed a best-fit line between each experimental variable (R1-R7) and final homework grade, and then analyzed the residual values that result when subtracting the predicted values from the actual values. The residual values essentially tell us how much the student was over- or under-performing on this assignment compared to his or her usual performance in the class. When comparing the groups using these residuals,

- The groups were still not significantly different for rubrics R1, R2 and R4,
- The differences in R5 and R7 were no longer significant, and
- The Matrix group outperformed the SEURAT group in R3 and R6, with levels of $\alpha \approx 0.07$ in both cases

Again, we are not making any strong claims about the validity of this approach, but we felt that in the interest of completeness, as well as fairness to the Pugh method, we should present a grade-corrected version of the results.

Experimental Result: Rationale and Creativity

One other interesting trend is the very strong negative correlation, -0.4906, between R1 (Consider all relevant alternatives) and R6 (Consider a more creative set of alternatives). While we cannot make any claims about statistical significance, this relationship seems an intriguing topic to explore in future research.

We hypothesize that this results from some preconceptions among the study participants about the number of alternatives that the instructor expected them to generate. In particular, the instructor indicated that each student should have "at least 4-5 alternatives."

Some commentators have suggested that rationale might be inherently contrary to creativity (see Carroll, 2009). We do not view our results as supporting that claim, because we believe the key problem was the process by which students decided whether or not they had considered "a sufficient variety of alternatives." Students may have been rushing to escape the liminal/undecided state, and so simply stopped when they had four alternatives. We believe that this is not a problem inherent in rationale, but instead a mistake in the instructor's design of the grading rubric for the assignment.

Summary of Claims

Because of the nature of the experiment, we want to be very careful to precisely state what we think our experiment shows. First, our experiment gives some evidence that using a RMS instead of a more traditional (writing) assignment does not negatively impact student learning. Though it now seems obvious that this would be the case, the course instructor initially had serious reservations about using SEURAT in class.

Second, the somewhat weaker scores in evaluating alternatives seem to support our claim that students in the course have not reached the "commitment" stages (levels 7 to 9) of Perry's (1970) scheme. On the other hand, we would be able to obtain much better data on this topic by giving the subjects appropriate critical thinking inventories, and we should do this in the future.

Third, our results suggest future experiments on rationale and creativity in education must be more careful about the instructions given to students about how to assess their own idea generation process. It appears that the use of rationale did indeed inhibit creativity, but probably primarily as a result of a poorly designed grading rubric, which focused on quantity instead of variety. If our theory, that increasing time spent in a liminal state increases creativity, is correct, then it might make more sense to require students to spend a predetermined amount of time on idea generation, instead of aiming for a predetermined number of ideas.

REFLECTIONS ON VERIFYING THE CONTRIBUTION OF RATIONALE TO CREATIVITY IN THE CLASSROOM

In this paper we proposed a theory that use of design rationale with 2nd-year computer science students should lead to improved creativity as well as cognitive development. We reported on an experimental assignment that put these ideas into practice in a Data Structures course, and we reported on our assessment of the impact of this intervention on student learning. It is clear, however, that there are many lessons to learn from the first attempt about *how* one ought to try to verify the effects of such an assignment.

First, we wished to make claims about student cognitive levels or epistemologies. At the most basic level, we assumed that students in our class would exhibit some dualistic tendencies, and that few would be contextual or committed knowers. We are particularly interested in how students at different levels of cognitive development respond to the challenge of creating design rationale, but there was no way for us to assess this because we did not collect this data. In the future we are considering applying epistemic inventories, such as Baxter Magolda's (1992) measure of epistemological reflection, as well as attempting to develop an inventory that specifically measures the student's epistemology of knowledge as it relates to software design. We hypothesize that students in transition or multiplicity may view relativism as normal in humanities classes, but not in engineering classes.

It is unlikely, however, that such inventories can directly assess the impact of our intervention on students, even if given as both pretest and posttest. Because students take many courses, most of which aim to increase student cognitive level, it seems unlikely that the effect of our intervention can be teased out from such data. In the future we should directly ask the students questions about why they did what they did and how they made decisions on our particular assignment. The examples of student reflection presented in earlier in this paper were suggestive, but we did not systematically ask students to comment on their processes. So we only have such data for a very small number of students.

Similarly, general-purpose measures of ideation similar to those used by Shah et al. (2003) should be adopted to make our results on creativity more comparable with results presented by other researchers. This would still leave us, however, with the problem of discovering *why* the student produced the set of design ideas that he/she produced. This again requires some qualitative methods that we did not use in our initial assessment. We need to question students, either on paper or through interviews, as to why and when they stopped generating ideas. If, for example, they did not start building their design rationale until the hour before it was due, we should not be surprised that the alternatives discussed were canonical or familiar. An approach similar to the verbal protocol analysis used by Atman et al. (2005) would be most useful.

FUTURE WORK—RATIONALE ACROSS THE CURRICULUM

The experiment described in this paper focused on one group of students, those taking the Data Structures course (typically sophomore computer science majors and junior engineering majors). If the goal is to aid student cognitive development, as shown by their progression through the levels of the Perry (1970) scale, appropriate exercises and evaluation measures need to be applied at multiple points throughout the curriculum and, ideally, cognitive development evaluated both as an aggregate over all students and for individual students as they pass through the program.

This would require that the exercises be targeted to specific stages of development. At the earlier stages of their education, students can be presented with the problems, candidate solutions, and the rationale. For example, in a Data Structures course, the students learn many different ways to represent collections of objects. The student focus is often on how to implement these collections. The implementation is certainly important, but since many of these constructs are often supplied with the programming language (and do not require implementation), it is often more important that the students understand why they might choose a particular data structure for a problem, that is, to *analyze* the possible solutions by determining which criteria are relevant to the specific problem. The students' emphasis on the implementation rather than the selection becomes apparent in later classes, where they tend to stick to one or two favorite structures that may or may not be the best choices for the problem at hand. Providing rationale in a form that can be easily understood and manipulated may be a more effective way to teach the students the tradeoffs involved in selecting between data structures. The ability to manipulate the argument criteria can also help the students to explore how changing priorities result in different preferred solutions. Rationale can be presented in a form where the criteria can be manipulated by modifying their relative importance in order to demonstrate how as criteria change, so should the recommended solution.

When the students are comfortable with the idea of multiple alternative solutions, the next step would be to involve them in exercises where the problems and criteria are provided but where the students need to identify (*synthesize*) the candidate alternatives based on what they have learned in class and on their own experience. An example of this would be if students were asked to provide alternative methods for data entry or visualization based on usability criteria that they have learned in an HCI course.

The ability to identify the problems themselves, propose solutions, and define criteria requires *evaluation*—identifying what aspects of the solutions are important to the problem and its context. This is an essential skill in both software requirements analysis and in design. The requirements elicitation process is one of defining the problem and the criteria under which the solution will be evaluated, while the design process involves identifying and selecting solutions to that problem.

The movement from dualism through multiplicity and into relativism and commitment is more of a challenge. Kloss (1994) recommends several strategies to move students from dualism towards relativism that stress the importance of analyzing and structuring different points of view. This requires looking at the alternatives and evidence, including understanding the role of assumptions. As students move from working with the rationale of others to producing rationale themselves, the rationale can serve as both an instructional tool and as a means of assessing their intellectual development.

Table 6 lists the levels of the Bloom Taxonomy (1956), how the reflection and rationale approach should support those levels, and how students at different levels of development, as measured by the Perry (1970) scale, would perform on the rationale-supported tasks.

Three courses in our department curriculum now contain explicit course outcomes regarding analysis of multiple alternatives: CS2 (1st year), Data Structures (2nd year), and Senior Design Project (4th year). Using rationale within these courses will provide an opportunity for studying how rationale can assist in the students' progression from dualism toward the higher levels of development.

SUMMARY AND CONCLUSIONS

Students progress through several stages as they gain knowledge and experience. They run into difficulty when they need to operate at higher cognitive levels than they are accustomed to. The ability to make decisions when confronted with uncertainty and ambiguity is important since the problems they will tackle become more realistic and beyond the point where, if they perform the right sequence of actions, they can produce a single correct answer. The ability to synthesize and evaluate solutions becomes critical for problem solving and creativity. Related to this is the need for students to move beyond duality, where there are always right and wrong answers, towards higher levels of thinking where they can begin to analyze the evidence and understand that not all criteria are equally valid in every context.

Our experiment with using two rationale representations, the SEURAT RMS and Pugh's (1991) total design methodology (as part of a writing assignment), indicated that students at the college sophomore level do indeed start out at a fairly low level. The experiment suggests that using the RMS does not inhibit creativity when compared to results using the more traditional writing assignment. Using rationale, however, did not result in a wider variety of ideas. This could be because the students were told how many ideas were required and some may have stopped searching once they achieved their "quota."

While not giving definite answers on rationale's impact on creativity, there were insights to be gained from the use of rationale. The rationale provided insight into student thinking for

Table 6. Relating Bloom's Taxonomy, Reflection and Rationale, and the Perry Scale.

Bloom	Reflection and Rationale	Perry
Knowledge	Given a general decision problem, list and define the alternative solutions, as described in class. Example: What data structures can be used to store lists of items?	Students at all levels should be able to do this since it could be directly recalled from their lecture notes.
Comprehension	Given a general decision problem, and a set of criteria for making a selection, explain why these criteria are important. Example: Why is it important to be able to efficiently remove elements from a list of items? Given a set of alternatives for a general decision problem, differentiate between them. (This may require giving the students the criteria). Example: What is the difference between two data structures that store lists of items?	Students at all levels should be able to explain the criteria, but students still in the dualism stage may show biases toward certain alternatives (the "right" one) when differentiating between options and may not explore them in detail.
Application	Given a specific decision problem, give a list of possible solutions. Example: Given a design that requires sorting and searching a list of items, which data structures could be used to solve it?	Students in dualism may have difficulty providing more than one solution or more than one valid solution.
Analysis	Given a specific design problem, provide a list of possible solutions and map those to a set of design criteria. Example: Given a design problem that requires sorting and searching a list of items, list the appropriate data structures for storing the items and how they relate to criteria, such as time required to search, time required to add new items, etc.	Students in dualism may have difficulty providing more than one solution or more than one valid solution. If multiple solutions are produced, they may have trouble proposing arguments opposing solutions they have already deemed "correct" or identifying arguments supporting solutions other than the "correct" one.
Synthesis	Given a specific design problem, define the criteria that should be used in order to make a decision. Example: Given a design problem that requires sorting and searching a list of items, what criteria should be used to evaluate candidate data structures? Which criteria are more important to the specific problem?	This is not clear. Will students in dualism only come up with criteria that apply to their chosen alternative, discarding any criteria that do not support their beliefs? Will students in the multiplicity stage have issues identifying some criteria as being more important than others or will they consider all criteria equally valid?
Evaluation	Given a specific design problem, define alternatives and the key criteria, and use this information to select a solution. Example: A design problem that requires sorting and searching a list of items, what are the candidate data structures, what criteria apply in evaluating the appropriateness of each data structure to solving the problem, and given those criteria, which solution is the best choice?	Students in dualism are likely to have the same issues listed above, and are likely to have difficulty getting to the evaluation stage. Students in multiplicity may have difficulty in making a selection, even after identifying alternatives and criteria.

the instructor to use to assess both the student's understanding of the problem at hand and where they are likely to be in their development along the Perry (1970) scale. Understanding where the students are developmentally, relative to where we want them to go, is important in deciding how to help them progress. Rationale can be a valuable tool in both aiding and assessing that progression.

Our experiment demonstrated that rationale provides a mechanism for students to express the results of the analysis, synthesis, and evaluation required to design solutions to problems and provides assistance during the process. Explicitly expressing rationale for their work encouraged both reflection on why they made their choices and the active consideration of multiple alternatives. This experiment demonstrated that students using rationale considered all reasonable alternatives and were able to select criteria and evaluate alternatives in a way that indicated they were progressing in their intellectual development.

REFERENCES

Amabile, T. M. (1983). The social psychology of creativity: A componential conceptualization. *Journal of Personality and Social Psychology*, *45,* 357–377.

Atman, C. J., Cardella, M. E., Turns, J., & Adams, R. S. (2005). Comparing freshmen and senior engineering design processes: An in-depth follow-up study. *Design Studies*, *26*, 325-357.

Baxter Magolda, M. B. (1992). *Knowing and reasoning in college: Gender-related patterns in students' intellectual development*. San Francisco: Jossey-Bass Inc.

Bloom, B. S. (Ed.). (1956). *Taxonomy of educational objectives: The classification of educational goals*. New York: David McKay Company, Inc.

Bruegge, B., & Dutoit, A. H. (2004). *Object-oriented software engineering: Using UML, patterns, and Java* (2nd ed.). Upper Saddle River, NJ, USA: Prentice Hall.

Burge, J. E., & Brown, D. C. (2004). An integrated approach for software design checking using rationale. In J. Gero (Ed.), *Design Computing and Cognition '04* (pp. 557–576). Cambridge, MA, USA: Kluwer Academic Publishers.

Carey, T., McKerlie, D., & Wilson, J. (1996). HCI design rationales as a learning resource. In T. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 373–392). Mahwah, NJ, USA: Lawrence Erlbaum Associates.

Carroll, J. M. (2009, June). *The essential tension of creativity and rationale in software design*. Manifesto from the workshop on Creativity and Rationale in Software Design, State College, PA, USA.

Casaday, G. (1996). Rationale in practice: Templates for capturing and applying design experience. In T. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 351–372). Mahwah, NJ, USA: Lawrence Erlbaum Associates.

Daughtry, J., Burge, J., Carroll, J. M., & Potts, C. (2009). Creativity and rationale in software design. *SIGSOFT Software Engineering Notes, 34*(1), 27–29.

Dewey, J. (1933). *How we think*. New York: D.C. Heath and Company.

Dym, C. L., Agogino, A. M., Eris, O., Frey, D. D., & Liefer, L. J. (2005). Engineering design thinking, teaching, and learning. *Journal of Engineering Education*, *94*, 103–120.

Fischer, G., Lemke, A. C., McCall, R., & Morch, A. I. (1991). Making argumentation serve design. *Human–Computer Interaction*, *6*, 393–419.

Fox, C. (2006). *Introduction to software engineering design*. Boston, MA, USA: Addison Wesley.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns: Elements of reusable object-oriented software*. Reading, MA, USA: Addison-Wesley.

Glass, R. S. (1995). *Software creativity*. Englewood Cliffs, NJ, USA: Prentice Hall.

Hamilton, H. R. (1974). *Screening business development opportunities*. *Business Horizons*, *7*(4), 13–24.

King, P. M., & Kitchener, K. (1994). *Developing reflective judgment*: *Understanding and promoting growth and critical thinking*. San Francisco: Jossey-Bass Publishers.

King, P. M., & Kitchener, K. (2002). The reflective judgment model: Twenty years of research on epistemic cognition. In B. K. Hofer & P. R. Pintrich (Eds.), *Personal epistemology: The psychology of beliefs about knowledge and knowing* (pp. 37–61). Mahway, NJ, USA: Lawrence Erlbaum.

Kloss, R. J. (1994). A nudge is best: Helping students through the Perry scheme of intellectual development. *College Teaching*, *42*, 151–158.

MacLean, A., Young, R. M., Bellotti, V. M., & Moran, T. P. (1991). Questions, options, and criteria: Elements of design space analysis. *Human-Computer Interaction*, *6*, 201–250.

Moran, T. P., & Carroll, J. M. (Eds.). (1996). *Design rationale: Concepts, techniques, and use*. Mahwah, NJ, USA: Lawrence Erlbaum Associates.

Perry, W. G. (1970). *Forms of intellectual and ethical development in the college years: A scheme*. New York: Holt, Rinehart, and Winston.

Pugh, S. (1991). *Total design*. Wokingham, UK: Addison-Wesley Publishing Company.

Rice, J. A. (1995). Mathematical statistics and data analysis (2nd ed.). Belmont, CA, USA: Duxbury Press.

Schön, D. A. (1983). *The reflective practitioner*. New York: Basic Books, Inc.

Shah, J. J., Smith, S. M., & Vargas-Hernandez, N. (2003). Metrics for measuring ideation effectiveness. *Design Studies, 24*, 111–134.

van Aalst, J. W., van der Mast, C. A. P. G., & Carey, T. T. (1995). An interactive multimedia tutorial for user interface design. *Computers and Education*, *25*, 227–233.

VanGundy, A. B. (1981). *Techniques of structured problem solving*. New York: Van Nostrand Reinhold Company.

Whiting, C. S. (1958). *Creative thinking*. New York: Reinhold.

Authors' Note

This work was supported by NSF CAREER Award CCF-0844638 (Burge). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation (NSF).

We would like to thank the students taking the Data Structures course for their willingness to participate in this experiment, and fellow participants in the workshop on Creativity and Rationale in Software Design.

Correspondence should be addressed to:
Janet Burge
Miami University
205V Benton Hall
Oxford, Ohio, USA
burgeje@muohio.edu

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

HUMAN TECHNOLOGY

An Interdisciplinary Journal on Humans in ICT Environments ISSN: 1795-6889

www.humantechnology.jyu.fi Volume 6 (1), May 2010, 129–149

DOES DESIGN RATIONALE ENHANCE CREATIVITY?

Jing Wang
College of Information Sciences and Technology
The Pennsylvania State University, USA

Umer Farooq
Microsoft Corporation
USA

John M. Carroll
Center for Human-Computer Interaction
and College of Information Sciences and Technology
The Pennsylvania State University, USA

Abstract: *Creativity and rationale are often viewed as two contrasting facets in software design. A lack in recognizing the facilitative relationship between creativity and rationale not only underestimates the benefits designers can obtain from rationale practices, but also confines the approaches that support creativity in software design. Our exploratory study provides empirical evidence of the positive correlation between rationale and creativity. Furthermore, we found that the feasibility of design alternatives and the comprehensiveness of tradeoff evaluation are critical to enhancing novelty, persuasiveness, and insightfulness. We also discuss future directions to further understand how these properties, or rationale quality in general, affects design creativity.*

Keywords: *design rationale, creativity, software design, quality, empirical study*.

INTRODUCTION

Creativity is often desirable in design activities. In order to create innovative artifacts, designers and design teams need to generate novel ideas, the originality and usefulness of which should be recognized and appreciated by others. In solving ill-defined complex problems, like software design and development, such creativity can hardly be achieved solely by individuals' one-shot or random thoughts, but requires designers to keep track of what has been done and why that has been done.

Design rationale can help keep track of those activities and reasons. Moreover, it can provoke designers to analyze and evaluate their design critically. These critical thinking processes are crucial to design creativity. However, we do not advocate capturing every detail of the design process in rationale. Indeed, that extreme form of design rationale is often criticized for the tedious work involved.

URN:NBN:fi:jyu-20105241910

Overemphasizing the cost of articulating design rationale will disguise the relationship between rationale and creativity as pure contrast. Instead, by focusing on the benefits of design rationale, the integrative potential between the two may emerge. Therefore, our study aims at bridging the gulf between rationale and creativity in software design by exploring why and how rationale and creativity can be mutually facilitative. Design rationale delineates the assumptions behind the questions in design. It may further stimulate designers to problematize design options and reframe or recreate design. By critically evaluating options, designers may create new possibilities to augment the strength and attenuate the weakness of current options. They may also make more rational decisions by converging on an option. However, such a statement has not been fully verified by theories, or by empirical studies. Our study is an empirical attempt to examine the relationship between rationale and creativity, and, in the long term, to understand how to support creativity by design rationale.

Following the presentation of the conceptual background, we will describe our study's context and design. We then will present the results, followed by an interpretation and discussion of those results and a reflection upon the whole exploratory study. We conclude with a discussion of future work.

CONCEPTUAL BACKGROUND

Design rationale is often characterized from various perspectives (Shipman & McCall, 1997). Instead of recording every detail of design processes, our view of design rationale focuses on reasoning and argumentation. From this perspective, design rationale emphasizes the articulation and representative reasoning underlying design (Fischer, Lemke, McCall, & Morch, 1991; Moran & Carroll, 1996; Buckingham Shum & Hammond, 1994). Argumentation-based design rationale attempts to stimulate designers to think and discuss design within a structured or semistructured representation. For example, in QOC (MacLean, Young, Bellotti, & Moran, 1991), questions are framed to pose key issues in the design, options are proposed as possible alternative solutions to the questions, and criteria are bases for reasoning and evaluating the options so as to choose among them. Argumentation-based design rationale is also illustrated in the task-artifact framework (TAF; Carroll & Rosson, 2003), in which tasks are represented as scenarios of use and claim analysis, enumerates the features of a system being used and their upsides and downsides of consequences. Many other endeavors have been invested to capture design rationale with such purpose: for example, an issue-based information system (IBIS; Kunz & Rittel, 1970) and its variants (gIBIS, Conklin & Yakemovic, 1991; PHI, Fischer et al., 1991), three extensions of IBIS (Potts, 1996), and DRL (Lee & Lai, 1991). By adapting notations from these, we will investigate whether and how design rationale facilitates evaluation abilities and critical thinking, and further affects creativity in software design.

The use of argumentation-based design rationale has been investigated in terms of both its benefits and costs (see the review in Buckingham Shum & Hammond, 1994). Despite the distraction and difficulty in externalizing design rationale into one semistructured form (Fischer et al., 1991; Buckingham Shum, 1993), empirical studies have shown evidence that in the design domain (a) argumentation-based design rationale can facilitate reasoning by augmenting both the product and process (Bellotti, 1993; Burgess-Yakemovic & Conklin, 1990; Buckingham Shum, 1996; Buckingham Shum, MacLean, Forder, & Hammond, 1993), and (b) an existing argumentation-based rationale of earlier design is useful for both the owner and others (Conklin & Yakemovic, 1991; McKerlie & MacLean, 1994; Buckingham Shum, 1993). Studies have indicated that design rationale is particularly beneficial when it is applied to driving construction, to facilitating breadth-first exploration, and to early stages of design when less abstract externalization is required. Other than evaluating design rationale for its usefulness, research on the use of argumentation-based rationale also invested great efforts in the usability of its notation and representation (see the review in Buckingham Shum & Hammond, 1994), which is beyond the scope of this paper. However, of all the empirical work, none has reported any direct assessment of the quality of design rationale itself that we could build on.

Creativity is also conceptualized in various ways. Although it is sometimes regarded as mythical and unable to be explained, or as a revolutionary innovation that rarely happens, we consider creativity to be embedded in everyday activities and their social contexts (Amabile, 1996; Csikszentmihalyi, 1996; Gardner, 1993), which may have more implications for education and engineering practices. Even from this perspective, creativity can be characterized in many different ways, among which three features were tapped for our study of creativity measurement. Novelty is the most agreed aspect of creativity (Mayer, 1999; Sternberg, 2006; Sternberg & Lubart, 1991). It implies originality, such as new ideas. With regard to its social context, creativity requires the ability to persuade others the value of the work (Sternberg & Lubart, 1999) so as to be accepted by the field (Csikszentmihalyi, 1999). In our study, persuasiveness as a metric of creativity is assessed by examining the interim product of software design, that is, the rationale documents, with respect to the argumentation elaborated within them. Though insightful thinking cannot guarantee creative design, it is indispensible during the analytical design processes, ensuring powerful critical thinking. Without such scrutiny, novel ideas may not work eventually. The three dimensions of creativity operationalized in our study—novelty, persuasiveness and insightfulness—are expected to demonstrate the three intellectual abilities required to achieve creativity: synthetic ability to see problems in new ways and to escape the bounds of conventional thinking, analytic ability to recognize which of one's ideas are worth pursuing and which are not, and practical-contextual ability, to know how to persuade others of—or to convince other people of—the value of one's ideas (Sternberg, 1985).

The connection between design rationale and creativity in our view derives from the role of rationale in evoking critical thinking, which is often conceived as indivisible from creative thinking. In particular, evaluation abilities are crucial to creative thinking. Runco (1992) and Houtz, Montgomery, Kirkpatrick, and Feldhusen (1979) examined evaluation abilities in creative thinking, and both studies concluded that evaluative abilities play a significant role in creative functioning. Guilford (1967) also assigned an important role to evaluation ability in his models of creative thinking and problem solving.

However, research has been sparse with regard to the integrative potential of rationale and creativity in software design. Ball, Lambell, Ormerod, Slavin, & Mariani (2001) proposed that design rationale can provide insights into how best to represent and retrieve design knowledge in order to support innovative design reuse. They developed a design reuse system, Desperado II, to elicit and retrieve design rationale. By comparing the performance of the Desperado group and a control group, they found that the Desperado group considered up to three times as many options per question as the control group, and up to six times as many criteria. They claimed such performances as evidence that Desperado encourages innovative design, even though the increased number of options and criteria were not examined in terms of their quality. Even if it is validated that Desperado can assist in overcoming satisficing tendencies and confirmation biases, their results only demonstrate that the representation of design rationale in Desperado can support creative design better than previous notations. It cannot explain the underpinning relationship between design rationale in general and creativity. Fischer (2004) argued that temporal barriers should be overcome to support social creativity. In a long-term design project, creativity can be supported by recording design rationale with minimal efforts and by providing a search utility to retrieve rationale easily. However, their paper did not provide much detail about empirical validation of this claim.

CLASSROOM STUDY

As a preliminary step to explore the relationship between rationale and creativity in software design, we conducted a classroom study that lasted for a whole semester. The design processes involved in such educational setting may not be exactly the same as those of professional designers, for instance, in terms of the process complexities. Nonetheless, the problem-solving practices in our study can still be creative in ways similar to those in professional work settings, especially the roles of reasoning and reflection throughout the entire design process. Moreover, all participants were assigned the same tasks without any other direct manipulation because of the ethical concerns in the educational setting. With particular interest in group collaboration, we asked participants to work on the tasks in groups, while we did not require them to communicate with each other through a particular medium.

Our study context lends itself to directly explore the relationship between rationale and creativity in software design. First, this naturalistic setting allows more flexibility to observe more than one property of design rationale that may influence creativity and to discover multiple ways, if there are any, in which the influences occur. Second, the benefits of articulating rationale on critical and analytical thinking cannot be achieved in a fabricated task that lasts only for a couple of minutes. Participants in our study carried out the task of articulating rationale during their real design practices over the semester.

Research Questions

As speculated in the workshop manifesto[1] (Carroll, 2009), creativity and rationale could have a mutually facilitative relationship. In this empirical study, we focused primarily on one direction of that relationship, that is, the effects of rationale on creativity in software design. More specifically, we wanted to examine whether rationale can play a positive role in enhancing creativity in software design and how. Therefore, we proposed the following research questions to guide our exploration.

Research Question 1: Does Better Quality Design Rationale Lead to More Creative Software Design?

Design rationale can be classified as prospective and retrospective (Carroll, 2009). People have different perspectives and thus different representations and usages of design rationale. From our view, the greatest potential for integrating rationale and creativity is in the activities of constructing and articulating rationale during software design process. Rationale developed in such scenarios can be regarded as prospective because it is generated within the design activities and enables further software development. By creating and capturing rationale themselves, designers can benefit from their own efforts rather than recording the processes for others. Rationale of this kind can facilitate designers framing the problems, evaluating and problematizing alternative solutions, and then approaching the optimal decision. It could also convey the usefulness and value of the design to other people. It may even surface more design options or new design solutions. Therefore, high quality design rationale should enhance designers' reasoning and critical thinking, and thus their creative thinking and creative design. In contrast, retrospective design rationale affects the ongoing design process less and costs more recording efforts.

Research Question 2: What Properties of Design Rationale are Critical to Enhancing Creativity of Software Design?

Other than demonstrating the facilitative effect of rationale on creativity, the manner in which design rationale can enhance creativity is even more important to design tools that support creative design, improve education in software development, and manage innovation of system development in organizations. Since design rationale consists of multiple elements involving a variety of quality characteristics, it is plausible to explore what properties of design rationale, with respect to quality, are critical to creative design.

Not every property of design rationale has positive influences on creativity. As the conventional view on the relationship between rationale and creativity implies, there is inevitable tension between these two concepts. Functioning as documentation, rationale may confine creativity by restraining divergence or adventure. Therefore, some properties of rationale may be valued in terms of rationale quality but not promising for fostering creativity.

Additionally, analyzing creativity from process perspective is compelling given our special interest in group collaboration. This research question focuses on the creative abilities of persons involved and creative characteristics of design products. However, design rationale, as a means to communicate and facilitate discourse, may impact the creative design processes of the group. Therefore, we proposed the following research questions with respect to groups' collaborative processes.

Research Question 3: What Properties of Design Rationale Will Enhance Creative Design Processes of the Group?

Other than mutual facilitative correlation, it is also important to understand the causal effect from design rationale to creativity, meaning, why and how design rationale can evoke and foster creativity. Thus investigating creative processes may shed light on an underpinning relationship between the two. Group creativity requires divergent thinking, convergent thinking, shared information and objectives, and reflexivity (Farooq, Carroll, & Ganoe, 2005). The first impression of the integrative relationship between rationale and creative processes is that the evaluative thinking evoked by rationale might facilitate reflexivity and convergent thinking of the group. Claims (Carroll & Rosson, 2003; Lee & Lai, 1991), criteria (MacLean, Young, & Moran, 1989), or arguments (Kunz & Rittel, 1970) motivate designers to examine design options rationally so as to approach the optimal decision. Furthermore, they provide opportunities to amplify strengths and ameliorate weaknesses of the final solution. Prospective rationale can assist groups with planning the design implementation, while retrospective rationale can offer traceable records for designers to ruminate over previous decision making processes or other people's reasoning trajectory so that group members can further refine an old design or create a new one. However, such impression might underestimate the potential of rationale influencing creative processes. By questioning design options with persuasive claims, rationale might inspire more alternative solutions to overcome the downsides of current ones. It may also evoke new ideas by improving designers' understanding of the total problem, or by bringing more opportunities for designers to encounter unexpectedness. Therefore, investigating how rationale affects creative processes other than the final product is needed.

Research Question 4: How Will Sharing Design Rationale Across Groups Influence a Group's Creative Design Processes?

Design rationale is not just used by its creators for the current design practices, but also visited after the design cycle by both its creators and other people outside the design team. Although people are motivated by different purposes to comprehend rationale documents (e.g., reusing and adapting the design, creating new design, or even are not motivated), design rationale makes it possible to communicate with the software designers about what was going on and why. By collaboratively making sense of rationale, designers may reach a shared understanding effectively and acquire insights into their own design problem and possible solutions. Moreover, open information exchange across organizational boundaries is an important determinant of creativity (Henry, 2004; Woodman, Sawyer, & Griffin, 1993). Design rationale, as one type of information shared between groups, may also augment groups' information sources, introduce flexibility, and open new opportunities. Despite more effort required to take advantage of such rationale, groups will obtain better chances for learning and cooperation. All of the benefits from intergroup sharing of rationale are possible but need to be examined.

Participants

Participants were undergraduate students majored in computer and information sciences in an advanced object oriented design and software course. The advanced course required prerequisite courses, including one introducing general computer languages and another on a specific computer language (either Visual Basic or C++). Thus all of the participants had basic knowledge and experience to some extent in software design before our study. The course had two sections taught by different instructors. Participants came from both sections and shared the same syllabus and class activities. The 49 students participated in our study by voluntary consent. 3 of them are female and the other 46 are male.

Task Description

The task required participants to solve specific design problems by capturing design rationale and implementing the design by Java. It was integrated into every lab programming assignments of the course over the semester. Each lab assignment specified the goal of design that students should achieve. For example, one assignment asked students to implement a graphical user interface to support a decision model given by the instructor. Students had more than a week to work on each assignment. In the middle of each assignment, they were asked to submit their design rationale as their progress reports, which were counted as part of their grades. After students had turned in their implementation towards the end of each lab, the instructors showed their own solution.

The design problems of each lab assignments allow students to act creatively. Although these assignments are generally close-ended, they do not confine the students' approach to the problems along a single definite path. Therefore, participants had the opportunity and enough flexibility to perform creatively. Furthermore, all of the lab assignments were related to each other, contributing as a component to a complete project. Specifically, the later tasks were supposed to be built upon the earlier ones. This may facilitate students' reuse of prior designs.

We set the submission time of design rationale a few days before that of implementation. We expected this time arrangement could enable design rationale to facilitate students in thinking critically. Based on our hypothesis about the relationship between design rationale and creativity, such prospective design rationale created during design processes should drive the construction as well as augment the reasoning and reflection of its owners. When it is used by other people, it may also have similar benefits for them.

Design rationale documents were specified in a uniform format for all the students of the course. To minimize the overhead of recording rationale, we simplified its representation into four components, including (a) the toughest design problems you are facing or did face, (b) the design alternatives for solving these problems, (c) the pros and cons for each alternative, and (d) what option you are leaning towards (the decision on alternatives). These elements are consistent with those of other methods, such as IBIS (issues, positions and arguments; Kunz & Rittel, 1970) and QOC (questions, options, and criteria; MacLean et al., 1991). Since it was not a topic in the curriculum, the instructors explained what benefits design rationale could bring to software design, such as keeping track of the design state. They further illustrated design rationale by a sample. The sample presented design rationale for a concrete design problem in the format that students were required to use. Because this representation is not highly abstract but rather more narrative, the students did not find difficulty in articulating rationale in this way. Students were already familiar with externalizing design rationale in the way we defined since we did not start our analysis until their third lab assignment.

From the third lab assignment, students also began cooperating on their lab assignments in pairs or triads. Each team had to submit only one design rationale and one implementation for each assignment. No particular medium was specified for their communication and collaboration. They chose teammates by themselves but could not pick the same person more than once. They were also given a short time in class to collaborate with their team member(s) on these assignments. In other words, their initial collaboration was face-to-face, although they also might have communicated virtually sometimes. Such rotated pairing may better motivate students to share design rationale across groups because each student in the group can serve as a boundary object (Fleischmann, 2006) between his or her prior and current design. Intergroup information exchange is likely to happen when there are people who have sufficient knowledge of practices of both groups (Henry, 2004).

Data Collection

Data collected in our study comprised two parts: (a) an assessment of design rationale quality and design creativity; and (b) the responses to surveys with respect to design collaboration. The first part was mainly gathered to answer research questions 1 and 2 (i.e., the relationship between rationale quality and design creativity). The second part was used to address research questions 3 and 4 (i.e., the effects of rationale quality on perceived creative processes). Data were sampled from three out of all the lab assignments of the course: one was at the beginning of the semester when they just started to work in teams (i.e., the third lab assignment); one was in the middle of the semester, and the other was at the end. Surveys were disseminated immediately after rationale submission for each of the three lab assignments. The number of respondents varied across the three assignments, based on their own voluntary decision. The assessment on design rationale quality and creativity was conducted with the criteria we developed (see Appendix for details) by two teaching assistants.

To investigate the relationship between rationale quality and design creativity, we developed criteria for evaluating design rationale documents in terms of these two focuses. Design rationale and creativity are each comprehensive concepts with various perspectives. Because, as noted in the previous section, no established evaluation scales can be directly applied in examining our specific data, the three authors brainstormed and decomposed rationale quality and creativity into measurable dimensions, respectively.

Judging rationale quality (6 dimensions) encompassed concerns for each element in the design rationale representation we developed, that is, problems, alternatives, tradeoffs, and decision. The overall quality of the rationale as a communication tool was also operationalized as *clarity of articulation* criteria. Problem identification and definition is critical to all the effort involved in problem-solving activities, determining the level of critical thinking (Garrison, 1992; Henri, 1991). Given the problem-solving nature of software design, identifying and defining the design problem are also important in articulating design rationale. These require critical thinking. Moreover, problem-solving influences the quality of other elements in design rationale as well, such as how well the proposed alternatives address the design problem, and whether the decision made is wise. Thus, in our rationale quality evaluation, we asked for judgment on whether the statement of design problem captured a critical issue of the design task (i.e., *toughest design problem identification*). Alternatives are the possible solutions designers generate to tackle the problem. Good alternatives should be able to the design issue as a solution candidate (i.e., *relevance of alternatives* in our evaluation criteria; see Appendix). Furthermore, they should provide not only conceptual guidance but also feasible ideas, since software design is expected to lead to practical results. Therefore, we employed *feasibility of alternatives* in our assessment. Specifically, we customized the definition of feasibility by students' programming ability acquired from the course. Tradeoffs are articulated when designers justify or problematize their alternatives. Maintaining high quality of tradeoffs requires exploring a wide range of possible consequences of a design alternative. Analysis from a single perspective may bias designers' judgment of an alternative. *Comprehensiveness of tradeoffs* in our criteria aimed at addressing this requirement of rationale quality. One outcome of critical thinking involved in design rationale articulation is the decision, namely the final problem solution to be implemented. The quality of the *decision* indicates the performance of analytical work engaged in design rationale documents. Thus, we included this dimension by asking whether the design alternative selected is the optimal solution in our evaluation criteria. As an artifact for communication purposes, clarity also represents the quality of design rationale. Moreover, clarity implies how thoroughly designers have considered the design problem, alternatives, tradeoffs, and the decision. These metrics were not exhaustive but rather reflected an argumentation perspective that involves critical thinking. Relevance, ambiguities (clarity in our study), practical utility (feasibility in our study), and width of understanding (comprehensiveness in our study) are also identified as critical factors in the model of assessing critical thinking, which was developed by Newman, Johnson, Webb, & Cochrane (1997).

The rubrics for evaluating creativity (3 dimensions) in the identified problems solutions were adapted from Farooq's (2008) doctoral dissertation. Novelty as a common feature of creative ideas was maintained in the adaptation (i.e., *novelty of alternatives*). *Insightfulness of tradeoffs* was added to our rating dimensions because design rationale has the characteristic of analytical thinking, which was not externalized in the task of Farooq's study. We evaluated creativity by rating rationale documents instead of directly assessing students' code. The underlying assumption for this decision was that the students would solve the problem in the way they stated in their design rationale documents. Furthermore, the overall criteria do not evaluate creativity only from the final product perspective, but also indicate the creative capacity of designers that may not be explicitly codified in their product. For instance, *persuasiveness of tradeoffs* represents the designer's ability to persuade others of the value of his/her design, which is suggested to be an important dimension of creativity (Sternberg & Lubart, 1991).The judges who rated the design rationale quality and design creativity were the teaching assistants for the two sections of the course. They were considered to be qualified in several aspects: (a) both had advanced object oriented design experience; (b) they were very familiar with the tasks; (c) they knew the students' design expertise well; and (d) they had the closest interaction with participants, which may have assisted in their understanding and judgment of the students' design processes.

Both judges followed the same procedure to assess the submissions. Prior to implementing the study, we made sure the criteria were appropriate for the context of the course and study and executable for the judges to apply. We walked through the evaluation criteria with the judges, during which they interpreted the rubrics item by item to confirm that their comprehension was consistent with our intention. Then they independently evaluated every document based on their overall impression of the performance of the whole class, with rating scale from 1 (*very poor*) to 5 (*very good*). To prevent the order effect, the judges rated the first half of the students' rationale documents in the order of rationale quality to creativity, and the other half of documents from creativity to rationale quality. In the analysis, we averaged the scores rated by the two judges as the final assessment output.

Surveys were designed to complement our understanding of the collaborative design processes. They consisted of questions with respect to students' perceived creative processes by articulating rationale as well as their use and reflection on previous rationale. These questions were rated on a 9-point Likert scale. We also collected data about some demographic information and personal creative characteristics in the last survey at the end of the semester.

RESULTS

For the three lab assignments we collected, 27, 28, and 17 design rationale documents submitted by teams, respectively, were rated by judges, while 16, 16, and 14 responses to our surveys by individuals, respectively, were collected. Since participants completed the surveys voluntarily, the number of responses was smaller than the number of design rationale documents that were consented to be analyzed. We conducted regression analysis on both data sets: one set included all 72 average ratings for the progress reports submitted; the other set included all 46 survey responses and the survey participants' progress reports (excluding the assessment scores for the students who did not fill out the surveys). In the analysis, properties of rationale quality were used as independent variables, and properties of creativity and ratings of perceived creative processes were set as dependent variables.

RQ1: Does Better Quality of Design Rationale Lead to More Creative Software Design?

All of the properties of rationale quality significantly positively correlated with the three aspects of creativity as indicated by Pearson correlation analysis. Furthermore, by using multiple regression analysis with stepwise approach, we found that the higher the quality of design rationale, the more creative the design, although the regression models suggested by the two data sets were slightly different. To be specific, for both data sets we performed stepwise regression on the three aspects of creativity the judges rated (i.e., novelty of alternatives, persuasiveness of tradeoffs, and insightfulness of tradeoffs) to select the properties of design rationale that affected these aspects individually. We also coded the lab number as independent variables in regression models to examine whether there was any confounding effect caused by the difference of lab assignments.

Given two judges performed assessment, interrater reliability was also tested on each criterion, as well as the entire rubrics. Comprehensiveness, persuasiveness, and insightfulness of tradeoffs had values of interrater reliability larger than the rule of thumb 0.7 (Cronbach's Alpha = Average Measures Intraclass Correlation Coefficient = 0.839, 0.736, and 0.794, respectively). Additionally, the overall creativity (novelty of alternatives, persuasiveness of tradeoffs, and insightfulness of tradeoffs) had fair interrater reliability (Cronbach's Alpha = 0.686, Average Measures Intraclass Correlation Coefficient = 0.674). The final three measurements did not achieve interrater reliability as high as these, and therefore could be decomposed or clarified more in the future to approach more agreement and consistency.

RQ2&3: What Properties of Design Rationale are Critical to Enhance Creativity of Software Design and How?

Regression analysis on the first data set indicates that two properties of rationale quality, feasibility of alternatives and comprehensiveness of tradeoffs, are crucial to enhancing design creativity. The first data set does not include any variables measured by survey responses but all the ratings on 72 design rationale documents. We built three models for each aspect of creativity with stepwise regression. Table 1 summarizes all these models for predicting the three aspects of creativity on the first data set. The model predicting novelty of alternatives is Novelty = 0.612 * Feasibility +0.238 * Comprehensiveness, $F(2, 69) = 97.88$, $p < .001$, $R^2 = .729$. The model for predicting persuasiveness of tradeoffs is Persuasiveness = 0.361 * Feasibility + 0.620 * Comprehensiveness, $F(2, 69) = 202.044$, $p < .001$, $R^2 = .854$. The model for predicting insightfulness of tradeoffs is Insightfulness = 0.254 * Feasibility +0.730 * Comprehensiveness,

Table 1. Relationship Between Rationale Quality and Creativity on the First Data Set (N =72).

Model	Quality of DR (IV 1)	Coefficient 1	Quality of DR (IV 2)	Coefficient 2	Creativity (DV)
1	Feasibility of Alternatives	0.612***	Comprehensiveness of Tradeoffs	0.238** (p = .001)	Novelty of Alternatives
2	Feasibility of Alternatives	0.361***	Comprehensiveness of Tradeoffs	0.620***	Persuasivene ss of Tradeoffs
3	Feasibility of Alternatives	0.254* (p = .013)	Comprehensiveness of Tradeoffs	0.730***	Insightfulness of Tradeoffs

Note: Significance level: *** $p < .001$, ** $p < .01$, * $p < .05$

$F(2, 69) = 166.254$, $p < .001$, $R^2 = .828$. Although the determination of a reliable relationship in this analysis does not imply causality, it will increase our understanding of what properties of design rationale are important to foster creativity.

Moreover, the feasibility of alternatives has a stronger positive relationship with novelty of alternatives ($\beta = 0.612$) than comprehensiveness of tradeoffs ($\beta = 0.238$) according to the values of coefficients in each model above. Conversely, with persuasiveness of tradeoffs and insightfulness of tradeoffs, comprehensiveness of tradeoffs has a stronger relationship than feasibility of alternatives. One reason for such variation can be attributed to the similarity of elements, which means feasibility and novelty are evaluated upon alternatives whereas comprehensiveness, persuasiveness, and insightfulness are evaluated upon tradeoffs.

The Variance Inflation Factor (VIF) values for these multiple regression models are 2.300, which is smaller than 5, indicating that these models do not have multicollinearity problems (O'Brien, 2007). That is, individual predictors in the regression model, meaning feasibility of design alternatives and comprehensiveness of tradeoffs, are not redundant or highly correlated. Their coefficient values provide somewhat precise estimate of their effects on the three aspects of creativity.

Furthermore, we tested whether there is any moderation effect or interaction effect of feasibility of design alternatives and comprehensiveness of tradeoffs by centering these two variables. No significant effect was detected.

Regression analysis on the second data set also suggests that feasibility of design alternatives and comprehensiveness of tradeoffs are two critical properties of design rationale to enhance creativity in software design. The second data set does not include rating scores on design rationale of students who did not participate in the surveys. Independent variables that entered into the three regression models were slightly different from those in the models shown in Table 1. Table 2 summarizes the models for predicting the three aspects of creativity on the second data set. The model predicting novelty of alternatives is Novelty = 0.554 * Feasibility +0.228 * Comprehensiveness, $F(2, 43) = 94.394$, $p < .001$, $R^2 = .832$. The model for predicting persuasiveness of tradeoffs is Persuasiveness = 0.295 * Feasibility +0.664 * Comprehensiveness, $F(2, 38) = 126.41$, $p < .001$, $R^2 = .869$. The model for predicting insightfulness of tradeoffs is Insightfulness = 0.297 * Decision +0.746 * Comprehensiveness, $F(2, 38) = 122.94$, $p < .001$, $R^2 = .866$. The main difference between results generated from the two data sets is that the independent

Table 2. Relationship Between Rationale Quality and Creativity on the Second Data Set (N = 46).

Model	Quality of DR (IV 1)	Coefficient 1	Quality of DR (IV 2)	Coefficient 2	Creativity (DV)
1	Feasibility of Alternatives	0.554***	Comprehensiveness of Tradeoffs	0.228** (p = .002)	Novelty of Alternatives
2	Feasibility of Alternatives	0.295* (p = .013)	Comprehensiveness of Tradeoffs	0.664***	Persuasiveness of Tradeoffs
3	Decision	0.297* (p = .044)	Comprehensiveness of Tradeoffs	0.746***	Insightfulness of Tradeoffs

Note: Significance level: *** $p < .001$, ** $p < .01$, * $p < .05$

variable feasibility of alternatives is replaced by decision optimization in the model predicting insightfulness of tradeoffs. However, the significance of the estimated effect from decision is only 0.044, which is not significant enough as compared to more conservative alpha value rather than the default 0.5.

Similarly, neither a multicollinearity problem nor interaction effect has been discovered. The values of VIF for each model are 2.588, 2.588 and 2.902, respectively.

The common structure revealed by both data sets was the positive correlation between comprehensiveness of tradeoffs and creativity. Other than analytical and critical thinking ability, comprehensiveness can be accomplished from knowledge and expertise in related areas. To analyze whether their prior knowledge in software design affected the comprehensiveness of their articulation of tradeoffs, we collected 14 participants' background information in our last survey. According to the results of a nonparametric test, no significant difference was found between students who had prior experience in object-oriented design before the advanced course (mean of comprehensiveness = 1.71) and those who did not (mean of comprehensiveness = 1.92), nor between students who had built software in their spare time during the previous year (mean of comprehensiveness = 1.30) and those who had not (mean of comprehensiveness = 2.13).

With respect to the third research question, we conducted regression analysis on variables of rationale quality, creativity, and perceived creative processes. We did not find any significant mediation effect of the perceived creative processes upon the quality of design rationale and the creativity in design, nor did we find any significant relationship between the quality of rationale and creative processes, or between creative processes and creativity.

RQ4: How Will Sharing Design Rationale Across Groups Influence Group's Creative Design Processes?

To investigate the fourth research question about the impacts of sharing rationale across groups on creative processes, we collected participants' ratings on perceived creative processes by revisiting their prior design rationale through these items in our second and third surveys (The first survey was excluded because students had not yet started working in teams at that time):

1. The pros and cons articulated for our prior labs or projects evoked more design alternatives of my team.

2. The pros and cons articulated for our prior labs or projects helped my current team members and me pick the best design solution to our current lab.

The first question concerns divergent thinking in creative processes, while the second one is about convergent thinking. Both of them were rated on 9-point Likert scale. However, since sharing design rationale across groups was neither an imperative for all students nor controlled for in different groups, we also asked students in the surveys whether they shared their previous design rationale with their current team members.

Twenty-six responses to these questions in total were collected: half of them (n = 13) did share their prior design rationale with the current group members, while the other half (n = 13) did not. Table 3 presents the descriptive statistics of participants' ratings on their current groups' divergent thinking process and convergent thinking process. According to the mean values of ratings (in the column "Mean"), whether or not they shared their previous design rationale across groups, participants on average felt that they benefited from the rationale of their previous design activities (mean of perceived creative processes > 5). Tables 4 and 5 show crosstabs for each survey question. They also suggest that most of participants thought sharing design rationale across groups was helpful.

We further examined whether sharing prior rationale affected the creative processes of the group. Results indicate that it affected and only affected groups' convergent thinking and decision making. Because neither set of 13 cases had normal distribution, we compared the perceived creative processes between groups who shared their rationale across groups and those who did not share across groups by conducting nonparametric *t*-test. According to the results of Mann-Whitney testing, sharing previous design rationale did not have significant effect on a group's divergent thinking (Asymp. Sig. = .202) but did have significant effect on a group's convergent thinking and decision making (Asymp. Sig. = .039). These results suggest that speculating and communicating with group members on rationale of a related design might assist a group's convergent thinking and decision making. Besides effect on the groups' perceived creative processes, we examined whether such intergroup sharing also affected the quality of rationale and the creativity of the new design. However, no significant difference was found between participants who shared rationale across groups and those who did not. This may indicate that introducing other groups' design rationale would boost their confidence in their consensus but did not make a big difference in real performance. These results will be explained and discussed more in the next section.

Table 3. Comparison of Perceived Creative Processes Between Groups with Intergroup Sharing of Rationale (n = 13) and Groups Without Intergroup Sharing of Rationale (n = 13).

Perceived Creative Processes	Intergroup Sharing of Rationale	*n*	Mean	Std. Deviation
The pros and cons articulated for our prior labs or projects evoked more design alternatives of my team.	0 (not share)	13	5.31	1.49
	1 (shared)	13	6.08	1.04
The pros and cons articulated for our prior labs or projects helped my current team members and me pick the best design solution to our current lab.	0 (not share)	13	5.31	1.55
	1 (shared)	13	6.46	1.13

Table 4. Cross Tabulation of Intergroup Sharing of Rationale*Perceived Convergent Thinking.

Count		The pros and cons articulated for our prior labs or projects helped my current team members and me pick the best design solution to our current lab.					Total
		Rating=1	Rating=5	Rating=6	Rating=7	Rating=8	
Intergroup Sharing of Rationale	0 (not share)	1	7	2	3	0	13
	1 (shared)	0	3	4	3	3	13
Total		1	10	6	6	3	26

Table 5. Cross Tabulation of Intergroup Sharing of Rationale*Perceived Divergent Thinking.

Count		The pros and cons articulated for our prior labs or projects evoked more design alternatives of my team.					Total
		Rating=1	Rating=5	Rating=6	Rating=7	Rating=8	
Intergroup Sharing of Rationale	0 (not share)	1	6	4	2	0	13
	1 (shared)	0	5	3	4	1	13
Total		1	11	7	6	1	26

DISCUSSION

Our hypothesis that rationale and creativity in software design are mutually facilitative and potentially integrable is supported by our classroom study results. Moreover, the feasibility of design alternatives and comprehensiveness of tradeoffs are found to be the most critical properties of rationale quality that are positively correlated with novelty of design alternatives, persuasiveness, and insightfulness of tradeoffs. These two properties of design rationale quality involve critical thinking and evaluation ability in different ways. Despite its limitations, our study opens up opportunities to further investigate how to take advantage of design rationale to enhance the effectiveness and creativity of software design.

Implications

Quality of Design Rationale Facilitates Design Creativity

Our study indicates the positive correlation between rationale and creativity in software design. Although we cannot assert a causal relationship between rationale quality and creativity through our regression analysis, all of the aspects of rationale quality we measured are positive predictors for design creativity. Thus it is plausible to foster design creativity by enhancing the quality of design rationale. The judges might tend to assign similar scores to rationale quality and creativity of each document based on their overall impression of the document. This consistency of individual's judgment can be mitigated by introducing more judges and asking each of them to either assess rationale quality or creativity.

Feasibility and Comprehensiveness of Rationale Enhances Design Creativity

Given the confirmation on their integrative potential, characteristics of rationale quality were examined to help us contemplate on why design rationale can promote creativity and how we can support creativity in software design by design rationale. Although analysis on the two data sets with different sizes ideally would have shown the same pattern, the results still indicate two critical properties of design rationale that facilitate design creativity: feasibility of design alternatives and comprehensiveness of arguments. By comparing the properties of rationale quality that entered into our final regression models with those that did not, it is not hard to discern that the ones with weak predictability to creativity (i.e., problem identification, relevance of design alternatives and clarity of articulation) are low-level requirements for designers' capacity.

Feasibility of design alternatives may manifest a higher level of designers' capability and the internal evaluation of designers and their groups, which involves their critically selecting the ideas that can be externalized and recorded in their design rationale. Constraints over design space are not always a negative within the creative process; rather constraints are continually applied in good design (Singley & Carroll, 1996). They pose finites to the space, directing design turned into product. Additionally, creativity is not just about wild thinking; it requires action and implementation (West, 2003). One can hardly operationalize alternatives far beyond one's design knowledge. In this sense, it may also be reasonable to attribute feasibility as one aspect of creativity. Furthermore, creativity, especially divergent thinking, is often mistakenly simplified to represent the number of ideas generated. However, creativity is not only about quantity, but more about quality (Farooq, 2008). Emphasizing the feasibility of design alternatives may filter out some spontaneous thoughts, but it can ensure the design is doable. Aligned with the same concern, the grading rubrics provided to students did not require any specific number of design alternatives so that students would not be motivated to generate some invaluable options. Moreover, in general, the assessing feasibility may be biased by the judges' expertise due to the possible gap between judges' and designers' design knowledge, particularly when the design proposals may be executable for designers but beyond judges' skills. However, in our case the judges are teaching assistants who have privileged experience in the course content and design skills. Thus judgment on feasibility in our study should be considered fair and reliable.

Another aspect of rationale quality—comprehensiveness of tradeoffs—consistently contributes as a significant predicator in the six regression models shown in Tables 1 and 2. Comprehensiveness and correctness decide whether critical reasoning in rationale has positive or negative effect on design (Singley & Carroll, 1996). With comprehensive evaluation, designers will not be confined by the downsides of design options but may be able to create new options that can augment the upsides and mitigate the downsides. Comprehensiveness is neither complexity nor detailing every relevant issue. Nevertheless, it is necessary to capture and enumerate each critical issue in design alternatives in order to achieve comprehensiveness. Considering our study context, effects from comprehensiveness on creativity may also likely come from participants' efforts in their work required by achieving comprehensiveness. Even if they are capable of envisioning all of the critical upsides and downsides, designers are normally not motivated to think thoroughly the entire evaluation space and record all considerations. In general, people generally do not make sophisticated analyses to make rational decisions. They would rather just pick one solution candidate that works.

Comprehensiveness also requires adequate knowledge to justify design options. With limited knowledge or expertise, designers may foresee only part of possible consequences, or they may exert all of their efforts on trivial problems but lose sight of the whole picture. The facilitative relationship between comprehensiveness of tradeoffs and creativity may motivate designers to take a more positive attitude toward constructing their design rationale, rather than negatively consider it as overhead, like any other documentation. Furthermore, tradeoffs encompass both pros and cons. Comprehensiveness does not specify a certain portion or weight for each part; even tradeoffs without many cons can be comprehensive. Thus it may provide us more insights to further examine how comprehensiveness of argumentation influences creativity by decomposing argumentation into pros and cons.

Intergroup Rationale Sharing Assists Group's Convergent Thinking

Our analysis on intergroup sharing of design rationale indicates that sharing prior design rationale other than reusing it across groups may facilitate convergent thinking. There are always motivational obstacles that inhibit information flow across organizational boundaries and difficulty in making sense of unfamiliar contexts. Designers may be even more reluctant to revisit their previous or other designers' rationale documents than to create their own for current practices due to the cost of making sense of those documentations. Groups are often not quite motivated to share information with or incorporate information from other groups unless they have specific needs. In our study, the sharing rationale across groups may be less inhibited by those factors. Students maintain a consistent context because they all know the tasks of each lab. Prior design rationale is reusable because posterior lab assignments are built on the design of anterior ones. Moreover, the rotation of group members ensures that each member in a group has adequate knowledge about the rationale created by his or her prior groups. One incentive to revisit previous design is that students were aware of something wrong in their prior design. By explaining their previous design rationale to their current group members, students might have developed shared understanding and common ground, which assisted with their decision making. However, they did not perceive much difference in the process of coming up more alternatives, whether or not they shared their prior rationale. This may have resulted from the superiority of the instructors' solutions to prior lab assignments. In other words, students may only have applied their previous design rationale to prune poor design options to protect themselves from making the same mistakes. They relied more on instructors' previous designs to generate options for current problems. They judged their previous design based on instructors' solutions. As long as they did not discern any significant difference, they will stop exploring other design alternatives. Such satisficing tendencies (Ball, Maskill, & Ormerod, 1998; Ormerod, Mariani, Ball, & Lambell, 1999) restricted the impact of intergroup sharing of design rationale on groups' divergent thinking. For example, more than one student believed that he or she used the same approach for the next lab assignment as the one for the previous lab because their prior design fit instructors' solution well. Therefore, the motivation for intergroup sharing determines how students reflect on and use their previous design rationale, which is part of their creative design processes. By further investigating the various motivations, we could more precisely understand the effects of sharing design rationale across groups on group creativity.

Limitations

Our findings are constrained by the characteristic of the task. We did not deploy a direct measurement on the design product (i.e., the code), which arises from the concerns that the lab assignments in our study were relatively close-ended problem solving. In order to obtain a more precise assessment on design creativity, we plan to design more open-ended tasks.

Moreover, our results are limited by our measurement of rationale quality and creativity. Each judge in our study rated both rationale quality and creativity of every rationale document. Thus the positive correlation between rationale quality and creativity may result from the inherent consistency of each individual judge. In the future, we may employ more judges to assess rationale quality and design creativity separately, with each judge rating only one part. This can also balance the individual differences among judges. Alternatively, we can ask judges to qualitatively evaluate the relationship between the design rationales and creativity on the basis of their informed interpretation of rationale quality and creativity. To improve the interrater reliability, we can decompose our assessment criteria and facilitate further discussion on them with all judges.

Additionally, our analysis is confined by the class size. For instance, the required number of cases for stepwise multiple regression should be 40 times the number of independent variables, as recommended by Tabachnick and Fidell (1996). One way to approach a more robust conclusion is to recruit more participants.

Further Issues

Since our evaluation criteria on rationale quality are not exhaustive, feasibility of design alternatives and comprehensiveness of tradeoffs may not be the only quality facets related to creativity. Other properties we assessed in terms of rationale quality may also predict facets of creativity other than the three (i.e., novelty, persuasiveness, and insightfulness) measured in our study. Nonetheless, the positive correlation between rationale quality and creativity demonstrated in our empirical study connotes rationale articulation as a way to enhance creativity in software development. This certainly does not imply that documenting design rationale with any approach will necessarily lead to creativity enhancement, but rather inspires the dedication to investigate how to appropriate design rationale and what qualities of rationale should be amplified to support creativity in software development. The rationale qualities facilitating creativity discovered in our study will guide the effort to further elucidate the underpinning reasons why these qualities are critical to enhance design creativity.

The ways that rationale and creativity influence each other need further investigation in collaborative settings. Software design is a complex and ill-defined problem-solving process, which has increasingly demanded collaboration among individuals as the scale of projects grows. To achieve creativity in such situations, it is desired to keep track of the development process. Furthermore, mere individual intellects are hardly sufficient to attain creative design artifacts. Instead, the collective accomplishment will arise from the interaction between and among group members. The role of design rationale in these scenarios may not only involve facilitating individuals' analytic thinking but rather influencing the communication and cooperation processes when rationale is constructed and captured by collective effort. Therefore, it is intriguing and promising to further explore how design rationale articulation affects creative design processes of the group.

CONCLUSION AND FUTURE WORK

When designers think about rationale, they often tend to believe it suffocates design and undermines the possibility of creativity. In this paper, our study provides empirical evidence to argue that the relationship between rationale and creativity is more than contrast. Instead, rationales and rationale practices can be adapted to enhance creativity in design. Furthermore, based on our assessment of rationale quality and creativity, the feasibility of design alternatives and comprehensiveness of argumentation or tradeoffs have significant positive effects on the novelty of design alternatives, the persuasiveness, and the insightfulness of argumentation. These effects may derive from designers' internal evaluation and critical thinking on design alternatives. They are not bounded by the particular domain in our study (i.e., software design); instead, reasoning and critical thinking can have such effects in any other domain in which they are involved and creativity can happen. Therefore, we can expect that rationale and creativity are mutually facilitative in other domains beyond software engineering and design.

Similarly, our assessment criteria on rationale quality may also be adapted to real-world contexts outside of classrooms. Previous work allows evaluating rationale in terms of their usability by analyzing cognitive costs of different notational forms or in terms of their usefulness by observing their use and narrating anecdotes. Our rubrics provide a quantitative approach to evaluate the quality of design rationale, emphasizing the quality of critical thinking that is related to design creativity. It can be developed to assess real design practices by integrating concerns with organizational factors as well as management issues.

Yet to explain exactly how rationale facilitates creativity and why these two properties are strong predicators, we have to investigate their effects on the creative processes by refining survey questions and collecting more qualitative data. For comprehensiveness, we may also need to look at tradeoffs from pros and cons separately.

Our observation on sharing rationale across groups stimulates us to explore further the various motivations for sharing design rationale and design reuse to understand when to facilitate intergroup sharing. This is worth investigation because design rationale is usually expected to convey the reasoning and decision process of other designers.

In the even longer term, understanding the benefits of design rationale for creativity in software development will inform how to build tools to support creative design.

ENDNOTE

1. This was a workshop on creativity and rationale in software design sponsored by NSF CreativeIT program. It was held at University Park, PA in June, 2008. John M. Carroll wrote a manifesto, "The Essential Tension of Creativity and Rationale in Software Design," for this workshop.

REFERENCES

Amabile, T. M. (1996). Creativity in context. Boulder, CO. USA; Westview Press.

Ball, L. J., Lambell, N. J., Ormerod, T. C., Slavin, S., & Mariani, J. A. (2001). Representing design rationale to support innovative design reuse: a minimalist approach. *Automation in Construction*, 10, 663–674.

Ball, L. J., Maskill, L., & Ormerod, T. C. (1998). Satisficing in engineering design: Causes, consequences and implications for design support. *Automation in construction, 7,* 213–227.

Bellotti, V. (1993). Integrating theoreticians' and practitioners' perspectives with design rationale. In S. Ashlund, K. Mullet, & A. Henderson (Eds.), *InterCHI'93: Human Factors in Computing Systems* (pp. 101–106). New York, USA: ACM Press.

Buckingham Shum, S. B. (1993). QOC design rationale retrieval: a cognitive task analysis and design implication. Retrieved May 26, 2010, from http://ftp.xrce.xerox.com/Publications/Attachments/1993-105/EPC-1993-105.pdf

Buckingham Shum, S. B. (1996). Analyzing the usability of a design rationale notation. In T. P. Moran, & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 185–215). Mahwah, NJ, USA: Lawrence Erlbaum Associates, Inc.

Buckingham Shum, S. B., & Hammond, N. (1994). Argumentation-based design rationale: What use at what cost. *International Journal of Human-Computer Studies, 40,* 603–652.

Buckingham Shum, S., MacLean, A., Forder, J., & Hammond, N. (1993). Summarising the evolution of design concepts within a design rationale framework. In S. Ashlund, K. Mullet, A. Henderson, E. Hollnagel, & T. White (Eds.), *InterCHI'93: the 1993 Conference on Human Factors in Computing Systems* (pp. 43–44). New York: ACM Press.

Burgess-Yakemovic, K. C., & Conklin, J. (1990). Report on a development project use of an issue-based information system. In F. Halasz (Ed.), *CSCW'90: Computer Supported Cooperative Work* (pp. 105–118). New York, USA: ACM Press.

Carroll, J. M. (2009, June). *The essential tension of creativity and rationale in software design*. Manifesto from the workshop on Creativity and Rationale in Software Design, State College, PA, USA

Carroll, J. M., & Rosson, M. B. (2003). Design rationale as theory. In J. M. Carroll (Ed.), *HCI models, theories, and frameworks: Toward an interdisciplinary science* (pp. 531–561). San Francisco: Morgan Kaufmann.

Conklin, E. J., & Yakemovic, K. C. B. (1991). A process-oriented approach to design rationale. *Human-Computer Interaction*, *6*, 357–391.

Csikszentmihalyi, M. (1996). Creativity: Flow and the psychology of discovery and invention. New York: Harper Collins Publishers.

Csikszentmihalyi, M. (1999). Implications of a systems perspective for the study of creativity. In R. J. Sternberg (Ed.), *Handbook of creativity* (pp. 313–335). New York: Cambridge University Press.

Farooq, U. (2008). *Supporting creativity: Investigating the role of awareness in distributed collaboration*. Doctoral dissertation, The Pennsylvania State University. Retrieved May 26, 2010, from http://etda.libraries.psu.edu/theses/approved/WorldWideIndex/ETD-2990/index.html

Farooq, U., Carroll, J. M., & Ganoe, C. H. (2005). Supporting creativity in distributed scientific communities. In M. Pendergast, K. Schmidt, G. Mark, & M. Ackerman (Eds.), *GROUP 2005: The 2005 International ACM SIGGROUP Conference on Supporting Group Work* (pp. 217–226). New York: ACM Press.

Fischer, G. (2004). Social creativity: Turning barriers into opportunities for collaborative design. In A. Clement & P. Besselaar (Eds.), *PDC 2004: The Eighth Conference on Participatory Design: Artful integration: Interweaving media, materials and practices* (pp. 152–161). New York: ACM Press.

Fischer, G., Lemke, A. C., McCall, R., & Morch, A. I. (1991). Making argumentation serve design. *Human-Computer Interaction*, *6*, 393–419.

Fleischmann, K. R. (2006). Boundary objects with agency: A method for studying the design-use interface. *The Information Society, 22,* 77–87.

Gardner, H. (1993). Seven creators of the modern era. *Creativity,* 28–47.

Garrison, D. R. (1992). Critical thinking and self-directed learning in adult education: An analysis of responsibility and control issues. *Adult Education Quarterly, 42*(3), 136–148.

Guilford, J. P. (1967). The nature of human intelligence. New York: McGraw-Hill Companies.

Henri, F. (1991). Computer conferencing and content analysis. In A. R. Kaye (Ed.), *Collaborative learning through computer conferencing: The Najadeen papers* (pp. 117-136). London: Springer-Verlag.

Henry, J. (2004). Creative collaboration in organisational settings. In D. Miell & K. Littleton (Eds.), *Collaborative creativity: Contemporary perspectives* (pp. 158–175). London: Free Association Books.

Houtz, J. C., Montgomery, C., Kirkpatrick, L., & Feldhusen, J. F. (1979). Relationship among measures of evaluation ability (problem solving), creative thinking, and intelligence. *Contemporary Educational Psychology, 4*(1), 47–54.

Kunz, W., & Rittel, H. W. J. (1970). Issues as elements of information systems. Retrieved May 26, 2010, from http://www.technologyforge.net/enma/6010/ENMA288LecturesAssignments/ENMA288DTD/ENMA288DT4/KunzRittel.pdf

Lee, J., & Lai, K.Y. (1991). What's in design rationale? *Human-Computer Interaction, 6,* 251–280.

MacLean, A., Young, R. M., Bellotti, V. M. E., & Moran, T. P. (1991). Questions, options, and criteria: Elements of design space analysis. *Human-Computer Interaction, 6,* 201–250.

MacLean, A., Young, R. M., & Moran, T. P. (1989). Design rationale: the argument behind the artifact. In J. Carrasco Chew, & J. Whiteside (Eds.), *CHI'89: Human Factors in Computing Systems* (pp. 247–252). New York: ACM Press.

Mayer, R. E. (1999). Fifty years of creativity research. In R. J. Sternberg (Ed.), *Handbook of creativity* (pp. 449–460). New York: Cambridge University Press.

McKerlie, D., & MacLean, A. (1994). Reasoning with design rationale: Practical experience with design space analysis. *Design Studies, 15,* 214–226.

Moran, T. P., & Carroll, J. M. (1996). Overview of design rationale. In T. P. Moran & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 1–9). Mahwah, NJ, USA: Lawrence Erlbaum Associates, Inc.

Newman, D. R., Johnson, C., Webb, B., & Cochrane, C. (1997). Evaluating the quality of learning in computer supported co-operative learning. *Journal of the American Society for Information Science, 48,* 484–495.

O'Brien, R. M. (2007). A caution regarding rules of thumb for variance inflation factors. *Quality and Quantity, 41,* 673–690.

Ormerod, T. C., Mariani, J., Ball, L. J., & Lambell, N. (1999). Desperado: Three-in-one indexing for innovative design. In M. A. Sasse & C. Johnson (Eds.), *INTERACT'99: The Seventh IFIP Conference on Human-Computer Interaction* (pp. 336–343). London: IOS Press.

Potts, C. (1996). Supporting software design: integrating design methods and design rationale. In T. P. Moran, & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 295–321). Mahwah, NJ, USA: Lawrence Erlbaum Associates, Inc.

Runco, M. A. (1992). Children's divergent thinking and creative ideation. *Developmental Review, 12,* 233–264.

Shipman, F. M., & McCall, R. J. (1997). Integrating different perspectives on design rationale: Supporting the emergence of design rationale from design communication. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 11*(2), 141–154.

Singley, M. K., & Carroll, J. M. (1996). Synthesis by analysis: five modes of reasoning that guide design. In T. P. Moran, & J. M. Carroll (Eds.), *Design rationale: Concepts, techniques, and use* (pp. 241–266). Mahwah, NJ, USA: Lawrence Erlbaum Associates, Inc.

Sternberg, R. J. (1985). Beyond IQ: A triarchic theory of human intelligence. New York: Cambridge University Press.

Sternberg, R. J. (2006). The nature of creativity. *Creativity Research Journal, 18*(1), 87–98.

Sternberg, R. J., & Lubart, T. I. (1991). An investment theory of creativity and its development. *Human Development, 34*(1), 1–31.

Sternberg, R. J., & Lubart, T. I. (1999). The concept of creativity: Prospects and paradigms. In R. J. Sternberg (Ed.), *Handbook of creativity* (pp. 3–15). New York: Cambridge University Press.

Tabachnick, B. G., & Fidell, L. S. (1996). Using multivariate statistics. New York: Harper Collins Publishers.

West, M. A. (2003). Innovation implementation in work teams. In P. B. Paulus, & B. A. Nijstad (Eds.), *Group creativity: Innovation through collaboration* (pp. 245–276). New York: Oxford University Press

Woodman, R. W., Sawyer, J. E., & Griffin, R. W. (1993). Toward a theory of organizational creativity. *Academy of Management Review, 18*(2), 293–321.

Authors' Note

This research was supported by John M. Carroll and Umer Farooq's grants NSF SGER IIS-0749172 and NSF CreativeIT Workshop IIS-0742392. We thank instructors of the course in which we implemented our study, Craig Ganoe and John Daughtry, for their assistance in the study design and coordination with participants. We also thank the teaching assistants of the course Ishita Ghosh and Haibin Liu for assessing design rationale documents.

All correspondence should be addressed to
Jing Wang
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16802 USA
jzw143@psu.edu

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

APPENDIX

Evaluation Criteria

Quality of Design Rationale (1 = *very poor*, 5 = *very good*)

Toughest Design Problem Identification:
Does the statement of toughest design problem capture a critical issue of this lab?

Relevance of alternatives:
Can the design alternatives solve the problem stated?

Feasibility of alternatives:
Can the design alternatives be implemented by using the technique taught in class?

Comprehensiveness of tradeoffs (pros and cons):
Do the tradeoffs reveal main concerns about each design alternative?

Decision:
Is the design alternative selected the optimal solution?

Clarity of articulation:
Can the report be well understood?

Creativity of Design

Novelty of design alternatives:
Are the design alternatives novel?

Persuasiveness of tradeoffs:
Are the tradeoffs persuasive?

Insightfulness of tradeoffs:
Do the tradeoffs provide insightful justification of design alternatives?

Human Technology:
An Interdisciplinary Journal on Humans in ICT Environments

www.humantechnology.jyu.fi