Understanding Innovation

Hasso Plattner
Christoph Meinel
Larry Leifer *Editors*

Design Thinking Research

Studying Co-Creation in Practice

Springer

Understanding Innovation

Series Editors
Christoph Meinel
Larry Leifer

For other titles published in this series, go to
http://www.springer.com/series/8802

Hasso Plattner • Christoph Meinel • Larry Leifer
Editors

Design Thinking Research

Studying Co-Creation in Practice

Springer

Editors
Prof. Hasso Plattner
Hasso Plattner Institute for IT Systems Engineering
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam
Germany
hasso.plattner@sap.com

Prof.Dr. Christoph Meinel
Hasso Plattner Institute for IT Systems Engineering
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam
Germany
meinel@hpi.uni-potsdam.de

Prof.Dr. Larry Leifer
Center for Design Research (CDR)
Stanford University
424 Panama Mall
Stanford, CA 94305-2232
USA
leifer@cdr.stanford.edu

ISBN 978-3-642-21642-8 e-ISBN 978-3-642-21643-5
DOI 10.1007/978-3-642-21643-5
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011935346

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This volume presents the second comprehensive collection of research studies carried out by the HPI-Stanford Design Thinking Research Program, a joint program of the Hasso-Plattner-Institute of Design at Stanford University in California and the Hasso-Plattner-Institute (HPI) for IT Systems Engineering in Potsdam, Germany.

Design Thinking is a framework to understand the issues people are experiencing in their daily lives and to generate accordingly helpful innovations for them. In Design Thinking, interdisciplinary teams set off to learn about people's concerns and the obstacles they are facing. By means of Design Thinking, the teams head for solutions regarding the identified problems which are supposed to be genuinely new as well as extensively useful. Thus, Design Thinking teams work towards products or services that are technically feasible, economically viable and, in addition, truly desirable for people.

While practice has proven that Design Thinking is a promising approach for companies in particular and society in general, looking at it systematically and with scientific rigor is a rather new endeavor. Therefore, we may still be curious in what ways exactly this research will shape our understanding of innovations in general and of Design Thinking in particular. The predominant questions to us are: if we will arrive at new and sensible descriptions of how to generate innovations, and if and to what extent we will apply new methods or tools. Or, might it even be possible for us to change our ways?

But now that I have contrasted Design Thinking as a long-established practice, and Design Thinking research as a rather new outlook on the matter, let me point out an issue that is pivotal to both: communication. In Design Thinking as a practice, much is done to facilitate communication – be that within the design team, when consulting users or other stakeholders. But communication is likely to be as central to the research endeavor as it is to Design Thinking itself.

The evolution of our society runs parallel to the evolution of communication: from the sign language of early hominids, the development of spoken languages, the introduction of script, the conveyance of information via signals, the invention of printing, the distribution of information thru phone, radio and television, to the advent of the internet. The more information we can share, the more likely we are to progress. In the early days of civilization, a joint location was the common precondition of information exchange. Once we were able to reproduce large amounts of information by printing books and journals, spreading them to other locations and times, the process of generating new ideas and technologies sped up immensely.

If Design Thinking Research is not to be an idle exercise, communication is certainly crucial: we need to share our observations and thoughts, we need to sift them, to meld them, and, essentially, to make something new and valuable out of them. Thus, it is a rewarding pleasure for me to disclose and share the results of our latest research work in this book. May it contribute to a prospering discussion and stimulate new, sensible solutions.

Potsdam/Palo Alto Hasso Plattner

Contents

Contributors

Avantika Agarwal Center for Design Research, Stanford University, 424 Panama Mall, Building 560, 11950, Stanford, CA 94309, USA, avantika@stanford.edu

Shilajeet Banerjee Department of Mechanical Engineering, Stanford University, Building 550, Stanford, CA 94305-4021, USA, banny@stanford.edu

Rebecca Currano Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, bcurrano@stanford.edu

Mark Cutkosky Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, cutkosky@stanford.edu

Steven P. Dow Human-Computer Interaction Group, Stanford University, Gates Computer Science Building, 353 Serra Mall, Stanford, CA 94305, USA, spdow@stanford.edu

Jonathan Edelman Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, edelman2@cdr.stanford.edu

Gregor Gabrysiak System Analysis and Modeling Group, Hasso-Plattner-Institute for IT Systems Engineering, University of Potsdam, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany, gregor.gabrysiak@hpi.uni-potsdam.de

Lutz Gericke Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany, lutz.gericke@hpi.uni-potsdam.de

Holger Giese System Analysis and Modeling Group, Hasso-Plattner-Institute for IT Systems Engineering at the University of Potsdam, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany, holger.giese@hpi.uni-potsdam.de

Raja Gumienny Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany, raja.gumienny@hpi.uni-potsdam.de

Christopher Han Department of Management Science and Engineering, Stanford University, Huang Engineering Center, 475 Via Ortega, Stanford, CA 94305-4121, USA, Christopher.han@stanford.edu

Pamela Hinds Department of Management Science & Engineering, Stanford University, Stanford, CA 94305-4026, USA, phinds@stanford.edu

Robert Hirschfeld Software Architecture Group, Hasso-Plattner-Institute, University of Potsdam, P.O. Box 900460, 14440 Potsdam, Germany, hirschfeld@hpi.uni-potsdam.de

Wendy Ju Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, wendyju@cdr.stanford.edu

Malte Jung Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, mjung@stanford.edu

Hannah Hyunjee Kim Department of Management Science & Engineering, Stanford University, Stanford, CA 94305-4026, USA

Scott R. Klemmer Human-Computer Interaction Group, Stanford University, Gates Computer Science Building, 353 Serra Mall, Stanford, CA 94305, USA, srk@cs.stanford.edu

Eva Köppen Hasso Plattner Institute, University of Potsdam, P.O. Box 900460, 14440 Potsdam, Germany, eva.koeppen@hpi.uni-potsdam.de

Thomas Kowark Hasso Plattner Institute, University of Potsdam, August-Bebel-Str. 88, 14482 Potsdam, Germany, thomas.kowark@hpi.uni-potsdam.de

Greg L. Kress Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, glkress@stanford.edu

Micah Lande Department of Engineering, Arizona State University, 7231 E. Sonoran Arroyo Mall, Mesa, AZ 85212, USA, Micah.Lande@asu.edu

Sabine Lang Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany, sabine.lang@hpi.uni-potsdam.de

Larry Leifer Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, leifer@cdr.stanford.edu

Tilmann Lindberg Hasso Plattner Institute, University of Potsdam, P.O. Box 900460, 14440 Potsdam, Germany, tilmann.lindberg@hpi.uni-potsdam.de

Lei Liu Department of Management Science & Engineering, Stanford University, Stanford, CA 94305-4026, USA, liulei@stanford.edu

Alexander Luebbe Hasso Plattner Institute, University of Potsdam, P.O. Box 900460, 14440 Potsdam, Germany, alexander.luebbe@hpi.uni-potsdam.de

Sophia Mark Center for Design Research, Stanford University, 424 Panama Mall, Building 560, 11950, Stanford, CA 94309, USA, samark@stanford.edu

Christoph Meinel Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany, meinel@hpi.uni-potsdam.de

Siddharth Mishra Department of Management Science & Engineering, Stanford University, Stanford, CA 94305-4026, USA

Christine Noweski Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany, christine.noweski@hpi.uni-potsdam.de

Cole Paterson Center for Design Research, Stanford University, 424 Panama Mall, Building 560, 11950 Stanford, CA 94309, USA, colepatt@stanford.edu

Ingo Rauth Hasso Plattner Institute, University of Potsdam, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany

Mark Schar Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, mfschar@stanford.edu

Andreas Seibel System Analysis and Modeling Group, Hasso Plattner Institute for IT Systems Engineering, University of Potsdam, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany, andreas.seibel@hpi.uni-potsdam.de

David Sirkin Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, sirkin@cdr.stanford.edu

Neeraj Sonalkar Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, sonalkar@stanford.edu

Bastian Steinert Software Architecture Group, Hasso Plattner Institute, P.O. Box 900460, 14440 Potsdam, Germany, bastian.steinert@hpi.uni-potsdam.de

Martin Steinert Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA, steinert@stanford.edu

Julia von Thienen Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany, julia.vonthienen@hpi.uni-potsdam.de

Matthias Uflacker Hasso Plattner Institute, University of Potsdam, August-Bebel-Str. 88, 14482 Potsdam, Germany, matthias.uflacker@hpi.uni-potsdam.de

Mathias Weske Hasso Plattner Institute, P.O. Box 900460, 14440 Potsdam, Germany, mathias.weske@hpi.uni-potsdam.de

Alexander Zeier Hasso Plattner Institute, University of Potsdam, August-Bebel-Str. 88, 14482 Potsdam, Germany, alexander.zeier@hpi.uni-potsdam.de

Design Thinking Research

Christoph Meinel and Larry Leifer

1 Design Thinking as Innovation Foray

Innovators love creating an idea and are constantly trying to invent new things or to improve already existing products and services. When people are creating ideas, they get excited about it, they take ownership, and they make commitments. They do everything possible to make sure the concept can become a reality that others appreciate. When the creation process is performed by a team the effort is magnified and the energy multiplied. For this reason, we consider co-creation as a crucial aspect in the complex socio-technical field of design thinking in action.

This year's book summarizes the results of the 2nd year in the Design Thinking Research Program, a joint venture of Stanford University in Palo Alto and Hasso Plattner Institute in Potsdam. We have taken a closer look at the issue of co-creation from different points-of-view. The concept behind co-creation may sound simple, however, it is both an essential element of Design Thinking and highly complex. It is about creating positive synergies for all parties involved.

The concept of co-creation can also be applied to the phase in which new ideas, and those related to them, start to influence companies, the economy, our culture, and society. The perpetual pursuit for inventions, new creations and innovations is inherent to human nature. Looking back on the history of mankind it becomes obvious that people have always been looking for a better way, and a better life. Innovation is not new, but the pace has changed. Quicker and faster becomes the mantra. While

C. Meinel (✉)
Hasso Plattner Institute, P.O. Box 900460, 14440 Potsdam, Germany
e-mail: meinel@hpi.uni-potsdam.de

L. Leifer
Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA
e-mail: leifer@cdr.stanford.edu

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_1,

innovation may once have been coincidental, a by-product of some other intention, it is today the ultimate goal of our research and development institutions.

An innovation can also evolve from putting together already existing pieces in a new way. The same goes for the process of co-creation. Many different influences, ideas and goals must be synthesized by the team that is co-creating with the intention to find something new, something ground breaking, and an innovation that sells.

The "**Innovation Foray: Hunter-Gatherer**" model has emerged from our ways of thinking, discussing, and testing to increase the probability of successful innovation from research, development, and marketing activities. The model has proven useful as a means to communicate the core ideas in design thinking, meaning, the key ingredients, to others. It emphasizes that there are different roles to be played in the activities of innovating, and gives flavor to what these roles must be: (1) the Hunter and: (2) the Gatherer. There is a time to hunt and a time to gather. There are times to seek the next big thing and times to deliver the next big thing. The **Innovation Foray** is all about finding and delivering "ideas that sell." The enterprise must understand this distinction and do both well.

Figure 1 is the first published embodiment of the **Hunter-Gatherer** metaphor for innovation and delivery. We have made a personal commitment not to render the concept in power point slides. This is because any seemingly final and definitive figure would destroy the meaning of the metaphor that every hunt has a unique path. The model is all about actions, activities, and movements, what we do in the moment, what we learn on the fly, and how we discover the unknown in unfamiliar terrain. There is no roadmap. Don't even ask for one. It would be wrong minded to

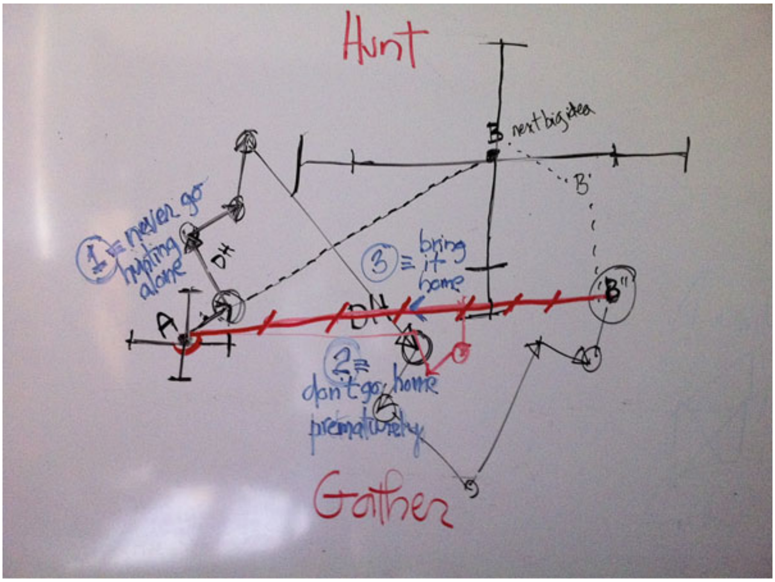


Fig. 1 This white-board sketch represents a design thinking conversation that was seeking to define the research question and methodology for a pending research proposal

do so. We do not want to even imply a fixed model of how to come up with innovations, but we do hope to offer a model that will help to everyone grasp the seemingly diffuse endeavor of seeking (=hunting for) an innovation – a metaphor that inspires sensible action.

This model also describes the dynamics of design thinking. These activities have time constants on the order of milliseconds. The **Hunter-Gatherer** Model is about enfolding events, awareness, observation, and real time intervention. It is the whole mind-body alertness of the hunting team and the optimized mind set of the gathering team.

We are hunting for an "idea that sells," and it had better be big. We want to solve a problem, perhaps remove the problem itself through design thinking, new products, and/or remarkable services.

It documents the necessarily ad-hoc (discovery journey) "**Innovation Foray: hunter-gatherer model**" for the activities and thinking processes design researchers employ to do design-research. They are looking for the next "big idea" using "Designerly Ways of Thinking" (Nigel Cross, 2005). You will find an expanded discussion of the process representation in the following text.

We find that the designerly ways of approaching design research are decidedly different than the approach derived from the physical sciences. It is highly adaptive and evidence based versus theory centric. When hunting for the next big idea it is absolutely essential that we observe and act upon all of the signals at hand. The associated activities (behaviors) differ from those used to validate theories and frame works. In Fig. 1 we have captured one example of a search scenario related to finding a better way to communicate new ideas within small teams that are widely distributed in space and time. We imagine starting on the lower left at point A. Vertical and horizontal bars roughly represent the magnitude of our uncertainty about the present. We have an initial understanding of what the need/problem we are seeking to satisfy or solve. The uncertainty error bars are rather concise at point A. Towards the upper right, surrounding point B, we imagine the concept zone we are targeting. Here, there are many concepts that might allow us to solve the problem. Until we get there, it is unclear which ideas we may find, how valuable they will turn out to be, how many alternative understandings there could be and how to find them. Thus, the uncertainty bars surrounding point B are nothing but huge.

We are about to go hunting and we invoke the **Human Rule**, "**never go hunting alone**."

Go hunting in teams, choosing one that is small and agile with a maximum of diversity. Include at least one really good hunter for sure, but don't forget to also include good gathering talent too. We need their insights as to whether or not the "big idea" can actually be delivered. We recommend including a realist, someone who pays attention to time, money, and the weather. And it is wise to include people persons who pay attention to team dynamics, feelings and communication. At the outset, we have no idea when or where we will find that big idea and therefore the expeditionary team needs to be carefully chosen, trained, and well provisioned.

Let the divergence begin. First, make a straightforward move towards the perceived target, build one or more prototypes, test them against the known, and discover the unknown. Learn. And then abduct, the upward left arrow after the first move in Fig. 1. Abductive logic tells us that no amount of inductive and deductive thinking will reveal the unknown. It takes abduction to produce a discovery. Prototype, test, and learn. Then repeat.

Note that near the word divergence, the abduction was very large. We like to call this a “dark horse” prototype. It explores the most extreme ideas, meaning, those that might initially seem impossible, but if realized, they would be BIG. In Fig. 1 the dark horse prototype took the hunters into entirely new concept space where they discovered that the idea they were seeking was really closer to B’ in the lower right quadrant. Along the path they invoked the **Ambiguity Rule**, **“never go home prematurely.”** The journey has been long. The ambiguity has been frustrating. Team dynamics have become fragile. Is it time to quit, admit defeat, and get some sleep? Never. When you get home others in your organization may say “thank you,” but they were really hoping to say “wow.”

Keep hunting, keep abducting, keep learning and in time you will get that gut feeling that the big idea is just around the corner. You see its shape, but not yet the details. And then, pop, things come together, the fragments become coherent, the story has a wow ending at B’ in Fig. 1. Now what?

Invoke the **Re-Design Rule** and make it **Tangible**. It is imperative to understand how these needs have been addressed in the past. Now gather the embodied idea and all the evidence your team discovered during the hunt and “**take it on home**.” Make it real. Make plans, marshal resources, optimize, market, manufacture, distribute, and service. Do all the things we’ve been trained to do in engineering and science. Most of our organizations are experts in these regards and activities. These are the linear-thinking optimization steps. They are great, if, and only if, we apply them to great ideas, the product to the hunt.

But we don’t educate people to hunt. We don’t let them go hunting. These shortcomings inhibit, even prohibit innovation in our companies, schools and universities. **Innovation foray** is capturing the intellectual challenge of hunting for the next big idea, whether it be theoretical, empirical, or commercial new product development. It makes it obvious that there is a critical distinction between the behaviors (actions) of hunters tasked to find the next big idea versus the activities of gatherers: those who are tasked with implementing the big idea. We assure you that life requires hunters and gatherers, companies do, and academia is beginning to see the need to understand the symbiotic relationship between hunters and gathers in pursuit of innovations.

2 A Proof of Successful Co-creation – The HPI-Stanford Design Thinking Research Program

Starting in 2008, the HPI-Stanford Design Thinking Research Program was financed by the Hasso Plattner Foundation.

Program Vision. The research program engages multidisciplinary research teams. Scientifically they investigate the phenomena of **innovation** in all its holistic dimensions. Researchers are especially encouraged to develop ambitious, long-term exploration projects that integrate technical, business, and human points of view using design thinking tools and methods.

The HPI-Stanford Design Thinking Research Program is a rigorous academic research effort, aiming to understand the scientific foundations of how and why the innovation methods of Design Thinking work. Its researchers study the complex interactions between members of multidisciplinary teams that engage in design co-creation. Beyond descriptive understanding, the goal of the program is to develop metrics that allow assessment and prediction of team performance in order to facilitate real-time performance management. Researchers are encouraged to design, develop and evaluate innovative (analogue and digital) tools that support teams in their creative work. One program focus is on exploring the use of Design Thinking methods in the field of Information technology and IT systems engineering. An important feature of this domain is the need for creative collaboration across spatial and temporal boundaries. In the context of disciplinary diversity, the question of how Design Thinking methods mesh with traditional engineering and management approaches is addressed. Why does the structure of successful design teams differ substantially from traditional corporate structures?

The Program involves multidisciplinary research teams from diverging backgrounds such as science, engineering, design, and the humanities. A prerequisite is being passionate about developing ambitious, long-term, discovery research projects is the need to expand our understanding of Design Thinking in its technical, business, and human dimensions.

Program Priorities. A strong cooperation in the offering of both Design Thinking education programs is a priority. Both of the design thinking schools at Stanford University and the Hasso Plattner Institute in Potsdam focus on fostering collaboration between researchers of Stanford University and the Hasso Plattner Institute. It is about teamwork.

Multi-year funding favors projects that set new research priorities for this emergent knowledge domain. Projects are selected based on intellectual merit and evidence of open collaboration. The following guiding research questions are of special interest:

- What are people really thinking and doing when they are engaged in creative design innovation? How can new frameworks, tools, systems, and methods augment, capture, and reuse successful practices?
- What is the impact on technology, business, and human performance when design thinking is practiced? How do the tools, systems, and methods really work to get the innovation you want when you want it? How do they fail?

3 The Program Book

3.1 Design Thinking – Envisioning Co-creation

The overall topic of co-creation is leading the way through this book, the second volume of the series "Understanding Innovation – Design Thinking Research". Starting without a road map, we are designing the hunt. We are creating as we go. We are producing a multi-faceted foundation for looking at design thinking from different perspectives and through the affordances of new creativity support tools. The next step on the way is exploring the concept of creative collaboration that crosses spatial and temporal boundaries. This is design thinking in the information age.

3.2 Part I: Road Maps for Design Thinking

The **second chapter** entitled "Tele-Board: Follow the Traces of Your Design Process History" explores how digital tools can be integrated into creative work settings. The authors Lutz Gericke, Raja Gumienny and Christoph Meinel from Hasso Plattner Institute provide a comprehensive description of how to extend the concept of creative collaboration across spatial borders. The chapter deals with real-time design work at different locations. It includes functions for time-delayed interaction. It describes how design teams can be supported to fulfil their tasks more efficiently in dispersed teams and how design researchers can deepen their understanding of how designers work in a predominantly digital setting. They are focused on documenting the hunt.

The **third chapter** by Jonathan Edelman, Avantika Agarwal, Cole Paterson, Sophia Mark and Larry Leifer from Stanford University called "Understanding Radical Breaks" deals with radical transitions in the design thinking process. It explores how small horizontally organized design teams perform radical redesigns, a process in which designers make a major departure from the provided artefact. They introduce three imbricated concepts as a mechanism for understanding how the design process determines the design outcomes: scoping, behaviours, and shared media. They are focused on the hunter's strategy.

The authors of the **fourth chapter**, Julia von Thienen, Christine Noweski, Ingo Rauth, Christoph Meinel and Sabine Lang take a close look at the context in which co-creation takes place. How do spaces (e.g. room-setups) influence the people who work in them?

What places propel towards or thwart innovations? Their article entitled "If You Want to Know Who You Are, Tell Me Where You Are: The Importance of Places" reports an experimental study in which a variation of the spatial setup has an exorbitant effect on how innovative people are and on the amount of personal initiative they show. Going beyond the context of innovation, the authors devise a scheme in order to predict how people are going to behave and feel at a particular place. Correspondingly, the scheme may ease the design of places such that they truly serve the functions for which they are intended. They are focused on the hunter's territory.

Chapter five is captioned with the title "Creativity and Culture: State of the Art" and deals with the topic of how creativity is defined across cultures. The authors Hannah Kim, Siddarth Mishra, Pamela Hinds and Lei Liu from Stanford University are identifying key stimuli for fostering creativity in different cultures, and for understanding how creative performance differs by culture. Based on the current research on this topic they found that research on creativity and culture is biased toward Western conceptions and that this Western-biased view of creativity leads to the conclusion that the West shows greater creative performance than the East. These different perceptions are based on the fact that there are different perceptions of creativity in different cultures. The authors have compared several factors such as extrinsic vs. intrinsic motivation or conformity pressure and are coming to the conclusion that new directions have to be found for research on creativity and culture as the former standards that have been applied so far are no longer valid. They are focused on the hunting team and expectations of the gathering team.

3.3 Part II: Creative Tools and the Importance of Prototypes in Design Thinking

Creative tools and prototypes are essential elements in the Design Thinking Process. In the **sixth chapter** entitled "Design Loupes: A Bifocal Study to Improve the Management of Engineering Design Innovation by Co-evaluation of the Design Process and Information Sharing Activity" authors Rebecca Currano, Martin Steinert, and Larry Leifer from Stanford University systematically explore the individual designer's inherent reflective loupe. The aim of this research is to deepen the understanding of the cognitive mechanisms of reflective design loupes. This research serves as a next step in addressing the challenge of the Design Loupes project, which in the previous year focused on the existence and modelling structure of general design loupes. They are focused on the inflexion points where the hunting team breaks with the past to abduct to a divergent goal.

In the subsequent **seventh chapter**, Gregor Gabrysiak, Holger Giese and Andreas Seibel from Hasso Plattner Institute explore how and to what extent design thinking benefits from the usage of tangible prototypes to communicate, validate and explore insights and design ideas. In their chapter headlined "Towards Next-Generation Design Thinking II. Virtual Multi-User Software Prototypes", the central question of their research is how prototyping can become more feasible for domains dealing with immaterial objects and intangible concepts. Based on their research work that was executed in the first program year that was dealing with the conceptualization of tangible prototypes of multiuser software systems based in executable formal models, this chapter elaborates how these models can be experienced and evaluated by end users through simulation and animation. Furthermore, the authors discuss the results of an evaluation comparing the usability of their approach with traditional formal and informal modelling approaches. They are focused on the gathering team and the communication challenges they face.

Authors Steven Dow, Alana Glassco, Jonathan Kass, Melissa Schwarz, Daniel Schwartz, and Scott Klemmer from Stanford University explore the value of prototyping in Design Thinking. In the **eighth chapter**, entitled "Parallel Prototyping Leads to Better Design Results, More Divergence, and Increased Self-Efficacy," the authors run an experiment where participants create Web banner ads. The study examines whether creating and receiving feedback on multiple prototypes in parallel – rather than simply iterating in serial – affects learning, self-efficacy, and design exploration. The results show that parallel prototyping produces better design results; more web visitors clicked on parallel ads and expert judges rated them higher. Moreover, parallel prototypers create more divergent ideas and react more positively to critique. The chapter outlines a theoretical foundation for why parallel prototyping produces better design results and discusses the implications for design education. They are focused on the hunting team's search pattern, finding that parallel paths outperform hunting in single file.

3.4 Part III: Distributed Design Collaboration and Teamwork in Design Thinking

According to the Human Rule in Design Thinking that says that all design activity is ultimately social in nature and that has been elaborated in the first book "Design Thinking: Understand – Improve – Apply", it is the imperative to solve technical problems from a human-centric point of view. But it is equally important to understand the psychology of co-creation activity amongst team members on the design team. The chapters in Part III of this book are all dealing with collaboration, communication and team formation.

Chapter nine, "Towards a Shared Platform for Virtual Collaboration Monitoring in Design Research" has been written by Thomas Kowark, Matthias Uflacker, and Alexander Zeier from Hasso Plattner Institute. Based on prior research results that provided new insights into the collaboration behaviour during the early phases of concept creation and prototyping, they are drafting in this article an architecture for a platform that aims to establish 'out- of-the-box' monitoring capabilities for virtual team environments and to facilitate the sharing and evaluation of recorded activities within a larger research community. Furthermore, they are presenting the results and experiences they have gained from a recently conducted observation of software engineering teams that demonstrates the flexibility and applicability of their instrument and underlines hereby their vision of a common service for capturing and analysing virtual collaboration activities that promote comparative research and team diagnostics in engineering design. They are focused on instrumenting design team activities while engaged in hunting and gathering.

David Sirkin and Wendy Ju from Stanford University take a closer look in **chapter ten** at the allegory of design as a conversation. Their chapter, entitled "Communicating Meaning & Role in Distributed Design Collaboration: How Crowdsourced Users Help Inform the Design of Telepresence Robotics" emphasizes the language of design collaboration, which is made up of words and images, actions and behaviors. Focusing on the role of gesture in design collaboration, they ran two studies to explore how embodied telepresence robots can support better communication in distributed teams. The research team found out that when the robots displayed physical motions, teammates on both sides of the interaction were perceived as more involved in the conversation, more composed in demeanor, and more equal in stature. As a next step, they are planning on applying these requirements to the design of their next generation communication avatar. They are focused on tangible communication between distributed team members engaged in hunting and gathering.

In **chapter eleven** that is entitled "Teamology – The Art and Science of Design Team Formation", the authors Greg Kress and Mark Schar from Stanford University are thoroughly analysing the thesis that all design work is collaborative work. The phenomenon of the "design team"is increasingly common in both industry and project-based education. Existing organizational behaviour research has shown that diversity on a team has mixed and frequently negative effects, particularly when outward indicators such as gender, ethnicity, age and experience measure diversity. However, relatively little research has been conducted on the problem solving capabilities and preferences of individual team members, or "team cognitive diversity." They found out that students with similar backgrounds and experience level reveal a wide variety of cognitive problem solving preferences. Additionally, they discovered that overall cognitive diversity does not appear to correlate with overall team project performance. However, team project performance positively correlates with team level "social sensitivity," the cognitive ability to relate to other team members problem solving preferences. Finally, cognitive diversity does not correlate with either individual and team level satisfaction, indicating that cognitive differences may be successfully accommodated over the life of the project. The implications of these findings are discussed in this chapter. They are focused on whom you should go hunting with.

The authors of **chapter twelve**, Micah Lande, Neeraj Sonalkar, Malte Jung, Christopher Han, and Banny Banerjee from Stanford University are taking a closer look on the impact of emotion coding, improvisation, ethnography, social psychology, and decision analysis into key metrics that are being called Design Thinking Metrics (DTM). They applied these metrics to analyze and assess videos of software design teams and then conducted a workshop series with a professional software design team to use DTM as a *perceptual* tool to test a number of *action-repertoires* and building theory that could be used to improve Design Thinking practice. The result is multi-disciplinary perceptual monitoring of design thinking activity in professional software practice. They are focused on how the design team interacts and what outer influences they are facing during the process.

3.5 Part IV: Design Thinking in Information Technology

The authors of **chapter thirteen** that is entitled “On the Perception, Adoption and Implementation of Design Thinking in the IT Industry” are exploring the social aspects of IT products. The author team of Tilmann Lindberg, Eva Köppen, Ingo Rauth and Christoph Meinel from Hasso Plattner Institute are taking a closer look at the shift in IT development that is gradually focusing more on user-centeredness and the non-technical aspects of design problems. Against this background, design thinking has been discussed and applied as a new design paradigm for IT development. Basing on expert interviews and case study research, they examined what it means to put design thinking into operation in an IT context. It is being explained why design thinking is complementary to traditional IT design paradigms and what issues are involved in the subjects of perceiving, implementing and adopting design thinking in IT development. They are focused on how design thinking can augment the performance of gathering teams (a neglected strategy).

In **chapter fourteen** entitled “Determining the Effect of Tangible Business Process Modeling” by Alexander Lübbe and Mathias Weske from Hasso Plattner Institute, the authors have created a haptic toolkit that people can use to map and discuss their working procedures that is called tangible business process modeling (t.BPM). While in the 1st year, they iterated towards the solution, they have conducted in this year a controlled experiment that compares t.BPM to structured interviews. They found out that people have more fun, learn more, do more reviews and corrections with t.BPM and also, that people take more time to think and talk about their processes. In this chapter, they outline their approach and their research agenda and present the experiment setup and results. They are focused on communication strategy distinctions for hunting versus gathering teams.

In **chapter fifteen** entitled “Applying Design Knowledge to Programming” authors Bastian Steinert and Robert Hirschfeld from Hasso Plattner Institute are scrutinizing the process of programming that involves design. Computational logic is constantly reorganized to keep complexity manageable and provide for current and future coding activities to be feasible. However, design practices have gained less attention in the field of programming, even though decades of research on design have led to a large body of knowledge about theories, methods, and best practices. This chapter reports on first results of the research efforts to transfer and apply design knowledge to programming activities. The research team improved tool support for software developers in two respects, both of which are based on key concepts in design practices: continuous feedback and ease of exploration. They are focused on hunting team behavior in software innovation.

4 In Summary

The term creation is derived from the Latin verb “crescere” which means growing or prospering. And creations, ideas, inventions and innovations can only grow and prosper if they are being constantly nourished and if they have the space to grow.

For that reason, we hope that this book is a starting point for co-creation in regards to a dialogue with our readers. We want you to enjoy the hunt and benefit from the gathering. We invite you to share your insights, impressions and ideas.

We are thankful to all who have contributed to the book. These are not only the authors but also Martin Steinert and Julia von Thienen as well as untold helping hands from friends within the Stanford design and engineering community and the HPI. They all have successfully managed the program and various community building activities and workshops, all of which have contributed considerably to the success of the HPI Stanford Design Thinking Research Program.

We are particularly thankful to Sabine Lang for her work in preparing this book and supporting its editors.

We sincerely hope that you will enjoy and benefit from the content, format and intent of this book. We hope to instigate and contribute to scholarly debates and strongly welcome your feedback. You can contribute directly by submitting papers to the **"Electronic Colloquium on Design Thinking Research" (ecdtr)** which you can find here: http://ecdtr.hpi-web.de.

We invite you to visit this innovative platform of dynamic and rapid scholarly exchange about recent developments in design thinking research and to join the dialog with us.

Part I
Road Maps for Design Thinking

Tele-Board: Follow the Traces of Your Design Process History

Lutz Gericke, Raja Gumienny, and Christoph Meinel*

Abstract Introducing digital tools to creative work settings is challenging; capturing creative work and conveying design ideas to absent team-members is even harder. In this article we show a new way of saving and presenting creative work data that enables users to browse through past design activities. We extended our existing Tele-Board system – previously intended for real-time design work at different locations – with functionalities for time-delayed interaction. The "Tele-Board history browser" is a web-based user interface offering functionality to go back and forth in the timeline of a whiteboard. Additionally, it is possible to view the whiteboard's usage statistics to gain insights on creative work. With our tool we can support design teams in fulfilling their common tasks more efficiently in dispersed teams and we can also assist design researchers to understand how designers work in an all-digital setting.

1 Collaborative Design Across Distance and Time

Design Thinking and creativity methods make use of analog, tangible tools, artifacts and methods [1].

The extensive use of sticky notes, whiteboards, walls, pens, all imaginable handicraft objects, role-play and storytelling is substantial. Bringing together the insights on research and different perspectives of a diverse team are key factors of successful design work.

Being sure to incorporate different cultural aspects as well, such as input sure to incorporate different cultural aspects as well, input from international team members is important.

L. Gericke • R. Gumienny • C. Meinel (✉)
Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900460, 14440 Potsdam, Germany
e-mail: lutz.gericke@hpi.uni-potsdam.de; raja.gumienny@hpi.uni-potsdam.de; meinel@hpi.uni-potsdam.de

*Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_2,

But how can teams reasonably utilize the above-mentioned analog tools if people are geographically dispersed and time zones separate them by several hours?

Can digital equipment support Design Thinking teams when they are not located at the same place?

To answer these questions, we developed the *Tele-Board* system, which provides the possibility to work creatively over distances and all the same retains the feeling and working modes of traditional tools [4]. People can work with whiteboards and sticky notes as they are used to and additionally have the advantage of digital functions that don't exist in the analog world. For remote settings, all whiteboard actions are synchronized automatically and are assigned to every connected partner. To facilitate a really interactive session we included a video-conference between the distributed team: the translucent whiteboard is an overlay on top of the full screen video of the other team members. This setup lets everyone see what the others are doing and where they are pointing. Additionally, you can see their gestures and facial expressions (see Fig. 1).

But as we learned from feedback on this prototype, it is not only important to enable synchronous working modes for distributed design teams, but asynchronous collaborative work as well. To address the problems of Design Thinking teams who are working asynchronously over distances, we developed the *Tele-Board history browser*: a web-based interface that provides the opportunity to go back and forth in the timeline of a whiteboard. It enables designers to view the collected data from different perspectives and thereby gain a deeper understanding of the project context. Additionally, it supports the team to analyze the overall project progress

Fig. 1 Working remotely with the Tele-Board system

and decision paths taken by the respective distributed sub team or by the team itself in an earlier project phase. The team can also continue at any past state by duplicating the whiteboard content, i.e. starting a parallel session. All data is persisted implicitly, meaning that the user has the freedom not to think about saving data. Furthermore, it is possible to view the whiteboard's usage statistic to gain insights on how the designers work. Important areas on the whiteboard (hot spots), time periods with a lot of interaction or different project phases can be detected when analyzing the collected data.

In this article, we describe the general architecture and setup of the Tele-Board system and how it can be used for synchronous work as well as for asynchronous work. We present a novel way of capturing creative work data and thus enabling others to understand the evolution of a design team's work. Furthermore, Tele-Board and its new history function give unlimited possibilities in easily analyzing and evaluating how creative teams manage their work and how innovations arise.

2 Translating Creative Work to the Digital World

The Tele-Board system is an electronic whiteboard software suite, which works like a traditional whiteboard: you can write and draw on the whiteboard surface and – if you are not really satisfied with your work – erase all of your scribbles afterwards. Additionally you can write digital sticky notes: on tablet PCs, an iPad, smartphones or directly on a whiteboard, just as you prefer. You can move the created sticky notes, edit and resize them or group several sticky notes in a cluster. All of the mentioned actions are synchronized automatically and propagated to every connected whiteboard. Every user can manipulate all sticky notes and drawings, no matter who created them. This is a major advantage compared to *Clearboard* [5] and *VideoWhiteboard* [8] where you can only edit your own whiteboard marks.

To facilitate a real interactive session, we included a video-conference feature for distributed team members. The whiteboard can be displayed as a translucent overlay on top of the full screen video of the other team members (see Fig. 2). This setup gives the opportunities to see what the others are doing, where they are pointing and what gestures and facial expressions they are making. The flexible architecture of the Tele-Board system makes it possible to start the whiteboard software on every computer. Thus you can use it with all kinds of pointer input hardware – such as interactive whiteboards, interactive projectors or tablet PCs.

2.1 Projects and Panels

All activities in the Tele-Board software are centered around *projects*. A project can comprise different phases in a design process and can last several months. Applying design thinking methodology in a project often involves a fixed set of analog whiteboards that will be filled with sticky notes and handwriting over

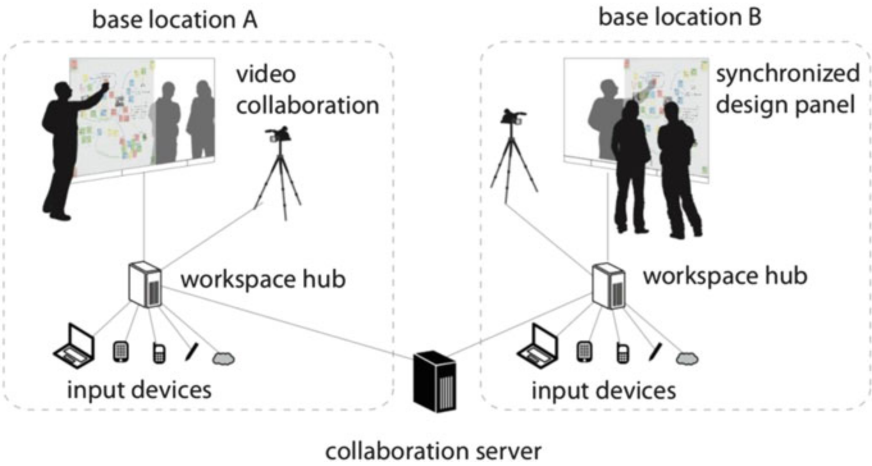


Fig. 2 General setup of the Tele-Board system

several hours or days. Later, these whiteboards will be photo-documented, cleaned, and used for new content. The digital pendants of these physical whiteboards are called *panels* in the Tele-Board system. Panels do not have to be cleaned after being used, but can be archived and restored. Moreover, an unlimited number of empty panels can be requested.

In an ideal setup, panels are viewed and modified with the help of interactive whiteboard hardware, which can be connected to any computer. Decoupling whiteboard hardware and the whiteboard's content adds flexibility, as fewer – potentially only one – electronic whiteboard is needed to replace a traditional setup with analog whiteboards. In addition to direct manipulation of a panel displayed on an electronic whiteboard, Tele-Board allows for indirect user input from different devices, such as mobile phones or laptops, preferably with touch or pen input (Fig. 3).

2.2 Tele-Board Components

The mapping of the Tele-Board data model on different hardware devices is achieved by using the Tele-Board software system, which consists of four components: a *Web application*, a *whiteboard client*, a *sticky note pad*, and a *server component*.

2.2.1 Web Application

The *Web application*[1] serves as the entry point to the Tele-Board system: users can browse and manage projects and associated panels. Here they can also start the

[1]http://tele-board.de/

Fig. 3 Tele-Board and its components (from *top-left* to *bottom-right*): StickyPad on iPod touch, whiteboard interaction, StickyPad on tablet PC, web application, screenshot of the whiteboard client, editing with a laptop computer

whiteboard client and work on the panel's content. Users only need to click the "START" button and the whiteboard client software is started from the browser. It is not necessary to install the software, which makes it easily accessible from any computer.

2.2.2 Whiteboard Client

The Tele-Board *Whiteboard Client* is developed in Java, as we were looking for a platform independent solution. Its main functions comply with standard whiteboard interaction: writing on the whiteboard surface with pens of different colors, erasing, writing sticky notes. Additional functions as panning the whiteboard surface, cut and paste, clustering and deleting elements enhance the working experience.

If no special devices are connected, the client takes the mouse input of the computer. For an optimal performance of interactive whiteboard equipment or tablet PCs we created an abstraction layer that can be the connection with programming interfaces (APIs) of different devices as e.g. the SMARTBoard API or the tablet PC API.

2.2.3 Sticky Note Pad

As an equivalent to paper sticky note pads we created different applications for writing sticky notes. The Java application is ideal for tablet PCs and other pen input devices. For fast finger input you can use the dedicated App on an iPad, iPhone or iPod Touch.

2.2.4 Server Component

The *Server Component* coordinates all communication between the remote partners. All interactions are transferred in the form of XMPP messages to keep the connected whiteboards synchronized. For advanced saving and resuming possibilities we extended the Server Component with additional functions (see Sect. 3 for details).

2.3 Video and a Translucent Whiteboard Surface

Remote collaboration on electronic whiteboards benefits from an accompanying videoconference showing the remote team interacting with their whiteboard. Without video, whiteboard interactions by remote team members appear as if made by a "ghost hand". For the current implementation we decided to use Skype because of its popularity, proven reliability and ease of use. However, Tele-Board can be used with any third-party video conferencing software. Instead of separating video transmission screen areas from whiteboard content, the Tele-Board whiteboard client can act as a translucent overlay on the video conferencing software to give the impression that the remote party is directly interacting with local whiteboard content.

The video cameras should be positioned next to the electronic whiteboards, capturing the foreshortened whiteboard and the people in front of it (see Fig. 4, angular position). Using this configuration, people can face the whiteboard and the camera at the same time. However, this introduces the trade-off that due to the camera angle on the electronic whiteboard the screen area of the whiteboard client is roughly reduced by half. Another possible setup uses a camera position directly in front of the whiteboard, capturing the whole whiteboard surface almost without any skewing and no loss of whiteboard space (see Fig. 4, orthogonal position). The person standing in front of the board is shown from behind. Eye-contact is

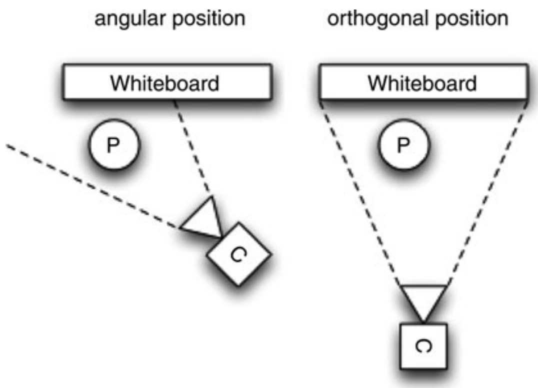


Fig. 4 Different camera setups; *left*: split-screen setup, *right*: full screen setup

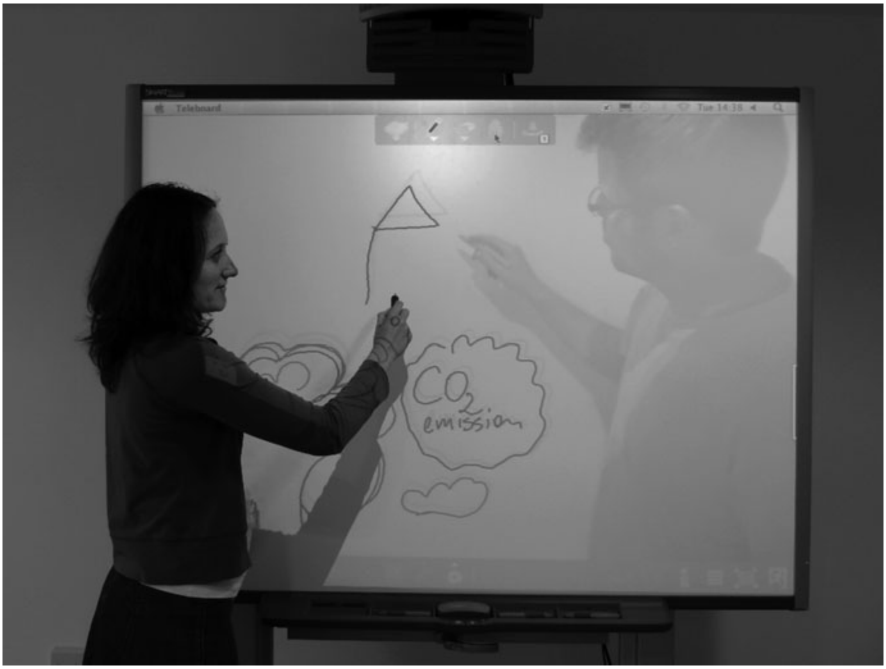


Fig. 5 Full screen video overlay setup (orthogonal position)

limited with this setup, but for the perception of pointing gestures, this setup is ideal (see Fig. 5).

The above mentioned functions of the Tele-Board system enable people – especially designers in a d.school-like environment – to work synchronously in a way they are often used to. However, we learned from user feedback and interview that people in remote teams are working asynchronously most of the time. To support these working modes as well, we developed a solution that helps teams members who cannot be connected at the same time to understand what the others were doing and easily hand over their work. The next section gives a detailed description of the "history" part of the Tele-Board system.

3 Tele-Board History

Many digital whiteboard tools already exist. Most of them are designed for being edited locally, some of them support synchronous settings over distances, but hardly any of these tools is able to support asynchronous working modes. However, the asynchronous scenario is one of the most crucial elements of collaboration within companies. Teams are often distributed over several time zones.

Collaboration starts with understanding and retracing work that has been done by the other team members in order to come to a common understanding within the team. Many solutions offer save and load of a whiteboard state, but only the latest state is kept and no history stays within the system. As one of the few exceptions, Klemmer et al. implemented the possibility to go back to different states and even try out parallel interactions from a certain (decision) point within the whiteboard's timeline [6]. However, the authors point out, it is sometimes problematic to reconstruct certain whiteboard states as their system is based on paper sticky notes (which are partly digitalized) and degenerated states can occur.

Often users bypass this problem by manually creating a file archive and use a conceived numbering scheme to keep track of crucial changes. If a solution kept all changes- which were made for an artifact and the whole document- instantly, this could help understanding what has happened and to better build upon the team members' insights and ideas. This behavior also includes the freedom of not having to worry about explicitly storing and loading the content and thinking about the "right" moment – without saving too frequently or too infrequently.

It is crucial for team interaction to understand what colleagues are doing and when they make certain decisions. Navigation through different whiteboard states and continuation of work at any previous point in time must be easy. A digital whiteboard solution can also offer the possibility of extensive and partly automated documentation. In traditional whiteboard settings it is time-consuming and troublesome to take detailed photographs after work is done. Written documentation for stakeholders and customers has to be prepared additionally. Another argument for the importance of an implicit documentation is the statistical relevance for people researching on teams and how they design over distances and time differences. Various questions could be answered using the history data: What is the main working time of the employees? How can the output become measurable? Not only design researchers could be interested in this information, but also the designers themselves would profit from gaining insights into key factors of their creative work.

3.1 Message Capturing

To address the mentioned challenges and realize the necessary functionality we extended the Tele-Board system with three main functional units:

- Interception of message flow
- Storage of communication data
- Enabling interaction with the history data in an appropriate user interface

The communication data should be captured on the fly, which has influenced our selection of technology insofar as it must be possible to analyze packets separated from the message routing [3]. The message server and its plug-in architecture, the

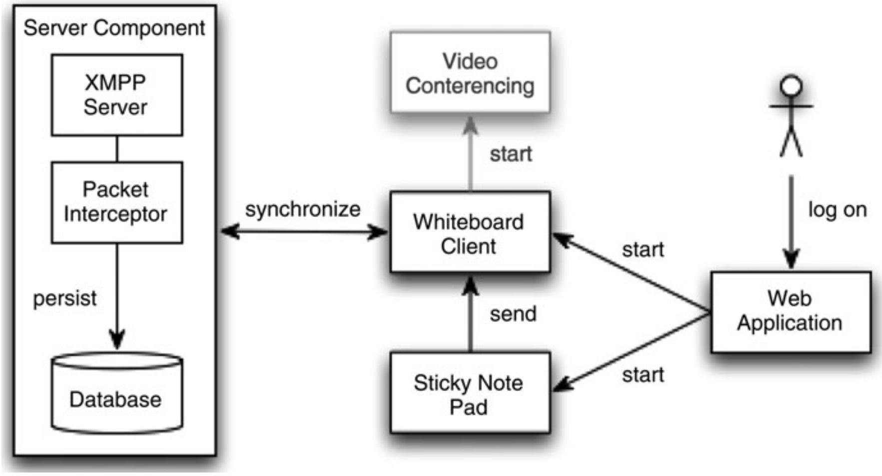


Fig. 6 Conceptual component model of the Tele-Board system

web-based management system, and the database management system represent central roles in the overall system (see Fig. 6).

The history functionality is a concept that is implemented as a crosscutting concern in all parts of the system. It cannot be realized as one single component. A central history archive is used to keep all data together in order to analyze it conveniently and enable asynchronous work. An interceptor was realized as a plug-in within the communication server. This so-called *Server-Buddy* plug-in thereby captures all incoming packets and stores them in a database. This allows the immediate analysis of the communication flow.

Panels and *Projects* in the Tele-Board portal are mapped to the corresponding concepts in the Tele-Board history, which have been described earlier. A Panel describes the sequence of events executed on a whiteboard in the temporal order of these events. Therefore, an event is a set of attributes describing which action has been undertaken and where, by whom, and when. Each event has an operation code, which can be NEW, CHANGE or DELETE to describe the event type. A Project is the collection of multiple Panels that can be configured to require certain access permissions to edit/view/delete.

3.2 User Interaction Points

Basically there are three major groups of users interacting with the developed system: the designer working on the whiteboard content, a manager tracing the design activity of the designers, and a design researcher who seeks to gather insights on how the designers worked. Designers and managers care more about what has happened, design researchers are more interested how the interaction took

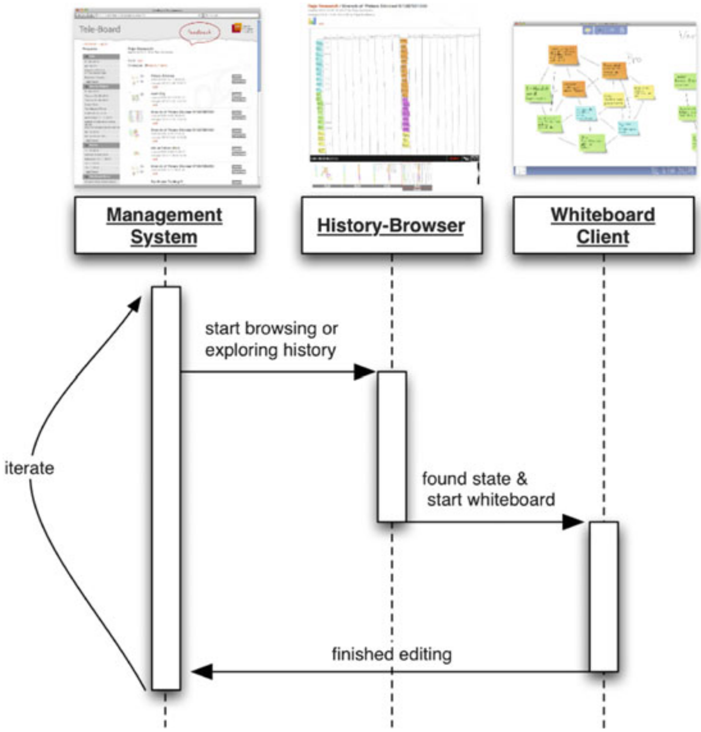


Fig. 7 User interaction points with the Tele-Board components

place. Indeed, there are large overlapping areas between these three activity categories.

Figure 7 shows the components that are interesting for understanding the interaction process. There are three main points, where users can interact with the system. The first one is the web-based management system. The user logs in at the web portal and browses through Projects and Panels in order to get an understanding of past design sessions. Embedded in the web-interface is the History-Browser. It is a user interface enabling people to go back into the history of a Panel. A user can immediately see changes between points in time by switching between the screenshot images of the whiteboard content. By scanning through days, hours, minutes or even seconds, differences can be found immediately. While searching for a certain series of events you can drill-down into the history of the Panel. At each level users only see time intervals when action took place.

The third component is the earlier described Whiteboard Client. This is the main component for synchronous whiteboard use, as design teams will spend a lot of their time creating content with this interface.

Every single component has very special needs in terms of user interface development, data structures, and communication methods. So it was an important decision to use the Extensible Messaging and Presence Protocol (XMPP) as a communication protocol, because a variety of input devices and different platforms should be supported. XMPP is an open standard and is typically used as a chat and instant messaging protocol. Over time it has been extended to support voice, video, and file transfer. Authorization, session and roster handling is managed by the server. People can connect with every possible client without transferring any configuration data from client to client except for username and password.

Technically, all communication is routed over the XMPP server. In terms of XMPP, the whiteboards "chat" with each other. All XMPP-Clients producing text-based sticky notes or other whiteboard content (including whiteboard clients) direct their messages to a chat room, which reflects a specific whiteboard. From there, messages are distributed automatically to all connected whiteboards.

XMPP as a communication language between the clients turned out to be very appropriate. The development of the whiteboard clients can rely on a sophisticated infrastructure e.g. for user handling and message routing, as part of the existing protocol. The chat messages between the Whiteboard Clients contain an XML-encoded text representation of a single whiteboard element.

The History-Browser is the tool, which enables users to browse in the archived whiteboard data. It is possible to go back in the timeline of interaction and reproduce every point in time of a collaborative session. This application is visually part of the portal interface, but logically separated from it. The aim of this application is to show users how a Panel has developed over time. It offers a read only view on the whiteboard content and offers an entry point into existing whiteboard sessions.

The current version of the History-Browser is the result of an iterative development process. The first versions used a time-synchronous approach to display the history data. It turned out to not always meet the user expectations, e.g. when working on a Panel for an hour on one day and a week later for another hour, the timeframe with no interaction at all took most of the screen area in the user interface. We decided to only show those periods with available interaction data to make navigation more convenient and leave out unnecessary parts without losing the information of a longer gap between two sequences. There are several temporal zoom levels for adjusting the amount of detail that is shown. The user can switch between units of days, hours, 10 min, 10 s, and even single seconds.

When the user has navigated to an interesting state, currently two options are offered: to resume or branch a Panel and comment or email an interesting point.

From a technical viewpoint, resuming is only possible at the end of a whiteboard session, because everything that is recorded lies in the past and cannot be modified without changing the ongoing events. In the user interface this limitation does not exist anymore. Users can resume virtually every whiteboard state by branching from any point and resuming the created branch.

The History-Browser is a very valuable tool, when finding certain points in time and retracing activity on a very detailed level. This is especially useful when a user – who was already participating in the design session – browses the history. Besides this interactive History-Browser, we developed an additional application that renders a movie from a series of screenshots. The still images are taken from every point in time when action took place. Thereby, every interaction can be seen in the video. This movie playback is a more passive way of exploring the history towards an overall understanding of an unknown session. The user can later explore the history more detailed and with a better temporal classification of the content by using the History-Browser.

3.3 Statistics

"How do designers work?" is a typical design research question. To be able to answer this question, continuous observation is needed towards a deeper understanding of how designers carry out their activities. Previous approaches such as iLoft [7] or the Design Observatory [2] use elaborated techniques and tools to capture the behavior in the room where the observed designers work. A reported disadvantage is the fact that these observation instruments can lead to distraction of the design activity. By implementing the observation instruments as a part of the tools designers actually use, this distraction will be eliminated and they can even benefit from the digital enhancements, such as resuming existing sessions.

A major drawback of the previously mentioned approaches is that they often only capture an image stream of the interactions. There is only little context information available and large effort has to be spent on the manual analysis of raw material. With an all-digital solution, this process can be automated for the most part. It also offers the possibility for immediate feedback loops because analyses take less time. This can lead to faster iteration cycles for experiment setups and better results in the end.

Designers can also benefit from statistics of their work. With a distant view to past work, they can replay what has happened and come to further insights for their future work. It also enables the participants in asynchronous settings to better evaluate their personal contribution to the design task and also better value the work of colleagues (Fig. 8).

First analysis of the history data visually revealed structural connections between certain Panels. To fulfill strong statistical criteria, the categorization algorithms have to be refined. Preliminary results therefore outline what kind of

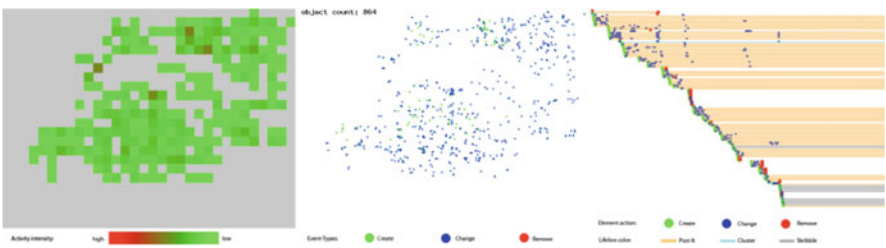


Fig. 8 Example analyses of the history data on different parts of the system: whiteboard usage areas (*left*), whiteboard events (*middle*), whiteboard elements (*right*)

statistics will be possible in the future. The data can be used for several applications such as an overall search function or recommendations within the portal. By analyzing the processes, important points of the whiteboard content could be made easily accessible in order to highlight the most important points. The definition and extraction of outstanding phases in the sessions could also lead to a compact report for an asynchronously working colleague by automatically giving hints on which points in time would be more valuable than others.

There are different perspectives on how the history can be analyzed; three of them are shown in the following.

3.3.1 Spatial Analysis

One of the most intuitive ways of analyzing the whiteboard content distribution is a map of the content. Hotspot analysis of activity on the whiteboard surface shows high activity regions, e.g. caused by creating and moving sticky notes or the creation of sketches. Another aspect of interaction that can be evaluated with our system is the panning interaction on the whiteboard. Patterns in whiteboard content distribution lead to a deeper understanding of how users use the panning capabilities and affect future developments of the software.

3.3.2 Temporal Analysis

The temporal activity distribution is also a very important dimension of analysis. It can give insights about the project lifecycle and when participants were active. The working time distribution can be analyzed – not only in general but also for every participant separately. Profiles can identify personal preferences.

3.3.3 Key Figures

Besides the mentioned visual ways of analyzing the whiteboard content, there can be a variety of other key figures describing the interaction process. One example is reasoning on resizing actions of sticky notes by counting actions to enlarge or shrink. From these numbers an argumentation can lead to optimized sticky note default sizes for different tasks. Very important is also to study the contribution of activities by each team member in a distributed setting. Significant differences can reveal certain team characteristics and their influence on team performance.

To sum up, the history can be a very efficient tool leading to a deeper understanding of people interacting with the Tele-Board system. Results can be made traceable and the implications for team performance become measurable in a way that the history infrastructure can be a valuable tool for design researchers as well as designers.

We are also elaborating on structural analysis of the history data in order to provide search applications or make meaningful recommendations to the user. The key challenge of this automatic learning from history data is how to find important points in time. Therefore we have to ask the question: What makes an episode in the history important for the person who wants to understand past design activity? Ongoing user observations and tests will give us more insights on this problem domain.

4 Outlook and Future Work

In this article we presented the possibilities and advantages of the Tele-Board system which supports creative work in synchronous and asynchronous settings. Tele-Board automatically captures whiteboard interactions and offers a history view to the minute. Users can browse through whiteboard screenshots at different time levels or generate a video of the history data. Thereby, we reduce the amount of time users need to view the whiteboard interaction as they do not need to watch several hours of standard video. Still, a lot of data is generated while working for hours, days or even months. For team members who want to understand what their partners did, it would be an enormous help if the system found important phases during their work. In the future, Tele-board should recognize the most important phases and present them in the history browser. A prerequisite for suggesting important points of a design session history is an analysis of the stored history data and identifying situations with high information value. Such moments can be, for example, when a team came to certain decisions or had seminal ideas. During the past project year we collected several hours of test data, which will be the basis for our research. The obvious commonalities in these processes can be transferred to the computational analysis. Usage pattern analysis is an approach that can lead towards a computational understanding of the interaction processes. Indicators for these usage patterns are: transitions between working modes, writing/sketching/clustering phases, count of new sticky notes and scribbles etc. In summary, user-generated feedback can be even more valuable and an indication for these reasoning methods. Users can comment on the history of a design process, but also during the sessions they can give direct feedback with certain devices, e.g. a multi-button buzzer.

Furthermore, we will evaluate the importance of additional audio or video recordings. We want to know if it is sufficient to only view whiteboard interactions of important points in time or if it makes a significant difference to add the related audio or video recordings as well.

Another feature that can support design teams during their work is the simplified generation of documentation. Designers could select history artifacts that would be directly inserted into text documents, presentations or any other kind of documentation. As an analysis of this automatically derived information, we want to find out how helpful this semi-automatic documentation can be.

4.1 Evaluating Our Ideas and Designs

In the upcoming year we also plan to set up the Tele-Board system in a team's design space for a longer time period to investigate whether or not people would utilize the system and how satisfied they are with its use. Students of the ME310[2] course will use it for working in distributed setups. We want to identify factors and project phases where the system is helpful and in which situations people prefer other tools and why. This will give us valuable research data on usage behavior and the practical and social influences of the system.

Overall we want to determine that a digital system can be used as comfortably and intuitively as traditional tools for creative work settings and, furthermore, even augment the analog experience. The added value of a whiteboard history, highlighting of important project phases and automatic documentation can make the Tele-Board an essential tool for geographically distributed teams.

References

1. Brown, T. Design Thinking. *Harvard Business Review*, June (2008), 84–92.
2. Carrizosa, K., Eris, O., Milne, A., and Mabogunje, A. Building the Design Observatory: a Core Instrument for Design Research. *International Design Conference*, (2002).
3. Gericke, L., Gumienny, R., and Meinel, C. Message Capturing as a Paradigm for Asynchronous Digital Whiteboard Interaction. *6th International ICST Conference on Collaborative Computing: Networking, Applications and Worksharing*, (2010).
4. Gumienny, R., Meinel, C., Gericke, L., Quasthoff, M., LoBue, P., and Willems, C. Tele-Board: Enabling Efficient Collaboration In Digital Design Spaces Across Time And Distance. In H. Plattner, C. Meinel and L. Leifer, eds., *Design Thinking, Understand - Improve - Apply*. 2011.
5. Ishii, H. and Kobayashi, M. ClearBoard: a seamless medium for shared drawing and conversation with eye contact. *Proc. CHI 1992*, ACM New York, 525–532.
6. Klemmer, S.R., Everitt, K., and Landay, J. Integrating Physical and Digital Interactions on Walls for Fluid Design Collaboration. *Human-Computer Interaction 23*, 2 (2008), 138–213.
7. Milne, A. and Winograd, T. The iLoft project: A technologically advanced collaborative design workspace as research instrument. *Proceedings of the 13th Annual Intenational Conference on Engineering Design (ICED03)*, Citeseer (2003), 1–10.
8. Tang, J.C. and Minneman, S. VideoWhiteboard: video shadows to support remote collaboration. *Proc. CHI 1991*, ACM New York, 315–322.

[2] http://me310.stanford.edu/

Understanding Radical Breaks

Jonathan Edelman, Avantika Agarwal, Cole Paterson, Sophia Mark, and Larry Leifer*

Abstract In this chapter, the authors propose an empirically supported framework for understanding how small horizontally organized design teams perform radical redesigns or radical breaks. The notion of radical breaks captures what is often thought of as "thinking out of the box", and reframing problems to find new and unique solutions. A radical break occurs in the course of a redesign when designers make a major departure from the provided artifact.

We introduce three imbricated concepts as a mechanism for understanding how design process determines design outcomes: scoping (what designers take to be the task), behaviors (how designers move through the task), and shared media (drawings, prototypes and gestures). Results of an experiment using small design teams in a redesign task suggests six primary modes of "scoping", two primary modes of design "behavior", and two primary modes of "shared media".

Motivation
The motivation for this work is two-fold.
The first is to craft a coherent theoretical approach to design process and it's relationship to how radical breaks are produced.
The second motivation is in service of the practitioner. Rather than a top-down model of process, it is our desire to craft a framework that makes sense to practitioners. Our aim is to provide a clear understanding of how process "works" for designers, so they can make informed choices about how to approach redesign tasks "in the wild".

J. Edelman • A. Agarwal • C. Paterson • S. Mark • L. Leifer (✉)
Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA
e-mail: edelman2@stanford.edu; avantika@stanford.edu; colepatt@stanford.edu; samark@stanford.edu; leifer@cdr.stanford.edu

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_3,

1 Radical Breaks

The notion of radical breaks captures what is often thought of as "thinking out of the box", and reframing problems to find new and unique solutions. A radical break occurs in the course of a redesign when designers make a major departure from the provided artifact.

1.1 Redesign

A major and by no means trivial challenge for designers is redesign. Redesign tasks provide a good platform for understanding radical breaks as they permit an evaluation of the differences in the model provided as a starting point and the newly designed model as an end point. In addition, redesign is a prevalent design activity in professional design practice.

1.2 Incremental Redesign

The task of redesign is often a matter of making incremental improvement to an existing design. The process of optimization and incremental improvement is practiced with rigor, and is a remarkable achievement for both industry and the academy.

1.3 Radical Redesign

At other times, the task of redesign is to make a radical break from the existing design. This process is less well understood than incremental improvement. The object of our work is to understand the dynamics of radical breaks in the redesign process, particularly in the context of designers working in small, horizontally organized teams. It is our hope that insights gleaned from this chapter will be applied to a broader range of design contexts.

1.4 Research Question

Our study was designed to observe how small teams make radical breaks. The original question we asked was, "What are the primary elements radical breaks?" This question evolved as we examined the data, and over the course of this study the original question gained precision and can be expressed in the form, “How do scoping, behaviors, and shared media interact and influence outcomes of a redesign task?”

1.5 Data Sample

Several factors went into developing our data set. We wanted to do a close analysis of how designers perform radical breaks. Video recording of small teams at work promised to be a good approach, though the problem of how to capture radical breaks meant we had to set up proper conditions for them to occur.

Our notion of radical break entails making a significant departure from an existing solution. The genesis of this study began with an investigation into how different kinds of media could trigger a radical break. This meant that we had to provide teams with media from which they could break away, and media that would possibly trigger a radical break. We assumed that we would have to cast a wide net in order to get a couple of examples of radical breaks.

We created this data set in order to examine the phenomena with a mixed methods approach. We video recorded 14 teams in a 30 min redesign task, constructed to tease out radical breaks. We fashioned the redesign task and stimuli in order to create cases primarily for qualitative case study observation, rather than to focus on quantitative analysis, as the object of this study was to develop theory about how radical breaks are accomplished, rather than perform hypothesis testing. We believe that qualitative insights harvested from our data will best serve designers in search of theoretically informed frameworks to apply to their practice.

2 Methods

2.1 Subjects

Subjects included a wide range of design engineers in the domains of product design and mechanical engineering: undergraduates, graduate students, professional product designers, professional engineers, and educators in both product design and engineering design.

Subjects were grouped into non-hierarchical teams of three. With one exception, subjects on a team were familiar with one another, either having worked together, or had socialized on many occasions. Our intent with putting together teams was to attempt to ensure that the subjects would be comfortable with each other during the redesign task.

2.2 Procedures

The idea behind the task was to fixate the subjects on an engineering drawing for 5 min and then provide an additional stimulus in order to see if it had an effect on the subjects interactions.

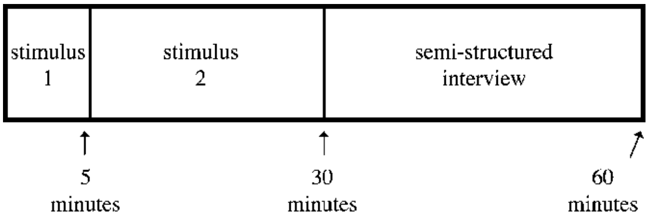
The procedure unfolded as follows:

Three subjects were brought into the Design Observatory at the Center for Design Research at Stanford. The room into which they were brought had a table set up in the center of the room and four HD video cameras focused on the table: one for each of the three team members, and one looking down at the table from above.

Subjects were asked to fill out a release and then given a brief biographical questionnaire.

Next, subjects were given a prompt and a stimulus. After 5 min, a new stimulus was place on the table.

At the end of a total of 30 min, subjects were engaged in a semi structured interview, which lasted between 20 and 30 min.



2.3 Stimuli

All subjects were provided with a printed prompt and toleranced, labeled engineering drawing of a device that could reportedly analyze the properties of materials.

2.3.1 Prompt

The prompt states "The object in front of you allows the identification of material objects. Redesign it." Because we were interested in observing the effect of the media we provided, we decided to keep the prompt as brief and open as possible. The notion of "redesign" would have to be determined by each team.

2.3.2 Engineering Drawing

The engineering drawing offered six views of the object: front, back, two sides, top, and an orthographic view. The former five views were wire frames, while the sixth, orthographic view was generically surfaced. The major features of the device were labeled with call outs (e.g. “display”, “focus”, “zoom”, “sensor”, and “start”) (Fig. 1).

The two control groups were given the engineering drawing alone. The 12 other groups were given an additional model after 5 min.

2.3.3 Secondary Stimuli

These models included a low resolution ID-style sketch (Fig. 2), a rough cardboard puck with features drawn on it with sharpie (Fig. 3), a rough cardboard experience prototype (Fig. 4), and a well rendered foam model (Fig. 5).

Stimuli were chosen through use of the Media-Models framework [4, 5, 8]. The Media-Models framework was developed in order to understand the kinds of media used in diverse design environments. Briefly stated, the Media-Models framework describes characteristics of the media that designers use in various stages of development. The framework utilizes three dimensions: abstraction, definition (resolution), and mutability. Development stage appropriate media can be enlisted for development based on these dimensions.

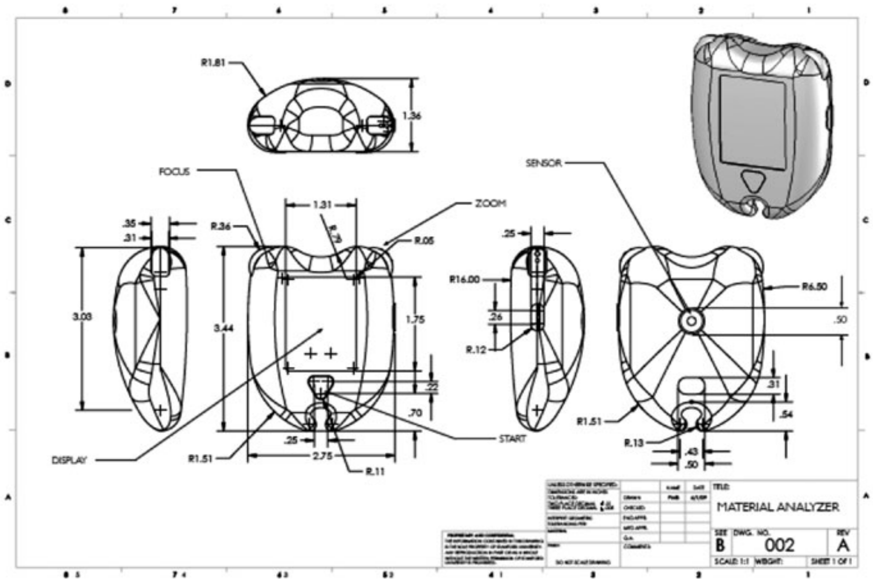


Fig. 1 First stimulus, material analyzer

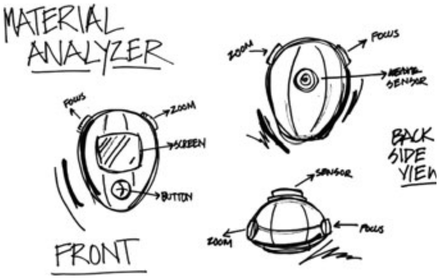


Fig. 2 ID sketch

Fig. 3 Cardboard puck

Fig. 4 Low resolution experience prototype

Fig. 5 Well rendered foam model

3 Analysis and Discussion

Results are based on analysis of video stream and on self-reporting by the teams during the semi-structured interview. Our interest is in linking the deliverable with scoping, behaviors, and shared media.

3.1 Deliverable

Assessment of each team's deliverable, the redesigned object, was made through analysis of video streams and through participants in responses to interview question Q1, "How did you redesign the product, what does your new model look like, and how does it work?"

3.2 Process

Assessment of each team's process, how they did their work, was made through analysis of video streams and through semi-structured interviews. The four separate video streams were combined using Apple Final Cut Pro, in order for us to see all four streams at the same time. Transcripts were made from the audio stream. The transcripts and video stream were then placed into the Nvivo analytic tool, and linked to one another.

Process is considered in the context of three imbricated elements: scope, behaviors, and shared media.

3.2.1 Scoping – Behaviors – Shared Media Triad

Our research suggests that three imbricated elements, "scoping," "behaviors," and "shared media" form a triad that determine design outcomes in an experimental setting.

3.2.2 Introduction to Scope

Scope is a distinction based on video stream analysis of the data set and a reformulation of classical engineering methods. By scope we mean how the problem/solution space is framed. Six classes of scoping were observed used alone or in combination: systems, use case scenarios, usability, features, functions, and core functionality.

3.2.3 Introduction to Behaviors

Our approach to observing and analyzing behaviors is based on contemporary work in sociology and behavioral science. We wanted to see if we could observe two classes of behavior, "wayfinding" and "navigation", in the actions of the experimental subjects.

3.2.4 Introduction to Shared Media

Part of the rationale for creating these cases was to observe how shared media contributed to making radical breaks. Shared media is exactly what it sounds like: the media that teams share and co-explore in the course of designing. Our questions about shared media include, "Do certain kinds of media promote radical breaks?", and, "Do certain kinds of media promote optimization?"

While we found that the influence of media was a nuanced affair, we also found that media does have a palpable effect on the kind of work designers do. Not only did the media we provided to subjects have and effect, but the media they produced characterized what they did and how they did it. We found two types of shared media in our data set, "scaffolds" and "anchors".

3.3 Scope

Scope refers to the dimensions with which solutions are framed. We appropriate two notions from developmental psychology in order to understand how scoping works in action, "counterfactuals" and "paracosmos".

3.4 Counterfactuals and Paracosmos

In her book, *The Philosophical Baby: What Children's Minds Tell Us About Truth, Love and the Meaning of Life*, Alison Gopnik [7] describes how children imaginations work. When children imagine things that don't exist, they are termed "counterfactuals". The production of counterfactuals are quite common to children. Gopnik notes that no matter how far flung a counterfactual is (for example an eccentric imaginary friend), counterfactuals have a logical coherence in respect to the contextual world from which they arise. This contextual world Gopnik calls a "paracosmos". In short, counterfactuals always exist within the context of an imaginary world which has rules of it's own.

3.4.1 Counterfactuals and Paracosmos in Redesign

Our research suggests that design engineers engaged in a redesign task follows Gopnik's structure. New solutions are, needless to say, "counterfactuals"; they stand in contrast to what exists, as exemplified by the object to be redesigned. The "world" in which they arise can be seen as a "paracosmos"; generated by design engineers to justify, support, and develop new ideas. The extent to which the paracosmos is unpacked, or developed determines the refinement of the idea.

3.4.2 Scope

In design scenarios we have observed the construction of paracosmoses and counterfactuals. In the case of redesign scenarios we have found it useful to understand them under the rubric of "scope".

Scope is the means by which the dimensionality of a paracosmos is defined, on the one hand, and on the other hand, what aspects of a counterfactual are taken up for development. Scope determines on which level or levels the object is approached and developed.

Six classes of scoping were observed used alone or in combination: systems, use case scenarios, usability, features, functions, and core functionality.

Projective Scoping

The first three classes (systems, use case scenarios, and usability) we call projective scoping, because these are contexts into which the original design is imagined, or projected, and thus are primary factors in shaping the paracosmos.

Extractive Scoping

The second three classes (features, functions, and core functionality) we call extractive scoping, because these are elements of the original design that are pulled out, or extracted, from the stimulus. These classes determine at what level the redesign of the object occurs.

One example of an interchange illustrates how the logic of the paracosmos shapes a counterfactual and visa versa. In this case, the team (D, K, and J) discusses possible use case scenarios for a material analyzer and it's possible functions:

K: You can take it camping and figure out if you can eat things.
[paracosmos: use case scenario]
D: Yes, camping.
K: Yes.

D: Camping.
K: There were so many times I wished I could do that. The little berries on the side of the road.
J: Is this good to eat?
K: Beep, edible or not?
[counterfactual: functionality]
J: Yay!
J: Oooh, water, it could tell you what water you could drink.
[counterfactual: functionality]
K: Yes. It could analyze the materials of the waters.
J: This would be really great for someone going to a different planet.
[paracosmos: use case scenario]
K: Uh huh.
D: Yes.
K: Yes. Spacemen."

This exchange shows how camping (paracosmos, use case scenario) leads to the detection what is consumable (counterfactual, functionality), which in turn leads to analyzing the potability of water (counterfactual, functionality), which then leads to someone going to another planet (paracosmos, use case scenario).

A bit later, the team returns to the outer space scenario and explores how the analyzer would be operated:

J: If you're on another planet you probably have one some sort of a space suit.
[paracosmos: use case scenario]
K: So, it needs to have big buttons because those fingers…
[counterfactual: features]
J: Or integrate it into what you're wearing.
[counterfactual: function]
K: Yes. Or, you talk to it.
[counterfactual: function]
D: Exactly.
J: Because every time you look somewhere…
[paracosmos: use case scenario]
K: Because you could just open your visor when you're in space and talk to it. (Laughter)

Here, the projective scoping of being on another planet (paracosmos, use case scenario) requires that space suits must be worn, which means that gloves will require "big buttons" (counterfactual, features). Big buttons are not the only way of responding to the space suit, and the team investigates either integrating it in the suit (counterfactual, function) or talking to it (counterfactual, function). These consideration give rise to a new scenario (paracosmos, use case scenario) in which the new functionality could exist, looking and speaking while wearing a space suit.

3.5 Behaviors: Wayfinding and Navigation

We begin our discussion of behaviors with a look at “wayfinding” and “navigation”, two methods of making our way through physical terrain proposed by contemporary social scientists. Our interest is in the nature of wayfinding and navigation in the context of engineering design. Some of the questions we asked ourselves were, “Can we see the wayfinding and navigation at work in small teams?” and, “Do these behaviors determine or effect the outcomes of a redesign task?”

These distinctions form a basis for understanding the behaviors of design engineers at work, and will later be complemented by the two analogous distinctions made in the domain of media, scaffolds and anchors.

In his book “Lines a Brief History,” Tim Ingold [13] suggests two methods for making one’s way in the world, wayfinding and navigation, which he illustrates with descriptions of how the Orochon people hunt wild reindeer, riding saddled domesticated reindeer. In this passage Ingold uses the word, “wayfaring” in place of “wayfinding” and “transport” in place of “navigation” [12], though the sense of both is consistent with the use in this chapter.

> The path of the saddle-back rider, according to anthropologist Heonik Kwon, is ‘visceral in shape, full of sharp turns and detours’. As they go on their way, hunters are ever attentive to the landscape that unfolds along the path, and to its living animal inhabitants. Here and there, animals may be killed. But every kill is left where it lies, to be retrieved later, while the path itself meanders on, eventually winding back at camp. When however the hunter subsequently goes to collect his kill, he drives his sledge directly to the site where the carcass has been cached. The sledge path, Kwon reports, ‘is approximately a straight line, the shortest distance between the camp and the destination’. Not only is the sledge path clearly distinguished from the saddle path: the two paths depart from opposite sides of the camp and never intersect. It is along the saddle path that life is lived: it has no beginning or ending but carries on indefinitely. This path is a line of wayfaring. The sledge path, by contrast, is a line of transport. It has a starting point and an end point, and connects the two. On the sledge the body of the dead animal is carried from one site, where it was killed, to another, where it will be distributed and consumed. [13]

Later, Ingold emphasizes that the activity of wayfinding is characterized by being perceptually attentive to the terrain as it appears in time:

> . . .the traveller’s movement — his orientation and pace — is continually responsive to his perceptual monitoring of the environment that is revealed along the way. He watches, listens and feels as he goes, his entire being alert to the countless cues that, at every moment, prompt the slightest adjustments to his bearing. [13]

Thus, wayfinding involves responding to immediate perceptual cues and information given by the environment. While the goal of the wayfinder may be predefined, the path is determined in the moment and in response to direct perceptual cues that are detected in the environment. Furthermore, while Orochon hunters do have a clear goal their meandering path is critical to reaching their goal. When Orochon hunters recount the hunt, they dwell on unusual things they saw or heard, not on the kill. The paths of wayfinders are often guided by narrative, and, like stories, they often meander, weave in and out, often turning back on themselves.

In, contrast, the practices of navigation are linear. Orochon hunters wrap the kill, leave it in place, and head directly back to their village. They return post haste on sleds, in the most direct route possible, and bring the carcass to the village along the most efficient route.

While wayfaring relies on in situ response to direct perceptual cues, often within the context of a narrative, navigation relies on a top down approach of charts and instruments to determine the most direct route, often a straight line. Gone are the narratives which play out in time. Instead, triangulation and charts which reduce a myriad of perceptual input to thin lines on a grid [10, 11]. The maps used by navigators depict the terrain and the predetermined route all at once.

The practices of navigation, Ingold tells us, are well suited for keeping both the navigators and their payload intact and unchanged, not unlike the practices of optimization in engineering design. Wayfarers, on the other hand, experience change during the course of their journey and what they carry may change, not unlike what happens when engineering designers perform radical breaks.

While wayfinding can be considered as determining one's path in response to direct contact with the environment, often accompanied by a narrative, navigation involves setting a predetermined route, the shortest possible path, often a straight line. Navigators rely on charts and maps, which provide a top-down view of all things at the same time. While navigators do look at the landscape, it is always in reference to maps, charts, and instruments like compasses, which tell navigators where they are [10, 11].

Ingold's description of wayfaring and navigation are well suited for describing the behaviors observed in design engineers when they are engaged in redesign exercises. What follows is an outstanding example of wayfinding behavior from one team of design engineers (A, B, and M). Rather than a tope down approach of determining what is wrong with the product and following a program for fixing it, these designers *feel* their way through, employing gesture to "feel" the object and the environment as the scenario unfolds:

B: Yes, how about some kind of a glove thing. 'Cause you want to touch a surface, right? You learn a lot by feeling . . .

[Subject B reaches out and feels the surface of the table]

A: Oh yea, the feeling it.

[Subject A reaches out and feels the surface of the table]

B: . . .the friction.

M: Oh, oh, nice. Nice.

B: You got multi-sensors for friction for hardness, for whatever. Then you've got the big sensor that collects the optical data.

[Subject B feels the surface of the table with individual fingers, subject A follows suit]

You just rub things. . .

[Subject B waves hand in the air, "rubbing" imaginary surfaces]

. . .like flesh.

[Subject B turns to subject A and rubs the flesh of subject A's arm]

A: What if you want to touch things that are far away or scary? Then you could have… I'm looking at this and thinking if it's touching wireless then you can throw it over there. Like it's on a string.
[Subject A makes a throwing motion with both hands, away from body]
B: Yea.
[Subject B copies the gesture with two hands, and then one hand. Subject B moves fingers in a carefully articulated fashion]
That also gives you gesture.
M: Ooooh, I like that.
B: That gives control.
[Subject B turns hand over and makes a new gesture]
B: Like you go, give me that. Give me that.
[Subject B points with index finger]
M: So, what's cool is you can do a whole series of gestures.
[Subject M points with all fingers and gestures with hand]
You can point
[Subject M points with index finger alone]
and then you have a display.
[Subject M touches wrist and back of hand]
So, this would be your display here.
[Subject M sketches the display on a glove]
So, let's say display is here.
A: Right.
M: Maybe a couple buttons.
[Subject M touches wrist with index finger several times, indicating button use and placement]
Then you have the different fingers. I love the ability to touch or point.
[Subject M sketches the glove]
Then this is … you're using your finger gestures to define functionality,
[Subject M touches table top with index finger]
which comes up on the display. So, glove…
B: Bracelet…
M: I love that. Oh, bracelet is interesting.
[Subject M points to wrist and then follows hand to index finger]
What about a bracelet with just a few connectors so it doesn't give you a whole glove. Cool. And a little ring thing.
[Subject M wraps his index finger around another finger, indicating a ring]
So, then it almost becomes a jewelry thing.

Nearly every move the team makes is in the context of feeling the real or imagined environment. The concept of the glove co-develops with the gestures, and eventually becomes a bracelet.

In contrast, teams who exhibit navigation behaviors limit gestures to pointing to the engineering drawing in front of them. The path of development is the result of creating a bullet point plan based on fixing what is wrong with the product in the engineering drawing.

Nearly all teams displayed a mixture of both behaviors. The behaviors of those teams who performed radical breaks were heavily weighted on the wayfinding side, while the behaviors of teams who optimized were heavily weighted on the navigation side.

3.6 Shared Media

How does media work in design? How is media related to other aspects of design? What is the impact of media on design teams?

These questions are often touched upon by design theorists and practitioners. While these writers have provided insights into the use of media in design, they have thus far have not provided a comprehensive framework that can be used by designers to make informed choices about the most effective kinds of media for the phase of design in which they are engaged.

The rich and growing corpus of design theory and methodology has begun to show interest in examining the media that design engineers use in their work. Much of the literature addresses formal models and optimization. A smaller body of work examines prototyping and tangibility.

These sources range from brief dictums about the roughness and refinement of sketches [6], to descriptive lists of characteristics of prototypes [3], to rules of thumb in situated design [20, 21], to lists of cognitive dimensions of types of prototypes [1, 2].

While all these approaches warrant merit, none offer a coherent framework for looking at media in the context of practice. In addition, there is little work done on the body of behaviors associated to the media. What are the mechanics of this movement? What are the mechanics of the use of the refinement of sketches?

In our analysis we consider both the effect of the media we introduced as stimuli, and the role played by media the team produced.

In the next sections, we build a functional description of how media works in action, and suggest the distinctions of "scaffolds" and "anchors" in media. Briefly stated, scaffolds serve as a vehicle for gaining insight while envisioning morphing possibilities. Scaffolds tend to be ambiguous models, which are clearly not the final object. Anchors focus the redesign on the object at hand. Anchors tend to be geometric/analytic models, or models that exhibit precision and high fidelity to the final object.

3.6.1 C-K Theory: Logical Status

Our notion of scaffolds and anchors is based in part on the work of Hatchuel and Weil [9]. For Hatchuel and Weil, some concepts have logical status and some lack logical status. Those concepts having logical status are those that fall under the rubric of true and false, either they exist in the world or not. For Hatchuel and Weil, those concepts that lack logical status are combinations of existing concepts, but have yet to be viably realized in the world. The notion that something is possible or impossible does not come into play.

Our interest in the notion of logical status and lacking logical status focuses on discovering if and how media can exhibit logical status or not, or how they elicit responses that indicate logical status or non-logical status. Indeed, we have found that media which claims logical status supports optimization, while media which makes no claim to logical status supports radical breaks.

3.6.2 The Grammar and Agency of Maps

We now consider how media has been taken up by leading archeological theorists. Insights drawn from these thinkers will form a basis for understanding of how media operates in design. In order to understand the characteristics of media that has logical status, we will first consider the grammar and agency of maps.

In his paper "Mediational Techniques and Conceptual Frameworks in Archaeology" [25], Timothy Webmoor examines how the production and use of maps conditions and restricts the thought process of the map reader. Maps, Webmoor argues, are not neutral. They are media, and as such they carry a "message". The map, according to Webmoor, is "a fundamental conceptual framework that archaeologists utilize in directing their methods and formulating interpretations" and that these frameworks "predispose certain interpretations." In Webmoor's view, the word "map" is considered to include, "any spatial representation conveying visual information in a strictly coordinate, graphical manner."

The purpose of mapping, Webmoor explains, "revolves around the identification of boundaries". These lined boundaries, like the lines of maps and engineering drawings, characterize "knowledge". This type of media facilitates the portrayal of all surfaces as abstracted and mathematized. "Mathema" (the Ancient Greek root of mathematics) is often translated as "knowing", a knowing that is associated with the contemplation of unchanging, immutable truths.

Webmoor speaks of how the grammar of the map changes the nature of the phenomena it represents: "The detail of the cartographic map can often, however, elide the very feature – or for that matter, an urban or built architectural space – that it presumes to envisage in its visual conventions. . .Furthermore, maps by virtue of their 'univocal scientific strategy' flatten sensory data into the restricted medium of articulated lines and create 'gaps' and 'blank spaces'".

Maps claim a truth of their own. They have an authority by virtue of the coordinate grid upon which illustrated boundaries are placed. The grid and the boundary are the stock and trade of the grammar of truth-making.

Webmoor's exploration elucidates Ingold's concept of "Wayfinding and Navigating" [12, 13]. The methodologies and tools of "Wayfinding" and "Navigating" suggest two paths similar to the methods and paths of design engineers. Wayfinding occurs when one walks through a landscape without a map and *relies on primary sensory data* to move through a territory. It is direct, specific, and immediate. Navigation, on the other hand, *requires a map*. One determines where one is through consulting the map, the compass and comparing it to what one sees. In this respect, navigation involves triangulation.

3.6.3 Media and Knowing

Maps, like charts and equations, are how we "know" in contemporary terms. In "Thinking with Eyes and Hands" [17], Bruno Latour examines the modern way of knowing in respect to embodiments of knowledge in common use. "We can hardly think of what it is like to know something without indexes, bibliographies, dictionaries, papers with references, tables, columns, photographs, peaks, spots, and bands".

Latour's characterization of the media of knowing is based on his observations of scientists at work [18, 19]. The media used by design engineers to support optimization and incremental change, often in the context of manufacture, have the self-same characteristics. Engineering drawings fundamentally exhibit the same form as maps. Optimization engages charts and equations in the process of literally mathematizing processes and materials for robust and predictable outcomes. When optimizing, engineers *know* their products in the same way that scientists *know* phenomena. This is aided and abetted though the use of the specific kinds of media that Latour traces in his examination of how knowledge is made.

3.6.4 Media in Cognitive Science

Contemporary studies in human cognition suggest that thinking goes beyond the classic notions of boundaries of thinking and the mind as promoted by Descartes and his followers. Contemporary researchers in human behavioral and cognitive science seek to fill in the gap between mind and object through examination of how different kinds of media influence thinking.

Barbara Tversky has made a very strong case for the effect of different kinds of thinking that is elicited when experimental subjects produce different kinds of maps. In short, Tverksy found that rough sketches accompanied what she calls "sketchy thinking". More formal maps indicated more formal thinking [22–24].

Tversky found that architects leverage rough sketches to create "new knowledge" and new scenarios in a manner that formal architectural drawings did not readily support. Significantly, the interpretation of rough, ambiguous sketches was a skill that improved with experience and practice.

3.6.5 Scaffolds and Anchors

These considerations lead us to posit two aspects of media: "scaffolds" and "anchors". These distinctions are based on situated cognitive and behavioral distinctions proposed by Kirsh [14–16] and Hutchins [10, 11]. For Hutchins and Kirsh, thought is often grounded in physical representations, to which they refer as "anchors". The anchor provides a place from which to build or understand new ideas. Kirsh speaks of scaffolds in respect to creating external structures connected to thoughts that allows the mental projection of more detailed structures than could be held in the mind. This affords driving thought further than unaided thinking alone.

In our study, we found that some media served as anchors, though we observed that at times the anchoring was so strong, that it precluded exploring alternatives. Other types of media seemed to invite letting go of an idea, and entertaining new idea. These media we call scaffolds. Scaffolds aided designers to find not only greater depth in their designs, as Kirsh observes, but scaffolds help build designs that are distant relations to the scaffold.

Scaffolds

Scaffolds make no claim to being the exclusive and final thing. They serve as a vehicle for building insight, for building other things. Scaffolds in construction the construction trade are explicitly not the edifice itself, but allow the construction of the edifice. They may in some way resemble the shape of the edifice, but are not usually confused with the edifice. Media scaffolds serve as vehicles for designing, they are not the finished product.

Media scaffolds often function as "metaphors", and are treated as metaphors. Experimental subjects who make radical breaks speak of "the mouse metaphor" or "the point and shoot metaphor". In this respect media scaffolds are abstractions, because they give the exploration direction and form under a single rubric.

In the same way that important features in the terrain appear and evanesce before the senses of wayfinder, the mediations of the design engineers who produce radical breaks appear and evanesce. Hand motions and many-ness of sketches, never meaning to be the final thing, but meaning to help unpack metaphors, vehicles, temporary signs that change along the way. Objects may sometimes stay the same but the gestures associated with them change. Stories abound, narratives and scenarios unfold often in wonderment.

An example of scaffolds and media used as scaffolds can be observed in the agency of rough, ambiguous sketches used to envision new and changing ideas. Another example was observed in several groups who produced rough paper prototypes, envisioning and acting out scenarios with them, and changing the prototypes as the scenario changed.

Anchors

Like Webmoor's maps, anchors make the claim to the essential truth of a thing, and elide specific sensory cues. Anchors elicit responses which suggest logical status.

Conversations surrounding anchors are limited to yes/no or optimization. Engineering drawings and refined models often act as anchors.

Webmoor's description of maps serves as a guide for understanding the grammar of engineering drawings. Like maps, engineering drawings represent physical volumes with thin, geometric boundaries.

We can easily substitute "cartographic map" with "engineering drawing" in Webmoor's account of the agency of maps: "The detail of the..." *engineering drawing*... "can often, however, elide the very feature...that it presumes to envisage in its visual conventions...Furthermore," *engineering drawing*s "by virtue of their 'univocal scientific strategy' flatten sensory data into the restricted medium of articulated lines and create 'gaps' and 'blank spaces'" [25].

An example of anchors and media used as anchors can be observed in the agency of a well resolved prototype. In one team, after five minutes, the "looks-like" model (Fig. 5), a well-resolved hard foam version of the engineering drawing, was given to the group. Each time the conversation veered to a new and different design, team members pointed to the foam model and returned to solving problems suggested by the foam model.

Note well, that with anchor objects, when the conversation hedges on a new idea, the object at hand brings the conversation back to the notion circumscribed by the object. Anchors are not employed as a vehicles to find alternatives. Instead, they are treated as the thing itself; they are the destination.

4 Discussion and Conclusion

The Scoping-Behaviors-Shared Media Triad in Action

In our study, teams who produced redesigns that exhibit radical breaks presented *many* alternatives to the engineering drawing. In contrast, teams that made incremental improvements presented a single solution to what they perceived to be the problem.

When asked how they redesigned the product, the strongest team in respect to radical breaks offered, "Obviously we don't have one new model. We've probably got three or four different themes and then we riffed on those."

This team began their work with 5 min of navigation behavior, asking "What is this?" and "How do we use this?", and "How can we fix this?" In the course of this investigation they enlist the engineering drawing very much like a map. They speak of features, and usability. By all indications, this group is fully anchored on the engineering drawing on the table in front of them.

After 5 min, the "experience-like" stimulus (Fig. 4) was introduced, at which point they enlist it to act out how and where it could be used. Next, they began to produce rough sketches, made hand gestures to develop and flesh out the sketch, and told narratives in which the object plays a role. The sketches changed with changing gestures and changing scenarios, as they felt their way through the imagined environment. Extractive scope rapidly shifted back and forth between features and functions and core functionality, while projective scope shifted in response and anticipation between usability, use case scenarios, and systems.

Subjects who achieve radical breaks seem to venture out, narrating and envisioning new terrain: the narratives support new object choices, and the object choices suggest new narratives. It is a land of multi-meaning and ambiguity. It is akin to the dream state, where causes change and things become other things and contexts change around things.

In contrast to teams who make radical breaks, teams who make exhibit incremental changes to the engineering drawing typically produce one alternative redesign. Their ideas seldom wander from the idea suggested by the engineering drawing. Indeed, in respect to navigating, they point to the engineering drawing when they begin to diverge from it, staying on course. These teams usually write requirements lists, often in the form of bullet points, and work from them to determine what is the next problem to be fixed. Their scoping is heavily weighted on features and usability, often concerned with ergonomics, the spacial organization of features, and the optimal type of button to use. In these groups we observe very little narrative/story telling. And they seldom do more than move their fingers on imaginary buttons.

We observed one unusual group, who reported two alternatives for their deliverable. This team spent 28.5 min, engaged in navigation behaviors (pointing to the engineering drawing when they began to diverge), enlisting anchoring media (pointing to the engineering drawing as primary, unchanging reference, creating hierarchical bullet points and requirements lists), and primarily limiting their scope to features and usability (toggle button versus scroll buttons). The result was making incremental changes to the design suggested by the engineering drawing.

With 1.5 min remaining, the team unexpectedly agrees to drop what they had been doing and "make a magical one". They changed scope from features to core functionality and usability to use case scenarios, exhibited wayfinding behaviors, (vigorously moving around, acting out how the rapidly changing object could be used in the imaginary environment, complete with grunting sounds) and enlist scaffolding media (one shared object leads to another rather than being the final and single version). At this point they made a radical break, producing a device that could be worn like glasses and is controlled by grunting, in order to leave hands free.

This occurrence was significant to us, because it demonstrated that teams with proper training (all three members of this group were trained as product designers), could turn on a dime and move from optimizing to making a radical break. This suggested to us that a more meaningful alternative to speaking about how an *actor is*, would consist in how an *actor acts*, which is to say to identify behaviors, rather than focus on actors.

This leads us to the proposition that teaching design engineers how to perform radical breaks may be a matter of teaching "moves". Rather than teaching "design thinking" strategies alone, practicing wayfinding behaviors, and manipulating scaffolding media may yield more effective results. "Design thinking" strategies alone do not tell the who story of how radical breaks are performed. High performing teams, in the context of radical breaks, jump from abstract concept to imaginary environments are co-evolved with new imaginary objects.

At the present, evidence suggests that radical breaks occur in the context of three imbricated elements which we describe as scoping, behaviors, and shared media. It appears that no single element works on it's own in creating a radical break, and all three dynamically influence one another.

We see scoping as a powerful conceptual strategy that designers enlist either tacitly or explicitly and is a primary factor determining outcomes in redesign activities. However, scoping alone gives an incomplete picture of how design teams work.

In the case of radical breaks, scoping is carried out through exploratory behaviors, like wayfinding. The paracosmos is effectively explored through "feeling" an imaginary terrain which has it's own rules. The terrain transforms as designers make their way through it, quickly shifting course as new possibilities appear on their horizons. Choices of where to go are made in situ, as narratives or stories are generated about how imagined objects, or counterfactuals, are to be used. Shared media in the form of scaffolds support wayfinding, suggesting new vistas and changing in response to unanticipated scenarios.

The data set we have created suggests that scoping and behaviors may trump media in practice. Nonetheless, shared media plays an essential role in supporting wayfinding and navigation, and thus is a determining factor in outcomes of redesign tasks.

Acknowledgements The authors would like to thank the Hasso Plattner Design Thinking Research Program HPDTRP for funding this project. We would also like to thank the folks at Stanford's Center for Design Research (a great place to explore how design is well done) and the Product Realization Lab (a great place where design is done well). A special thank you goes out to the design engineers who made our study possible and meaningful.

Bibliography

1. Blackwell, A. F. (2000, 8 December 2000). *Dealing with new Cognitive Dimensions*. Paper presented at the Workshop on Cognitive Dimensions: Strengthening the Cognitive Dimensions Research Community., University of Hertfordshire.
2. Blackwell, A. F., Britton, C., Cox, A., Dautenhahn, K., Green, T. R. G., Gurr, C., Jones, S., Kadoda, G., Kutar, M. S., Loomes, M., Nehaniv, C. L., Petre, M., Roast, C., Roe, C., Russ, S., A., W., & Young, R. M. (2001). Cognitive Dimensions of Notations: Design Tools for Cognitive Technology. In M. Benyon & C. L. Nehaniv & K. Dautenhahn (Eds.), *Cognitive Technology 2001* (pp. 325–341): Springer-Verlag.
3. Buxton, B. Sketching User Experiences: Getting the Design Right and the Right Design. Morgan Kaufmann, 2007.
4. Edelman, Jonathan Antonio, Alexander Grosskopf, Mathias Weske. *Tangible Business Process Modeling: A New Approach*. Proceedings of the 17th International Conference on Engineering Design (ICED), Stanford University, CA, USA, August 2009.
5. Edelman, Jonathan Antonio, Larry Leifer, Banny Banerjee, Neeraj Sonalkar, Malte Jung, Micah Lande. *Hidden in Plain Sight: Affordances of Shared Models in Team Based Design* Proceedings of the 17th International Conference on Engineering Design (ICED), Stanford University, CA, USA, August 2009.

6. Frederick, Matthew, 101 Things I Learned in Architecture School, MIT Press Cambridge MA (2007)
7. Gopnik, Alison 2009, The Philosophical Baby: what children's minds toll us about truth, love, and the meaning of life, Farrar, Straus, and Giroux, New York.
8. Grosskopf, Alexander, Jonathan Edelman, Mathias Weske. Tangible Business Process Modeling – Methodology and Experiment Design. Proceedings of the 1st International Workshop on Empirical Research in Business Process Management (ER-BPM '09), Ulm, Germany, September 2009.
9. Hatchuel, Armand and Weil, Benoît 2002 "C-K Theory: Notions and Applications of a Unified Design Theory" Proceedings of the Herbert Simon International Conference on « Design Sciences » Lyon.
10. Hutchins, E. Material anchors for conceptual blends. Journal of Pragmatics Volume, 37, Issue 10, 2005, pp. 1555–1577
11. Hutchins, Edwin, 1995. Cognition in the Wild. MIT Press, Cambridge, MA.
12. Ingold, Tim 2000 The Perception of the Environment: Essays on Livelihood, Dwelling and Skill London, New York: Routledge
13. Ingold, Tim 2007 Lines: A Brief History Routledge, London and New York.
14. Kirsh, D. (2010). Thinking With External Representations. *AI & Society*, Vol. 25, No. 4, February 2010.
15. Kirsh, D. (2009). Interaction, External Representations and Sense Making. In N. A. Taatgen & H. van Rijn (Eds.), *Proceedings of the 31st Annual Conference of the Cognitive Science Society* (pp. 1103–1108). Austin, TX: Cognitive Science Society.
16. Kirsh, D. (2009). Projection, Problem Space and Anchoring. In N. A. Taatgen & H. van Rijn (Eds.), *Proceedings of the 31st Annual Conference of the Cognitive Science Society* (pp. 2310–2315). Austin, TX: Cognitive Science Society.
17. Latour, Bruno 1986 "Visualization and cognition: thinking with eyes and hands", in Knowledge and Society: studies in the sociology of culture past and present Edited by H. and E. Long Kuklick, pp. 1–40.
18. Latour, Bruno 1993 We Have Never Been Modern Cambridge, MA: Harvard University Press.
19. Latour, Bruno 1999 Pandora's Hope: essays on the reality of Science Studies Cambridge, MA: Harvard University Press.
20. Schrage, M. Cultures of prototyping. In Bringing design to software book contents. 1996, 191–213
21. Schrage, M. Serious Play: How the World's Best Companies Simulate to Innovate. Harvard Business School Press, 1999.
22. Barbara Tversky, Masaki Suwa, Maneesh Agrawala, Julie Heiser, Chris Stolte, Pat Hanrahan, Doantam Phan, Jeff Klingner, Marie- Paule Daniel, Paul Lee, and John Haymaker "Sketches for Design and Design of Sketches" 2003
23. Tversky, Barbara 2006 "What Does Drawing Reveal About Thinking?" Lecture for Visualizing Knowledge Seminar, Stanford University
24. Suwa, M., and Tversky, B. External Representations Contribute to the Dynamic Construction of Ideas. Proceedings of the Second International Conference on Diagrammatic Representation and Inference, Springer (2002), 341–343.
25. Webmoor, Timothy 2005 "Mediational techniques and conceptual frameworks in archaeology: a model in mapwork at Teotihuacan, Mexico", Journal of Social Archaeology 5 (1):52–84.

If You Want to Know Who You Are, Tell Me Where You Are: The Importance of Places

Julia von Thienen, Christine Noweski, Ingo Rauth, Christoph Meinel*, and Sabine Lang

Abstract As we manoeuvre through life we often try to predict other people's behaviors and feelings; sometimes even our own. A classical take on the matter is to refer to character traits. But there is another source of information we may tap for our predictions – highly relevant and still often overlooked: knowledge of *where* the person is. At what place? In which context?

This article invites you on a journey of thinking about and exploring the marvellous impacts of places. We will start by visiting personality psychology, attending the quest of its professionals for ever-better behavior predictions. Subsequently, we will witness an experiment on the importance of places – seeing how a place setup may propel forcefully, almost mercilessly towards innovations. We will then browse personality psychology and other fields in search of fast and easy ways to make sense of places: How are they going to affect us? Who are we going to be there? Finally, we will draw together what we have found and construct a scheme to analyze or design places – which, of course, needs to be put to the test. . .

J. von Thienen • C. Noweski • I. Rauth • C. Meinel (✉) • S. Lang
Hasso Plattner Institute, Campus Griebnitzsee, P.O. Box 900 460, 14440 Potsdam, Germany
e-mail: julia.vonthienen@hpi.uni-potsdam.de; christine.noweski@hpi.uni-potsdam.de; Meinel@hpi.uni-potsdam.de

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_4,

When James Bond enters the opera of Bregenz in *Quantum of Solace* he does not wear a suit yet. But he makes sure he gets himself one even if that means that some less known, but well-built fellow from the actor staff will have to run around naked in the short scene he is allotted in the movie.

What James Bond does so cunningly here is something we all do, all the time – even though not necessarily with the same grace: We all adapt to the places we encounter. That may happen consciously, as in cases when we pick out clothing in such a way that it matches our surroundings. Yet, more often, we adapt without specific awareness. We speak up when entering noisy places like a well-populated schoolyard or a funfair. Conversely, many of us quiet down instinctively when stepping into a church or entering a graveyard.

Imagine an encounter between two complete strangers. One of them leaves no doubt he expects the other to undress instantly. (To set the scenario apart from the Bond movies, let's envisage an unattractive male in this case.) He does not have a gun, a knife or other weapon at hand to enforce the request. Yet, the other person obeys with little reluctance. In terms of feelings, neither of the two seems particularly joyful or disgusted. The undressing person could be you. Does that sound improbable to you? Chances are, however, you do behave and feel accordingly . . . when you are at the doctor's.

Settings affect our feelings and behaviors enormously.

Our ability to adapt both our feelings and behaviors to the changing contexts we encounter is sheerly amazing! And we tend to adapt so naturally we seldom recognize the changes at all.

While that seems true for us in our everyday life, it has also been true for many professional psychologists.

The scientific discipline of personality psychology has always tried to explain and predict human behaviors and feelings. But, how should we go about it? Many professionals equipped themselves with an approach taken straight out of everyday-life. It is the approach of trying to figure out what someone's character is.

James Bond is the daredevil type, isn't he? That's why, for instance, he does not engage in long-term relationships but rather opts for changing girlfriends over time. And that's why he is going to be accompanied by varying women in forthcoming films too. (Of course, if he were a real character we'd probably limit our prediction to saying that he will try to find favour with the diverse beauties he is going to meet, while being less certain regarding the success.)

For good or bad, the character trait of being "the daredevil type" has not been at the centre of scientific personality research so far. Traits that have been accorded more attention are, for instance, extraversion vs. introversion, conscientiousness or aggressiveness.

We often try to explain and predict behaviors in terms of character traits. Is that a good idea?

When you are determined to possess a particular personality trait, generally the notion is that you have corresponding behavior tendencies more or less regardless of the particular situation you are in. If someone is found to be "of the aggressive type" we should beware that he will readily engage in disputes, be that at home, at work or at the supermarket.

But it turns out: The belief in broad-spanning traits that account for behaviors in all the diverse contexts possible, may not be entirely warranted in the end.

If aggressiveness, for instance, was a general trait, someone who tends to act aggressively at home should tend to act aggressively at work too. Whether this is actually the case and whether people do behave consistently in different contexts, is, of course, something we can study. And the issue has been studied. Indeed, it has been studied over and over again.

Mathematically, the degree of behavior consistency is commonly calculated using correlations. When a person's behavior is perfectly consistent in different contexts the correlation takes on a value of 1. When there is no consistency, the correlation yields a value of 0.

After many decades of intense personality research all around the world it has become clear: There seems to be some kind of "ceiling value" that is hard to surpass for trait based studies on behavior consistency. This is a value of 0.3.

However, this is just the correlation coefficient. To find out what percentage of the observed behaviors are consistent across situations, the correlation value still needs to be squared. Thus, we end up with 0.09 or, to put it differently, 9%!

While this number means that people show some behavior consistency in changing situations, it also means that the consistency is not exactly sweeping. Something such as 90% or more of the behavior variance remain unexplained and unpredicted by the classical trait approach.

In 1968 the now-famous personality psychologist Walter Mischel shocked his colleagues by being very explicit about these shortcomings of classical trait theory. While many of his trait-searching colleagues tried to defend their outlook, especially at first, successive studies and meta-analyses generally confirmed the picture Mischel had outlined already in 1968.

As a consequence, personality psychologists tried to tap other sources of information that would allow them to improve their behavior predictions. In this regard, Mischel again led the way. And he led straight to the point where this chapter took off: the diverse contexts or situations people encounter. They are important predictors of behavior as well!

Character traits don't explain why the same person often behaves differently in differing contexts.

Mischel and many of his followers were not only interested in the degree of behavior variance that could be accounted for when now looking at contexts in addition to traits. Indeed, his primary interest was something different; and again many personality psychologists followed him. The new aim of personality research was to identify individual patterns, asking how probable a certain type of behavior was in a particular context – for a particular person.

Think of James Bond. He certainly shows a strong tendency of aggressive behavior whenever he is at the stronghold of a bad guy. Meanwhile, when Bond is with the MI6, he is somewhat more likely to act cooperatively. With Bond's antagonist, things are typically the other way round: aggression towards Bond and his fellows from the secret service, (some) cooperation with the fellow bad guys. So the average probability of aggressive behavior may well be the same for James Bond and his antagonist! A classical trait outlook on aggressiveness would thus deliver the same behavior predictions for the two. But with Mischel's approach our predictions improve because we don't adhere to overall-averages but rather look at context-and-person-specific behavior probabilities.

This chapter will take another turn. We shall not be concerned with context-specific behavior tendencies unique to some individuals. Instead, we will be looking at tendencies many people share when entering a certain context. These would be tendencies such as becoming quiet at a graveyard, speaking up on a busy schoolyard or, hopefully, much more interesting tendencies not to be thought of so easily.

Character traits don't explain why people of differing personality types often behave similarly when sharing one context.

The particular context that has been – and will be – central to our research is a very specific one. It is the context of working towards innovations. Indeed, we may even be more specific: It is the context of Design Thinking as a means to arrive at innovations, studied mainly at the d.schools in Potsdam and Stanford.

Yet, as will soon become clear, concentrating on such a specific domain does not mean you can shirk general questions, such as: What is the difference between "places", "situations" and "contexts"? Which places (or situations or contexts) shall we regard as alike, which as different? What would be a sensible way of comparing them?

But let's start at the beginning. Or, no, let's start with a confession regarding the beginning.

At first, like the early trait psychologists – just as many of today's laypersons out in the street – our research team also considered it the most obvious approach to look at *properties of persons*, potentially teams, but *not at contexts* to explain and predict the behavior we took interest in: the devising of innovations.

In 2009 we set up an extensive experiment in which we compared teams with d.school training versus those without, as well as mono- versus multidisciplinary teams. The question was whether they would work out equally innovative solutions to the design challenge we presented them with.

Our assumption was, of course, that there would be significant differences. We assumed that multidisciplinary teams would be more innovative than monodisciplinary ones. And that d.school trained teams would be more innovative than untrained teams. It turned out these predictions failed, and quite remarkably so (von Thienen et al. 2010).

Now, there may of course be characteristics of persons (or teams) that do a solid job in explaining and predicting the behavior of interest: the production of innovations. There may also be properties of the kind which relate to d.school training or academic diversity. But we didn't hit on them. And our reaction was much the same as that of Walter Mischel when faced with the ailing predictive powers of classical trait theory. We too wondered what other sources of information we could tap to improve our predictions. And we too ended up considering contexts, or rather places, as likely candidates.

d.schools are very special places. In what ways? Dear reader, please bear with us for a moment. The issue can be quite abysmal and we shall try to provide solid grounds soon. Suffice it to say for now that d.schools *are* special places.

Working at such a special place – specifically, working at the d.school in Potsdam – was something all teams had in common in our 2009 experiment. Potentially there is something about this place that simply makes people innovative. Maybe this place is actually such a powerful facilitator of innovations that all teams, with or without d.school training, with or without academic diversity, were forcefully propelled towards the great solutions which they all did deliver at the end. Maybe that's why we couldn't find differences in favor of multidisciplinary or d.school-trained teams.

Can a place make you innovative?

So we launched a series of experiments to study the effects of places, d.schools in particular. In one of them, students of the social sciences where invited to participate in a two-day-workshop on measurement and test theory. What a dry and daunting subject! The only way we could have exacerbated the challenge would probably have been to announce Latin vocabulary and declension tasks for the breaks.

As a free-of-charge way to prepare for exams, the workshop found favor with quite a few students, who came from three different universities to the campus in Potsdam.

The workshop began like many other preparation classes in Germany, with a lecture and a test. If you feel somewhat deflated as you read this: Good. You obviously understand the situation and empathize with the participants. If you don't feel deflated yet, please begin sensing a strenuous labor now, the labor of wrestling with some unmanageable material where there seems to be a clear line between right and wrong answers, between valid and invalid proceedings. But you can only guess where in relation to that line you manoeuvre from moment to moment... all the while you could have stayed at home to sleep late: It's weekend.

After the entry test, a challenge was handed out to the participants that they would work on for the rest of the weekend. The challenge was well in line with Design Thinking outlooks as it focused on human concerns. Actually, it even introduced a particular "persona", someone with individual needs, resources and limitations. That persona was Anna.

Up for a challenge?

Anna is a 16 year old girl who wonders how she comes across in different outfits. Having tried a number of hair colors and multiple clothing styles she noticed how hard it was to find out what other people really thought about her appearance. When she asked them personally, maybe some people lied because they didn't want to be offensive. Maybe some answers were meant ironically...

The challenge our workshop participants had to work on was to devise an approach for Anna so that she could find out what people really thought about her outfits. This could be done by taking an existing approach from measurement and test theory and applying it to Anna's concrete scenario. Or it could be done by devising a new approach, maybe tailored to Anna's particular needs, paying close attention to what was feasible for her.

Once the challenge had been given out, the group was split up so that the participants could begin to work in different places. And different they were!

In the "classroom" condition, the students were faced with a modern lecture room.

While all workshop participants remained within the same building, they now continued their work on different floors. In the "classroom" case the students did not see an old fashioned green chalk board, but its modern equivalent: two whiteboards at the front end of the room. Tables and chairs were of a common design and arranged in straight rows, facing the whiteboards – or – the place where normally a teacher stands. Apart from the lecture room there was only the hallway.

In the "d.school" condition, the students worked at a place looking somewhat like a crossover between an architect's studio and Ikea.

The first place the students would get to see as they entered the d.school was a lounge that looked somewhat like a living room: big red couches and white coffee tables, shelves with a manageable number of colorful books, boxes with loads of craft supplies and even toys.

Then, there was the workspace. It was about as big as the "classroom" on the other floor. Here, the tables and chairs were not of the conventional type. They were as high as bar counters and bar stools, so you could just as well sit or stand there. Then, there were whiteboards too, but a lot more of them, and additional craft supplies: post-its, many different colorful pen, glue sticks, scissors and the like.

In general, at the d.school the furniture is highly mobile since all the heavy equipment is installed on wheels, including the big red couches in the lounge. There is a great variety of equipment for prototyping, exercising or playing; there is lot's of technology available and team spaces are set up in such a way that groups may work actively in some sort of privacy while staying in touch with the world around them.

In both conditions the participants would work in teams about twice the size of typical d.school teams. Additionally, two "facilitators" were present on each floor. (Thus, later on there would be a way to check whether both facilitators agreed in their observations.)

It was not communicated to the participants which particular function the facilitators should serve.

Yet, in both places the participants initially expected the facilitators to act as teachers. E. g., when the students at the d.school wanted to say something they wouldn't just speak. Instead, they would raise their hands and expect the facilitators to call them. As the participants spoke, they would look at the facilitators rather than at their team mates, occasionally asking the facilitators explicitly whether what they had said was correct and admissible.

The point came when the groups had to decide what kind of an approach they wanted to prepare for Anna.

The participants wanted to be conventional.

In both places the students opted against the development of a new approach. They wanted to stick to the existing corpus of measurement and test theory – because that's what you would need to know for the exams.

To proceed in this way, the students could make use of work bags they had received. These included introductory articles as well as schemata which would guide them, step by step, through common approaches of measurement and test theory. But, it should be mentioned in passing, without being emphasized too strongly, that we had been slightly mischievous in the formulation of Anna's case. None of the existing approaches really delivered what Anna needed.

Let's focus on one condition now and see how the group fared, specifically the group at the d.school. Having picked the one approach which they wanted to apply, the d.school students equipped themselves with the corresponding step-by-step guide and started their work. Quite suddenly the two facilitators were dismissed out of their roles as lecturers. They became handymen. The new requests they'd have to work on would be things like: make sure the printer works properly, provide a laptop from which to print, dig up a camera... a camera? Yes, the students decided one of them would be Anna. And they would take pictures of her in differing outfits. (Of course, the d.school is a place were you can dress up easily as there is quite a bit of workable material around.) (Fig. 1)

As the pictures had been taken and the outfits had received a proforma-rating no one was more surprised than the students that they could not figure out how to do the final calculation of their step-by-step guide. What was wrong?

Fig. 1 A student at the d.school trying hard to avoid a smile as her team produces stimulus material: The spectator will have to judge differing clothing-styles. He is not supposed to be taken in by varying facial expressions such as a serious look in one picture, a big grin in another

Nothing was wrong. Except, in passing the students had so fundamentally altered the structure of the methodological approach which they intended to follow that there was no way back into the original scheme. They had altered the structure to deliver exactly what Anna was looking for: a feasible way of getting at people's opinions on clothing.

The d.school students were innovative despite their prior decision to be everything but!

Now what? The team was quite alarmed. In standard classes on measurement and test theory you are never asked to come up with your own approach (at least in Germany as things stand now). And there is probably a reason for that, right? Maybe it is too complicated, mere mortals are not fit for the task. Or, maybe there are no alternatives to the approaches already listed in common text books. Since they probably yield the correct answers already, there is obviously nothing else to look for.

Being re-affirmed by their facilitators that the approach chosen for Anna was sensible, despite the fact that it had just been made up, the d.school team visibly relaxed. For a short moment, the facilitators were hoisted back into the position of teachers again: They were supposed to provide authoritative judgements of what was right or wrong, viable or illegitimate.

But then, once re-affirmed, what should the team do with their remaining time? The team finally decided they wanted to practice for the exams. They would take the scheme which they had tried to follow at the outset. But now they would truly adhere to it. They would pay closest attention to the guidelines, would make sure to proceed step by step exactly as the instruction sheet required.

Still, there was something irritating: Did that methodological approach really deliver what Anna needed? It did not! The d.school team had stumbled on a bone of contention. And they wouldn't proceed without resolving the issue. So, what to do about it?

The d.school team decided they would simply make up a new challenge. The new challenge would accommodate both Anna's interest in dressing styles and their own interest in preparing for the exams.

The d.school students iterated the challenge they worked on.

So, now, Anna focused on one particular clothing and wondered how differing people would conceive of it. Maybe it would appeal to boys in her age but not to her parents and grandparents. That might be useful to know.

With this new challenge in mind it actually made sense to apply the guidelines which the d.school team wanted to work through. And they worked them through successfully, occasionally turning to the facilitators, mainly to attain assistance with some technical details.

As the calculations had been completed the facilitators referred to the workshop agenda which scheduled a meeting with the classroom group. Everyone should meet on the ground floor and both teams should present what they had in store for Anna. So the next step was to prepare a short presentation.

The d.school students reflected for a moment, deliberated and decided: No! A short presentation at the ground floor simply did not seem to be the best way to get across what they had done. Instead of giving a presentation they would rather set up a little museum at the d.school. They would show the material they had worked with. They would show their diverse working stations which reflected differing stages of their work, including the final results. Instead of a presentation, they would offer a guided museum tour.

The d.school students changed the agenda.

What did the facilitators think about such a change? Actually, they were barely asked. At this point, neither the schedule nor the facilitators were accorded indefensible authority any longer. Their prior suggestions did not outweigh the issue of what made the most sense under the given circumstances. And finding out what made the most sense was a matter the students felt responsible for themselves. Of course, the help of the facilitators would be appreciated none the less. They too could move around some whiteboards to help create a decent museum!

Within two workshop days the d.school students had completely altered their role behavior.

Initially, the participants at the d.school had acted as classical students: They had raised their hands, asked the facilitators for right-or-wrong judgements and made sure to equip themselves with detailed instructions which they intended to follow closely.

In the work process the d.school group became more and more autonomous, the students took on more and more responsibility. Very soon they began to monitor closely whether their proceeding was in line with Anna's needs and interests, whether it "made sense" for Anna. In the end, the students did not only monitor whether their methodological strategy made sense for the target subject, which they were supposed to focus on. They also monitored whether the workshop agenda "made sense", which no one had asked them to do, and autonomously launched corrective actions. Obviously, the students had come to monitor questions like: What is our ultimate concern? Do our proceedings make sense in that regard?

At the d.school participants became highly attuned to questions of sense making.

Now, that much said about the d.school group – what about the classroom students?

Suffice it to say that in the classroom no fundamental change was made to the structure of an existing methodological approach. Rather, a standard approach was taken and the pragmatic modalities of its application were elaborated. The students did not take offence at the fundamental mismatch between Anna's case and the methodological schemes of existing approaches that had been handed out. While the agenda was criticized (e.g., for lacking sufficient breaks and theoretical parts) it was generally followed.

In the classroom the participants maintained a student role.

They were ready to follow instructions and did not consider it their responsibility to change things for the better.

So, in this study – as in others – it seems to become clear:

The d.school makes a difference!

And the difference is big!

There are differences regarding work results, e. g. how innovative they are, but also differences on a personal level: What roles do people take on? How much self-confidence do they develop? How much responsibility do they take on? How likely are they to make changes they consider sensible? What questions do they bare in mind, what matters do they monitor as they go along?

But sure enough, studies of this kind can only be a first beginning. For one thing, replications are always appreciated, of course. And then, there is evidence for our digging in the right place now. We may want to sophisticate our digging strategy even more, such as to carve out the interesting details!

When making comparisons as in the study just reported the fact may be quite clear that places make a difference. But *what about these places* actually makes the difference? In this particular study, for instance, is the unequal height of the tables an important factor? How about the presence versus absence of couches? And the differing mobility of the furniture? There are just so many differences one might think of, including, potentially, differences in the brightness of the rooms, predominant colors, the carpet etc.

Imagine yourself as a Design Thinking researcher standing at the centre of Potsdam's d.school, a notebook in your hand. Look around and make a list of all the items that, potentially, influence how people behave and feel there! And please do not forget to include in your list the specific attributes of your items; they may be important too! So, if you think that a certain lamp may have an influence, mention its color, its height, the material it is made of, its shape and the particular light bulb that has been installed. What spectrum of light does it emit? In what angle does the light fall off? What is the degree of luminosity it creates in the room? And what does it mean for the luminosity in particular places of the room when the outside lighting conditions change in the course of the day, or when the weather conditions change?

So, please make sure your list includes all these things. Then vary each aspect that may have an influence. And carry out an experiment, or two (you know, replications are important!), to find out about potential influences of that aspect regarding people's behaviors and feelings.

Now, of course, that may be a bit of work. But just image: Once you are done, you may have something quite valuable to offer. You can present a long – a very long table – naming all the different factors that you have varied and the corresponding behavior or sentiment changes you observed. Then someone who wants to design a place can take a look at your table. He may specify what behaviors or feelings would be valuable in his place. There, he could realize exactly those factors in your list that have been associated with the favored behaviors and feelings in your studies.

Wouldn't that be a wonderful way of designing places that exert favorable influences? Of course, the designer won't know if the diverse factors listed in your table actually have these favorable influences in his place, because there they are installed *together*. In your experiments, most of them have been tested separately; the overall interactions have not been looked at yet. But that is a minute drawback, isn't it?

Well, we hope you agree with us that a conventional research strategy of designing experiments to test the impacts of singular factors is not going to bring us forward within reasonable time. And the kind of result it would get us after strenuous labor does not seem all that helpful either.

We need a shortcut that allows us to analyze places efficiently!

If we want to study places sensibly, we have to get beyond unmanageable lists of variables. We need something that *brings order* into the vast multiplicity of potentially important aspects. We need a strategy to make sense of them!

When looking for practical advise, such a strategy might well be helpful too. If we can come up with good enough schemes or rules of thumb, maybe it won't be necessary to work through all possible factor-combinations any more to decide what to put in a room. Because, maybe, we'll be pretty good at telling in advance what will work and what will not!

Of course, there are certainly interior designers who have such a good *intuition*, they can tell in advance how particular places need to be designed to bring about this or that favorable effect. And they don't need a scheme.

But, we are in a scientific research program! What we want is something explicit and systematic. It should serve the democratizing function of allowing basically *everyone* to arrive at sensible predictions regarding the propulsive forces of places.

It looks like we have stumbled onto quite a job!

TASK ***We could use a scheme that helps to predict how people are going to act and feel at a place***.

A scheme, what could that be? Well, it may include rules of thumb. It needs to provide strategies of analyzing and comparing places. And it should yield predictions regarding the effects of potential future room setups – so that we may pick a good one to be installed in practice. Generally, we are looking for a strategy almost anyone could apply to design places with favorable influences on feelings and actions.

But how should we go about it? Let's keep the challenge in mind and go out exploring a bit. We need inspiration! What is out there already? What can we borrow and learn from?

Obviously there are professions where people need to estimate in advance how place setups will affect their users. Interior design may be a good example. So, let's turn there first…

Welcome to the world of interior design.

Here is, for instance, a piece of strategic advice given by the design professional Beverly Murphy.

> When I begin working with a design client, the first thing I recommend is that they look through as many design magazines as they can. I ask that as they do this, they tear out the pages showing rooms they love and details they like. These tear sheets could illustrate things as diverse as a fabric design, the way a built-in bookcase is constructed, or a piece of furniture that especially appeals to them. As they do this a pattern develops that shows both the client and me the style the client likes best.
>
> (Murphy 2005 p. 1)

Murphy suggests two levels of analysis. On a first level, we stick to the details: fabric design, construction details, single pieces of furniture and so on.

Now, we have considered the option of working at this level of detail before. If we were to mount a research program on the effects of places at this level of detail, we'd have some good news to announce: Soon, all around the world involuntary unemployment will be no issue anymore!

INSIGHT To analyze places efficiently, we need abstract categories.

In a second step, Murphy brings in the abstractions and categorizations she needs as much as we do. Her suggestion is to sort according to styles. Thus, two things as different as a fabric and a piece of furniture may go in one category if only they reflect the same style.

That will help with the issue of feasibility. Yet, for our project there is a disadvantage: People prefer different styles. So we will hardly arrive at the more general recommendations that would be particularly helpful in our context – such as: If you want people to feel more positively at a place, design it in this particular style!

Again, there is something to take away from the attempt: Obviously the design of objects (such as their particular style) does not take us straight where we want to go. After all, we are interested in behaviors and feelings.

INSIGHT In our context, rather than focusing on design proper, we need to get at psychological significance.

So, while we will miss out on the richness of interior design, time presses on and we do have an idea where to turn next. As it happens, the roads of our little journey converge once more and take us back to the point where the chapter took off:

There is this prospering scientific discipline of personality psychology where experts all around the world work hard to provide ever-better behavior predictions. Shouldn't they have something on offer for us? What categories do they work with as they provide behavior predictions?

Welcome back to the world of personality psychology!

First of all, there are traits. But recall Mischel and his take on the classical trait theory! Traits hardly account for 10% of the behavior variance that there is. And for our purpose, that is actually good news!

We couldn't exert any influence on people's behaviors and feelings by setting up places in smart ways if their behavior was predetermined by their characters anyway.

Or, maybe that is not completely true. One option would remain: We might set up places that change people's character traits upon entry. Then, people might act according to their traits but still be influenced by our spatial setups.

Yet, character traits are said to be very stable. So, our prospects for immediate success wouldn't seem too good. But they don't have to be either. After all, traits *do not* determine how people behave.

So, what else is a predictor – and potentially a shaper – of behaviors and feelings? What else has psychology on offer?

Indeed, once traits were given up as ultimate predictors, personality psychologists introduced a second category of analysis: situations! And the couple of "traits" plus "situations" is actually what they work with up to this day. It turned out to deliver quite valuable predictions.

Given our interest in place analyses, traits don't seem to take us anywhere. But situations might actually do the trick for us!

Maybe we arrive at an illuminating analysis of places if we focus on "SITUATIONS" as the central category of analysis.

The interior designer Beverly Murphy used "style-boxes" to sort material stuff (despite all the pluralistic details that there are) into a manageable number of abstract categories. Maybe it is a good idea for us to do the same. Except, we don't use "style-boxes" for our analysis, but "situation boxes".

At first sight, situations seems pretty remote from places. (So why should they be of help in a place-analysis?) But that is only a cursory view. Just think about it!

The situation of "having to stop at a red traffic light" typically comes about at a red traffic light. The situation of "buying something in a grocery store" occurs in a grocery store. And the situation of "enjoying cake and coffee in a café" clearly takes place in a café.

While there need not be a one-to-one correspondence between places and situations, there are obvious regularities. It is in no way accidental what situations occur in which places. After all: If there is no café around, you can't get yourself into the situation of sitting in one, enjoying cake and coffee there.

Frequently in life, certain situations come about in the same kinds of places, again and again. For most of us, that is the case at school. We are taught in conventional classrooms where we are basically expected to absorb whatever the curriculum happens to dish up. For familiar settings like these we have our familiar role repertoires, including behaviors and feelings. And we may be very ready to access them when cued by our surroundings.

Recently, we accompanied a group of students who spent two days working on measurement and test theory. The participants got started in a context they were highly familiar with: Seated in an orthodox lecture hall (place), they listened to a talk (first situation) and took an exam (second situation). Sure enough they found their way into the roles of classical students!

Now, in this first phase of the workshop the situations (of lecturing and testing) might have been more important activators of student-role-behavior than the place (the orthodox lecture hall). We don‘t know. But after the first introductory round the students were brought into the same kind of situation. The only thing that differed then was the place. One place (the classroom) looked like a familiar school setting. The other place (the d.school) looked nothing like it. In the first place, people retained their classical student-role behavior. In the second place, people acquired alternatives. So, obviously:

Places cue us into particular roles (behaviors and feelings).

It certainly makes sense for people to scan their surroundings for cues to figure out what behavior (or feeling) will be appropriate. Misjudgements in that regard may have quite unfavorable consequences. Just imagine someone who failed to recognize that he should have turned around before crossing the motorway.

Whenever a particular place setup (e. g., a major street ahead) typically comes along with a certain situation (e. g., cars coming from the side), it is reasonable to use the place setup as an immediate cue. Thus, we may hold ourselves ready for the kinds of behavior that are commonly appropriate in the corresponding situation (e. g., turning around, checking if a car is coming).

Places may have a signalling effect, telling the person: "Hey, you enter this kind of a situation. Behave accordingly!"

Our journey has taken us around quite a bit. We've visited interior design and personality psychology, in search of inspiration. And we did hit on something that seemed promising: a category of analysis that helps with behavior predictions, namely "situations". Let's take home what we found to see what we can do with it.

At home a major challenge awaits us. We want to build a scheme that will help to analyze and predict the propulsive forces of places. We wish to skip the Sisyphus task of experimenting with tiny aspects of places. What we look for is a shortcut, yielding reasonably good answers to the question: How are people going to feel and behave in a particular setup?

Let's design a scheme to analyze the propulsive forces of places!

To warm up, we may look for rules of thumb first, before trying our hand at being truly systematic.

Indeed, it looks like we need little more than the good old and well established rules of association learning to merge the material we brought home into two decent rules of thumb.

Place	Situation	Material indicators	Behavior	Feeling
What place do you look at?	*What situation do you look at?*	*Which things are typically around? (In which arrangement?)*	*What behavior is likely in this situation?*	*What feeling is likely in this situation?*

Fig. 2 A *Place-Situation-Analysis* (PSA)

1. ***RULE OF THUMB When equipping places: What to put in?***
   Put in things that are typically around in places/situations where people act and feel favorably.
2. ***RULE OF THUMB When equipping places: What to leave out?***
   Leave out things that are typically around in places/situations where people act and feel unfavorably.

And how about a systematic approach?

Well, let's draw together the categories that seemed promising in our research phase. Figure 2 gives an overview.

So, here we have a scheme. And it is quite obvious how we could use it as a shortcut when assessing places.

ANALYSIS If you want to analyze a place and predict people's behaviors or feelings…

try to identify material indicators. Watch out for room arrangements that typically come along with certain situations. Predict whatever behaviors and feelings are common in those situations.

DESIGN If you want to design a place, optimizing its propulsive forces…

ask yourself, what particular behaviors and feelings you want to support. Think of (other) places and situations where these behaviors and feelings are likely. Investigate them. How are they equipped? What is typical? What may be a decent speciality of a single place? Use these model-places as sources of inspiration for the arrangements in your place.

Some Technical Details for the Nerdish Minds

How do places, situations and contexts relate to one another?

A place is, loosely speaking, what you can identify on Google maps. It's a concrete location which you might mark with a flag. Or you could draw a line around it to distinguish it from adjacent places. Of course, these lines may have a pragmatic character at times. Maybe you want to mark the outer border of the garden where James Bond recovers in *Casino Royale*. Does a particular blade of grass at the fringe still belong to the garden or is it already exterior to it? You may just have to make a decision.

A situation is what's going on at a place – insofar as it is brought under some label, construed in a particular way. When you consider something as a particular situation you typify it. Looking at Bond and his girlfriend in the garden of *Casino Royale*, we have a situation of recovery from bodily harm, but also a declaration of love, a situation of harmony – whatever aspect you which to stress.

If the term "**context**" is to be charged with a particular meaning here too, it probably makes sense to use it as a super ordinate concept such that it refers to combinations of places and situations. Thus, ask yourself in what context you would find yourself if, all of the sudden, you were Bond (at that moment just mentioned). The immediate context would be: a garden and a situation of love confessions. If that "context" is too narrow for your taste, bring in the bigger picture by giving a more comprehensive description of Bond's situation.

So, we have constructed a scheme. We should put it to a test.

TEST Does the scheme allow us to make sense of the ways people act and feel at a certain place?

Having spent so much time looking at the d.school, it seems quite clear which particular place we should make our immediate test case. To apply the scheme, there are blanks to fill in the table.

Much has been said about *behaviors* at the d.school. Recently, we saw how the participants of a workshop behaved as they studied measurement and test theory. In just two days, participants working at the d.school completely altered their role behavior, while participants in a classroom maintained their initial student-roles.

But what about *feelings*? Or *situations*?

To fill in the blanks and see how our scheme fares, let's voyage a final time – to a place where we can find out more.

We are at the d.school of Stanford in the fall of 2010 now, at a meeting of Stanford's and Potsdam's Design Thinking research teams.

Major experts of the field are around and willing to think about places, d.schools in particular. Individually, the attendees consider a couple of questions on the matter and put down their personal answers on sheets of paper. Generously, these sheets are left at our disposal.

One question out in the room is this:

FEEL ALIKES ***"What locations other than the d.School elicit a similar feeling? How do you feel there?"***

Even though the attendees consider this question individually, their answers show a remarkable congruence. One might even get the idea someone was cheating... unless, of course, the answers are obvious.

The top three *Feel-Alike-Contexts* named by the attendees at the research meeting are these:

Home
Café
Playground or Kindergarten

Interestingly, the attendees regularly specify situations too even though they are asked for locations only. Obviously, the situations seem crucial.

But looking at the different locations named, situations are not specified equally often, e. g. no particular situations are mentioned regarding the playground. Probably, that's because it is obvious what situation is meant at the playground: playing!

But almost everyone who mentions "home" adds a certain situation he thinks of. When referring to this place, clearly it is not obvious what situation one has in mind. It may be working, watching news, having breakfast etc.

Regarding "home", you bet the situation people associate with a d.school-type-feeling is not "asleep in bed". Rather, people write things like: doing something with friends in the living room – having a dinner party with friends – cooking with friends etc.

Obviously, (1) *friends or classmates* and (2) some kind of *open-creative activity* without a precisely specified goal are essential ingredients for the d.school-type-feeling. Also, the situations named are frequently (3) of the *unofficial* type.

But what kind of a feeling are we talking about?

"Energetic and engaged", one attendee answers. Most of us would probably agree that she truly hits the mark (Fig. 3).

Everyone who has ever been to a d.school will instantly recognize a pattern when looking at figure 3.

Things that are typical of Feel-Alike-Places are also present at the d.school.

There are the couches and coffee tables we know from our living rooms, there are toys and craft supplies like in kindergarten, there are little groups of tables and chairs as in a café, and there is food.

Well, but we do have to admit, of course, that there is a lot of stuff at the d.school anyway. Maybe we find all those material indicators of Feel-Alike Places at the d.school simply because we find material indicators for basically any kind of place or situation there.

So, let's do the test. Having looked at *Feel-Alikes* of the d.school let's try the other way round too and ask for *Anti-Spaces*!

Place	Situation	Material indicators	Behavior	Feeling
What place do you look at?	*What situation do you look at?*	*Which things are typically around? (In which arrangement?)*	*What behavior is likely in this situation?*	*What feeling is likely in this situation?*
Home	in living room with friends	coaches, coffee tables, a manageable number of books, ...	cooperation, conversation/ exchange, being creative,...	energetic, engaged, safe, relaxed, free, curious...
	cooking with friends	kitchen, food...	"	"
Café	chatting with friends	tables and chairs arranged in small groups; coffee...	conversation/ exchange,...	relaxed, interested,...
Playground/ Kindergarten	playing with mates	toys, craft supplies, colorful and mobile stuff,...	being creative, trying out something new, playing,..	free, curious, engaged...

Fig. 3 A *Place-Situation-Analysis (PSA)* on Feel-Alikes of the d.school

What would that be? Well, having said that the d.school is a place where a certain behavior is considered favorable (being innovative, working co-operatively. . .), the *Anti-Space* is a place that makes this favorable behavior highly improbable.

Here is the open question for our Design Thinking researcher colleagues to consider. . .

ANTI SPACES ***"What location (already existing or not) makes it difficult to live Design Thinking?"***

Once more, the answers given by the attendees of the Stanford workshop are highly concordant. Here are the top three:

Prison
Conventional classroom/office/cubicle
Library

Clearly, a prison instantiates the opposite of freedom. It is characterized by locked doors and scarcity. You can't delve into the world to experience first-hand what there is and what it's like. In terms of feelings, boredom and a lack of joy are going to figure prominently among them.

In conventional classrooms and offices, there is typically a clear hierarchy: The boss or teacher tells you what to do. He decides what is right or wrong, viable or inadmissible. As one attendee points out, in a classroom or lecture hall you are also quite likely to encounter a situation where your teacher acts as a "95%-of-the-time-talker". In any case, others have little to say, no communication on equal grounds, no mutual learning. Typical feelings would be: bored, unengaged or simply bothered.

Place	Situation	Material indicators	Behavior	Feeling
What place do you look at?	*What situation do you look at?*	*Which things are typically around? (In which arrangement?)*	*What behavior is likely in this situation?*	*What feeling is likely in this situation?*
Prison	being held captive	locked doors, bars, material scarcity	ever the same, restricted routines	no joy, bored
Classroom	being taught	tables in straight rows, chairs all facing the front end	obedient, passive, reserved	bored, unengaged, plagued
Library	studying, doing research	an unmanageable number of books	shutting up, working alone, taking in second-hand knowledge	lack of feelings

Fig. 4 A *Place-Situation-Analysis (PSA)* on Anti-Spaces of the d.school

What about the library? First of all, it is a place where you are required to shut up which also means that you typically work alone, since you are not supposed to speak much. And, once again: You don't go out into the world to experience it first hand. Rather, you take in second hand knowledge: what the authors tell you they observed, what they tell you is right. Since knowledge is transmitted in language, the original experiences of the authors have already been predigested by them and are now served in manageable language-boxes. In terms of feelings, we have what could be described as a blatant lack of feelings! (Fig. 4)

When now comparing the Anti-Spaces with the d.school, again a clear pattern emerges. And guess what?

Things that are typical of Anti-Spaces are left out at the d.school!

Despite the fact that the d.school is so pluralistic in its design, the material indicators of Anti-Spaces are actually *lacking* at the d.school!

Instead of locked doors that shut you off in a prison, at the d.school you find walls particularly designed to be most permeable: They are moveable and consist of holes rather than of wall-material (Fig. 5).

In contrast to the prison scarcity you'll find colorful stuff and craft supply everywhere.

Then, there are no tables arranged in straight rows, facing some "teacher's place" at the front end of a room – as in conventional classrooms. Neither are there endless bookshelves filling room after room, as in libraries. Of course, there are no "be-quiet-signs" either.

So it looks like we can actually make sense of the way people behave and feel at a place, like at the d.school, by searching the setup for its situation indicators. And if this is a workable shortcut that saves us from the Sisyphus job of varying every detail when analyzing places, we may just as well use the spare time won: Grab your favorite jacket, be on the look out for a café or playground or any other location that makes you feel comfortable and let the issue of places linger a little.

What to take away from the journey we just finished? Well, if nothing else, take away this: There is a regard in which it is perfectly true to say...

Fig. 5 Walls at the d.school in Stanford: They are designed to be mobile and most permeable, consisting of holes rather than wall

We are all James Bonds!

Because even James Bond allows the places he enters to strongly influence his behaviors and temper. Like him, we all accord our surrounding an enormous impact on how we behave and feel.

But that does not mean we are completely at the mercy of whatever happens to surround us. People are designers! We may design places so that they suit our needs and wishes.

Hopefully, we have whetted your appetite for looking at places anew – at best: monitoring issues that you haven't accorded as much attention before. And, maybe, you'll even try your hand at shaping the propulsive forces of places that matter to you.

References

Mischel, W. (1968). *Personality and assessment*. New York: Wiley.

Murphy, B. (2005). *Flawless interior decorating: A style-by-style guide*. New York: Mc Graw-Hill.

Thienen, J. von, Noweski, C., Meinel, C. & Rauth, I. (2010). The co-evolution of theory and practice in Design Thinking – or – „Mind the oddness trap!“. In H. Plattner, C. Meinel & L. Leifer (eds.), *Design Thinking: Understand. Improve. Apply*. Berlin: Springer.

Creativity and Culture: State of the Art

Hannah Hyunjee Kim, Siddharth Mishra, Pamela Hinds*, and Lei Liu

Abstract Our project goal was to understand how creativity is defined across cultures, to identify key stimuli for fostering creativity in different cultures, and to understand how creative performance differs by culture. Based on a comprehensive literature review and several field studies of designers, we find that current research on creativity and culture is biased toward Western conceptions. Applying this Western-biased view of creativity, research concludes that the West shows greater creative performance than the East. The East, in contrast, emphasizes the value of re-interpreting existing practices and de-emphasizing originality. Most recent approaches to the study of creativity, however, measure the number of ideas and the level of originality as key indicators of creativity. We also found that scant research has been conducted to understand the factors that stimulate creativity in different cultures. Though factors such as extrinsic vs. intrinsic motivation and conformity pressure have been explored, results are inconclusive. We speculate on new directions for research on creativity and culture.

1 Introduction

Creativity is an attribute considered highly desirable by all cultures in various contexts, from education to the business world. What it means to be creative, however, can significantly differ by cultures. One example would be that some cultures emphasize novelty and originality in design, while others place less emphasis on it. Some cultures focus on creativity as being highly artistic whereas others see creativity being applicable in every facet of life.

H.H. Kim • S. Mishra • P. Hinds (✉) • L. Liu
Department of Management Science & Engineering, Stanford University, Stanford, CA 94305–4026, USA
e-mail: phinds@stanford.edu

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_5,

There has long been an interest in finding out how creativity can be increased and if it can be taught to people. High creativity is known to be triggered by various stimuli. These include social stimuli, such as competition, accountability, upward comparisons, and goals, and cognitive stimuli, such as novel associations, priming, divergent styles and incubation (Paulus 2000). It is not clear, however, whether these stimuli can be applied to all cultures consistently or whether certain factors promote creativity differentially.

Scholars in a variety of fields, including those in organizational behavior and social psychology, claim that particular cultures are better suited for creative work than others. Many researchers argue that Western individualistic culture promotes a greater potential for creative performance. A few suggest, however, that the Eastern collectivistic culture shows greater creativity in certain domains, such as those with strong visual and technical qualities.

We therefore focus on three aspects of creativity and culture:

- How do different cultures define creativity?
- What key stimuli foster creativity in different cultures?
- How does creative performance differ by culture?

To answer these questions, we undertook a comprehensive review of the literature, across a variety of fields, on creativity and culture. We also used several field studies we have conducted of designers around the world to understand creativity *in situ*.

2 Cultural Variations in Definitions of Creativity

Although creativity can be defined in many ways, scholars seem to have reached a consensus that creativity refers to an individual's ability to produce ideas or products judged by others as both novel and appropriate (Amabile 1983, Cikszentmihalyi 1996, Sternberg and Lubart 1995, 1999). 'Novel' refers to divergence from existing solutions and 'appropriate' refers to usefulness, correctness and value (Amabile 1983) and fitting the demands of the situation and the creator's needs (Stefik and Brown 1989). Whether this consensus on defining creativity applies to all cultures is questionable.

To learn more about how different cultures approach creativity, we compared primarily the East and the West and how they differ in understanding creativity. Overall, the fact that creativity is defined differently by different cultures is well documented, thanks to international scholars based mostly in the East, such as Rudowicz, Hui and Yue. Even with their efforts to understand different definitions of creativity in the East vs. the West, most of the published research in top journals defines creativity based on novelty and appropriateness, which our review confirms reflects a Western bias.

2.1 Novelty vs. Modest Alteration and Reinterpretation of Tradition

The degree of novelty and originality that different cultures seek in assessing creativity differs greatly between the East and the West (Lubart 2010). Western cultures have been shown to consider something highly creative only if it shows a high degree of novelty or originality. Weiner mentioned Western views on creativity as ‘creating novel and appropriate objects and ideas that dramatically depart from existing ones’ (Weiner 2000). Terms like “out of the box” are common in conceptualizing creativity, influenced by an emphasis on individualism, freedom of expression, and democracy (Weiner 2000).

The East, on the other hand, has been shown to view creativity as subtle improvements in existing products and process with less emphasis on novelty (Averill et al. 2001, Lubart 1999, Li 1997, Gardner 1989). Creativity is seen as a continuous process, not a disruptive, one-time intervention. Rudowicz noted that the East may prefer to rearrange a pattern or alter existing knowledge or practices modestly rather than initiate radical change or a complete re-conceptualization (Rudowicz 2004). Gardner and Rudowicz find this stems, in part, from unique Eastern educational practices. Chinese children are educated in a system with a well-defined power structure in which they regard their teachers as a ‘mini-emperors’ who should be followed unquestioningly. With this ‘mini-emperor’, the Chinese educational system takes even the most complex activity, breaks it into components, starts the child out on the simplest part, has him or her perfect it, and then moves slowly to more complex and challenging work (Gardner 1989), thus training children to resist challenging the status quo and to think deductively.

Our observations of Chinese designers in a communication technology company in China supported this preference in China for being creative within an “ongoing flow.” Though creativity is encouraged, designers were not allowed to go “too wild” on a new product design, as they believed that continuity from the previous model and achieving harmony with the history of the company was a key to success and critical for maintaining a good brand image. During the brainstorming sessions, for example, there was an automatic exclusion of wild ideas, where a moderator did not document the wild ideas and both follow up and discussion of those ideas were discouraged.

This difference in the importance of novelty in defining creativity leads to significant implications in current research on creativity. Most researchers define creativity as ‘novelty’; they tend to measure it by counting the number of ideas generated or by subjectively assessing how radical the ideas are (Goncalo and Staw 2006, Goncalo and Duguid 2008, Jaquish and Ripple 1984). Such measurement schemes, we argue, contribute to the biased conclusion that creativity is found more in the West than the East.

2.2 Individualistic vs. Collectivistic Orientations

One of the common dimensions along which the West and East are compared is individualism vs. collectivism. Individualism characterizes cultures in which people see themselves as separate entities and primarily associate with others on an individual basis whereas collectivist cultures are characterized by people seeing themselves more as part of a group with their behavior primarily influenced by the considerations of the collectives of which they are a part (Hofstede 1991). Western culture is said to be more individualistic, Eastern more collectivistic. So, logically, the West tends to link creativity with a more individual orientation and the East more with a collectivistic orientation. Rudowicz asked Chinese to assess the key characteristics of a creative person. Interestingly, respondents described a creative person as one who 'inspires people', ['makes a contribution] to the progress of society', and 'is appreciated by others', descriptions that did not occur in US investigations (Rudowicz et al. 1995). Niu and Sternberg argued similarly that whereas an individualistic Westerner sees more of personal success in their creative endeavors, a collectivistic Easterner may see more of the social and societal value an individual brings (Niu and Sternberg 2002). Consistent with this,Yue and Rudowicz (2002) asked study participants to nominate the most creative people in the past and today in Greater China and found that politicians, scientists and investors together accounted for ninety percent of the total nominations. Interestingly, when compared with the West, artists and musicians were rarely nominated in China. This study reinforces the finding that Chinese may be more concerned with the creator's social influence or his/her contribution to society rather than the level of innovation in his or her ideas (Rudowicz and Yue 2002).

2.3 Differences in Key Characteristics of the Creative Personality

When people were asked to name key characteristics of creative people, innovation and imagination were ranked highly across cultures. However, there may be certain characteristics that some cultures value more than others. Rudowicz and Hui (1997) found artistic/aesthetic appreciation and sense of humor were absent in Chinese descriptions when participants at subway station were asked to describe the creative individual. They verified their findings with a sample of individuals who were supposed to be highly creative (Rudowicz and Hui 1997). On the other hand, Westerners tend to believe that energy levels of people, a risk taking attitude and a sense of humor play an important role in creativity (Niu and Sternberg 2002). A risk taking attitude is also believed to be present in highly creative individuals, along with a wide range of interests which expose them to ideas from many perspectives.

This difference in characteristics of creative people suggests important implications for the validity of well-known creativity tests, e.g. the Torrance Test of Creative Thinking (TTCT) and Urban-Jellen's Test of Creative Thinking-Drawing Production.

Both tests are known as key measurements for assessing creativity and have been widely used since the 1980s. Test results, however, may be biased toward the West since both include a sense of humor as a key dimension for creativity. As a result of extensive use of these tests, it is not surprising that research concludes that the West vs. the East shows greater creativity across a variety of tasks (Huntsinger et al. 1994, Rudowicz et al. 1995).

3 Cultural Variations in Stimuli That Affect Creative Performance

Common factors that stimulate creativity have been documented by Amabile and Paulus. Paulus (2000), for example, argued that high group creativity is triggered by *social stimuli*, such as competition, accountability, upward comparisons/goals, and *cognitive stimuli*, such as novel associations/priming, attention, conflicts, heterogeneity, divergent styles and incubation. On the other hand, low creativity is reduced in groups by social inhibition, such as social anxiety, social loafing/free riding, illusion of productivity, matching, downward comparisons and by cognitive interference, such as production blocking, task-irrelevant behaviors and cognitive load. Whether these stimulating and inhibiting factors are applicable for all cultures, however, has yet to be established.

3.1 Extrinsic vs. Intrinsic Motivation

Amabile noted that motivation toward creative work can be categorized into two distinct types: intrinsic motivation, which arises from the intrinsic value of the work for the individual (such as its interest value), and extrinsic motivation, which arises from the desire to obtain some outcomes (such as rewards) that are apart from the work itself (Amabile 1993). Both motivations can have very different effects on subjective feelings about the work, eagerness to do the work and the quality of performance. Amabile argued that the creativity of the artist's body of work positively correlated with intrinsic motivation. Amabile and Sternberg have argued that people are motivated to be creative from intrinsic rather than extrinsic rewards (Amabile 1983, Sternberg and Lubart 1991a, b).

Some researchers who focused on exploring creativity in the East, however, have argued that, for the East, extrinsic motivation might be more effective at fostering creativity than intrinsic motivation, as creative values are interwoven with the creator's social influence or contribution in society (Rudowicz and Yue 2002). Chan et al. (2002) demonstrated that when European-American and Chinese college students were instructed to be creative, both groups produced creative drawings. They pointed out that normally similar experiments favor the West over the East and argued that if creativity becomes instrumental to success, as defined in an Eastern, collective context, Chinese might strive to be creative (Chan et al. 2002).

Differences in intrinsic and extrinsic motivation among design students also appeared in one of our field studies. While observing an interaction between college students and professors in several top design schools in Korea, Europe and the United States, interestingly, only for Korean students, recognition by his/her professors (a factor generally considered to trigger extrinsic motivation), played an important role in motivating students to be more creative. Not reflecting feedback from professors and sticking to their own design was seen as a less desirable characteristic for Korean students in design schools. Collectivistic cultures are said to have a more blurry boundary between the self and others than do individualistic cultures. As a result, recognition, particularly by people who are closely tied to one's identity, such as one's advisor may actually operate more as an intrinsic motivator than in the West where it is clearly extrinsic and generally interferes with creative activity.

Though hypotheses are emerging around how extrinsic and intrinsic motivation can affect creativity differently across cultures, no study yet provides clear evidence to unpack of these differences.

3.2 Conformity Pressure vs. Greater Flexibility

Social control is known to be a significant barrier for creativity (Amabile 1998, Oldham and Cummings 1996). Social control includes control in decision making, control of information flow, or even perceived control in the form of reward systems that put too much emphasis on increasing extrinsic motivation. Moscovici argued that conformity pressure is a significant barrier to creativity because it can discourage people from diverging from their group to suggest a new idea that others may at first find strange or even offensive (Moscovici 1985). Group creativity is stimulated by free expression of dissenting opinions because, even when wrong, they cause groups to think and solve problems more creatively (Gruenfeld 1995, Nemeth 1986). Amabile argued similarly that social environments that encourage autonomy or self-directed learning should be better for people's creative expression and implied that individualistic cultures might have better creative potential (Amabile 1996). Niu and Sternberg (2002) also found that Americans as compared with Chinese show more ease and spontaneity in breaking through constraints and expressing their artistic creativity in diverse ways and concluded that a social environment that encourages autonomy or self-directed learning fosters creativity.

A counter argument, however, holds that conformity pressure may stimulate creativity if it is channeled in the right direction. Flynn and Chatman identified specific cases of innovative firms with strong cultures, such as 3M, that have produced thousands of new products by creating strong norms that encourage employees to be creative (Flynn and Chatman 2001). This can be particularly applicable to the East, where the educational system is driven by strong norms and instructions (Gardner 1989). Unfortunately, no clear evidence exists regarding the role that strong social norms play in creativity in the West vs. the East.

4 Cultural Variations in Creative Performance

There has been much debate by researchers about whether or not certain cultures are better equipped to be creative than others. Although the debate continues, the bulk of accumulated scientific evidence suggests that individualistic cultures have more creative potential than collectivistic cultures. Deeper evaluation of these studies reveals that they are based mostly on tests already biased toward the West, due to measurement bias or the use of stimuli that favor the Western conceptualization of creativity.

Western conceptualizations of creativity leak not only into the measurements and stimuli for creativity, but also into the type of task against which creativity is measured. There are, however, clear differences between cultures in creative performance depending upon the *type of task*. Some evidence suggests, for example, that the West tends to outperform the East in aesthetic aspects of tasks (Niu and Sternberg 2001), whereas the East performs better in more technical aspects of tasks (Rudowicz et al. 1995, Huntsinger et al. 1994, Chan et al. 2002).

4.1 The East Outperforms the West in Technical Tasks

Rudowicz et al (1995), for example, explored the role of education in shaping creativity in Hong Kong. Were differences in creativity found in Hong Kong children vs. other countries? Did creativity differ within the sample based on gender? Hong Kong children were found to excel in the figural part of the TTCT tests, faring much better than American, Taiwanese and Singaporean children, but just slightly lower than German children. But they did not perform as well on verbal tests. This variation is explained by the difference in the channels of creativity used by respective cultures and countries. America places a greater emphasis on language arts, while Chinese emphasize visual creativity in part a result of extensive experience in learning character-based Chinese. Huntsinger et al. also noted that Chinese Americans show higher skills in both technical quality and creativity in drawing and handwriting than Caucasian-Americans. This seems due mainly to Chinese-American parents setting aside more time for their children to focus on fine muscle activities than Caucasian parents (Hintsinger et al. 1994).

These results suggest that the cultural context, e.g. the educational system, parental training, etc., and not just cultural values, such as individualism and collectivism, might influence creative performance.

4.2 The West Outperforms the East in Artistic Domains

In some artistic domains, it has been established that the West outperforms the East (Niu and Sternberg 2001). Niu and Sternberg conducted an experiment with 70 students in the U.S. and China, respectively. They asked students to make a collage and draw an extra-terrestrial alien and the results were judged by two separate groups of judges composed of Americans and Chinese. The result showed that American participants produced more aesthetically pleasing works than their Chinese counterparts in all aspects, creativity, likeability, appropriateness and technical quality. In addition, American judges used more stringent evaluation criteria than Chinese judges, perhaps due to the higher artistic level of American students' artworks (Niu and Sternberg 2001). This finding fits the Chinese concept of creativity used in their education system, where artistic and aesthetically pleasing works were not seen as highly important and greater emphasis was given to process and social utility.

4.3 Reexamining Creative Performance in the West and East

As noted earlier, many scholars argue that the West outperforms the East in creative potential and performance, regardless of task-specific strengths and weaknesses.

A highly representative study (Goncalo and Staw 2006) argues that individualistic cultures outperform collectivistic cultures in creativity. They measured creativity based mainly on Western-centric views: the number of ideas and divergent thinking abilities. They studied how creativity differed among individualistic vs. collectivistic cultures and found that, given specific instructions to be creative, the individualistically-primed subjects performed much better than those primed to be more collectivistic. They also found that individualistic groups devised more creative solutions. The authors attributed this finding to the possibility that individualistic groups put forth more ideas for discussion that, when combined, delivered better solutions. Collectivistic groups, on the other hand, were not as firm in putting forth their ideas and hence their solutions were found to be less creative.

Though this experiment reveals that an individualistic culture may perform better in creative tasks than a collectivistic culture, a few aspects require deeper investigation. First, the way creativity was defined during the evaluation process was highly Western-centric. The number of ideas and a subjectively created list of attributes were used as metrics to measure creativity and originality. These metrics can be said to be result- or product-oriented. Second, the brainstorming task used this study emphasized number of and "out of the box" ideas—a highly Western approach to viewing creativity.

5 Conclusion

Our literature review on creativity and culture documents how different cultures view creativity. We present findings on how creative performance is triggered in various cultures. We also show ways in which one could hope to increase the levels of creative performance and begin to speculate on how that might vary across cultures.

We believe that the biggest gap in research on creativity and culture lies in understanding what stimulates creativity. Scholars have proposed many hypotheses after finding differences in creative performance across cultures, but none of these studies have yet identified the mechanisms that account for those differences. We posit that extrinsic motivation vs. intrinsic motivation, conformity pressure vs. greater flexibility, and collectivist vs. individualistic values may play out differently in different cultures, particularly Western vs. Eastern. In addition, we propose that more research is needed to understand the unique interventions that might spur creativity in the East as well as more exploration into key cognitive interventions, such as prototyping, which have been shown to promote creativity in the West (Dow and Klemmer 2009, 2010). We have yet to understand whether or not these interventions operate similarly or differently across cultures and believe that this represents an important next step for further research.

In general, this area is ripe for further investigation, but requires cultural sensitivity and well-grounded methods for cross-cultural research. It is critical, for example, that studies use creativity measures that reflect both Western and Eastern notions of creativity to avoid the pervasive bias toward Western accounts of creativity. Much existing research is experimental, which leads to a rather simplistic evaluation of culture and misses the larger cultural context in which these behaviors are embedded. We argue for field research that compliments experimental work.

In presenting this work, we have been asked whether or not creativity using the Eastern definition is, in fact, creativity. This is a difficult question to answer because perfect translation is impossible. One practical way to approach this question is to examine the extent to which definitions of creativity are widely held by consumers within particular cultural contexts. Definitions of creativity, for example, would suggest that products that are more harmonious with existing product lines and brands will be more successful in the Eastern marketplace, but evidence is still needed to understand the relationship between conceptions of creativity in different cultures and perceptions of products in the marketplace. The link between the creativity of designers and the desirability of products in different cultural contexts, we believe, is ripe for future research and could have significant implications for how we think about creativity and design around the globe.

References

Amabile T.S. (1983). A model of creativity and innovation in organizations. *Research in Organizational Behavior. 10 (1),* 123–167.

Amabile T.S. (1993). Motivational synergy: toward new conceptualizations of intrinsic and extrinsic motivation in the workplace. *Human Resource Management Review, 3(3),* 185–201.

Amabile (1996). Creativity in context: Update to *The Social Psychology of Creativity*. Westview Press. 317.

Amabile (1998). How to kill creativity. Harvard Business Review Sep-Oct. 77–87.

Averill J., Chon K.K., Hahn D.W. (2001). Emotions and creativity, East and West. *Asian Journal of Social Psychology, 4(3),* 165–183.

Chan, Cheung, Lau, Wu, Kwong and Li (2002). Assessing ideational fluency in primary students in Hong Kong. *Creative Research Journal*, *13(3),* 359.

Csikszentmihalyi, Wolfe (1996). New conceptions and research approaches to creativity: implications of a systems perspective for creativity in education. *The International handbook of giftedness and talent.*

Dow, Klemmer (2009). The efficacy of prototyping under time constraints. *ACM SIGCHI Conference on Creativity and Cognition.*

Dow, Klemmer (2010). Parallel Prototyping Leads to Better Design Results, More Divergence, and Increased Self-Efficacy. *ACM Transactions on Computer-Human Interaction, 17(4), 2010.*

Flynn, Chatman (2001). The influence of demographic heterogeneity on the emergence and consequences of cooperative norms in work teams. *Academy of Management Journal*, 44(5), 956–974.

Gardner H. (1989). The key in the slot: Creativity in a Chinese key. *Journal of Aesthetic Education, 23(1),* 141–158.

Goncalo J., Staw B. (2006). Individualism–collectivism and group creativity. *Organisational Behavior and Human Decision Processes*, *100,* 96–109.

Goncalo J.A., Duguid (2008). Strong Norms as a Stimulant to group creativity: When conformity pressure facilitates the expression of creative ideas.

Gruenfeld (1995). Status, ideaology and integrative complexity on the US. Supreme Court: rethinking the politics of political decision making. *Journal of Personality and Social Psychology, 68,* 5–20.

Hofstede, G. *Cultures and Organizations*. Berkshire: McGraw-Hill Book Company Europe, 1991.

Huntsinger et al. (1994). Ethnic differences in early math learning: a comparison of Chinese-American and Caucasian-American families. *The Annual symposium of the Jean Piaget Soceith (24th, Chicago, IL, June 2–4 in 1994).*

Jaquish A.G., Ripple E.R. (1984). Adolescent divergent thinking: a cross-cultural perspective. *Journal of Cross-Cultural Psychology, 15(1),* 95–104.

Li J. (1997). Creativity in Horizontal and Vertical Domains. *Creativity Research Journal, 10(2/3),* 107.

Lubart, T. (1999). The concept of creativity: prospects and paradigms. *Handbook of creativity by Robert J. Sternberg.*

Lubart, T. (2010). Cross Cultural Perspectives on Creativity. *The Cambridge Handbook of Creativity.* Chapter 14, pp 265–276.

Moscovici (1985) Notes toward a description of Social representations. *European Journal of Social Psychology, 18(3),* 211–250.

Nemeth (1986). Minority influence, divergent thinking and detection of correct solutions. *Journal of Applied Social Psychology, 17(9),* 788–799.

Niu W., Sternberg R.J. (2001). Creativity influences on artistic creativity and its evolution. *International Journal of Psychology, 36(4),* 225–241.

Niu W., Sternberg R.J. (2002). Contemporary Studies on the Concept of Creativity: The East and the West. *Journal of Creative Behavior, 36(4)*, 269–285.

Oldham G., Cummings A. (1996). Employee creativity: personal and contextual factors at work. *Academy of Management Journal, 39(3),* 607–634.
Paulus (2000). Idea Generation in Groups: A Basis for creativity in organizations. *Organizational Behavior and Human Decision Processes,82(1),* 76–87.
Rudowicz E. (1995). Use of the Torrance Tests of Creativity in an Exploratory Study of Creativity in Hong Kong primary school children: A cross–cultural comparison. I*nternational Journal of Psychology, 30(4),* 417–430.
Rudowicz E., Hui A. (1997). The Creative personality: Hong Kong Perspective. *Journal Social behavior and Personality, 12(1)*, 139–157.
Rudowicz E., Yue X. D. (2002). Perception of the most creative Chinese by undergraduates in Beijing, Guangzhou, HK and Taipei. *Creative Research Journal, 14(3/4),* 387–394.
Rudowicz (2004). Creativity among Chinese people: beyond Western perspective. Book *Creativity: When East meets West*. 55–86
Stefik M., Brown J.S. (1989). Toward portable ideas. *Technological Support for Work Group Collaboration* (pp 147–166). M.Olson.
Sternberg, Lubart (1995). Defying the crowd: Cultivating creativity in a culture of conformity. Free Press, 326.
Sternberg, Lubart (1999). The Concept of Creativity: Prospects and Paradigms. Book *Handbook of Creativity*, 3–15
Weiner (2000). Creativity & beyond: cultures, values and change.

Part II
Creative Tools and the Importance of Prototypes in Design Thinking

Design Loupes: A Bifocal Study to Improve the Management of Engineering Design Innovation by Co-evaluation of the Design Process and Information Sharing Activity

Rebecca Currano, Martin Steinert, and Larry Leifer*

Abstract After having identified the existence and having conceptually modeled the nature of general design loupes in the past year's project, this year's focus lies on the systematic exploration of the individual designer's inherent reflective loupe. Based on analyzing artifacts, surveying experts, conducting inductive and deductive conceptual framing rounds, and observing controlled explorative experiments we were able to: (1) show the existence of reflective loupes; (2) identify actual practices in use by designers; (3) use reflective practices as meaningful proxies for reflective loupes that are not directly observable; and (4) create, capture and analyze concrete reflective practices in the controlled experimental environment of a laboratory. We next proceed to build upon these results to deepen our understanding of the cognitive mechanisms of reflective design loupes.

These studies have identified digital artifacts that allow automatic collection and analysis through the d.store software currently under development at HPI in Potsdam Germany.

1 Introduction

Innovation is the basis for economic growth (Schumpeter, 2006) and should therefore be maximized. It is necessary to understand how innovation occurs in order to systematically increase innovative potential. The foundation for innovation is laid through new concept creation during the "fuzzy front-end of innovation" (Kim & Wilemon, 2002). This front-end of innovation is poorly understood and presents one of the greatest opportunities for improving the innovative process

R. Currano • M. Steinert • L. Leifer (✉)
Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA
e-mail: bcurrano@stanford.edu; steinert@stanford.edu; larry.leifer@stanford.edu

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*, Understanding Innovation, DOI 10.1007/978-3-642-21643-5_6,

(Koen et al., 2001) (Reinertsen, 1999). Design Thinking has been shown to be a successful method to encourage the generation of new concepts during the front-end of innovation (Dym et al., 2006) (Brown, 2008) (Plattner et al., 2009). At the same time, there is still little understanding of how Design Thinking works in action and how it is best managed. It is the goal of this research to close this gap. Our overarching research question is therefore:

> How does Design Thinking work, and how can designers and managers systematically maximize the potential for the generation of novel concepts that sell?

Our prior research has resulted in the hypothesis that Design Thinking leads to new concepts through insights, which are gained through experimentation rather than by deliberation. The same research also suggests that managers and other reviewers can have a detrimental effect on the ability of designers to gain insights by requiring that experiments be pre-validated. These findings are depicted in Fig. 1 (Skogstad, 2009).

This model and other studies of the design process (Cockayne, 2004) (Cross & Clayburn Cross, 1995) (Skogstad et al., 2008) (Van de Ven, 1999) suggest that communication within design teams and between design teams and reviewers is instrumental to successful design activity. In preliminary research, we created a tool for capturing team communication signatures (digital communication artifacts and their relationships) from online communication channels such as email, wiki and file share systems and to construct team communication networks (Uflacker & Zeier, 2009). This research suggests that the computationally observable structures in online team communication give evidence of Design Thinking elements and can indicate design process performance.

Our goal for this phase of the research is to explore hypotheses on the importance of design loupes during the design process and to ultimately discern performance indicators that can be measured and tracked automatically in real time. Design loupes are a series of focusing lenses at different scales for the iterative exploration of reflective design activity. This report focuses, in particular, on the individual designer's inherent reflective loupe. The results of our research are expected to

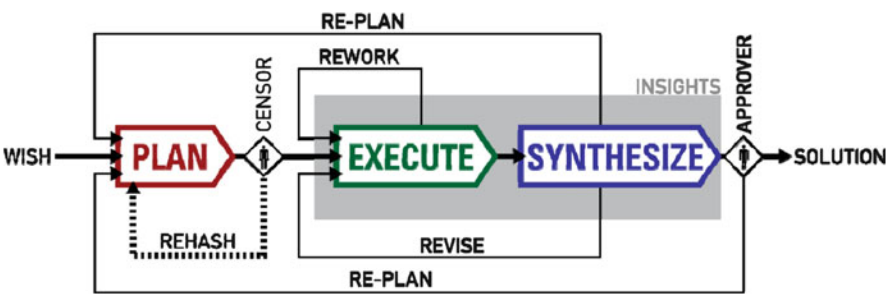


Fig. 1 "Unified Innovation Process Model for Engineering Designers and Managers" depicting the kernel of the design process. It shows where designers gain the insights to advance a design and where reviewers intercept the design process at the censor and approver gates (Skogstad, 2009)

provide designers, managers and researchers with a better understanding of design innovation in general and the internal reflective loupe in particular. We are developing a set of practices designed to enhance creative ideation for these stakeholders to put their understanding into effect.

2 Design Loupes as Fractal – The Inner Reflective Loupe

The model depicted in figure one represents generic design loupes, similar in its basic structure to the prototype and iteration model usually depicted as circles (Thomke et al., 2000). However, our research leads us to believe that those circles may in fact be seen as fractals. These design loupes are recursive activities that can be identified from a macroeconomic societal level to a microeconomic company level to the design and development team level and even to the level of the individual designer. As our research generally aims to understand, support, and facilitate the individual designer and the design team, we have focused on the last and tried to systematically identify existing reflective loupes and their impact on the design outcome. As the measurement of design outcome or design performance remains the holy grail of design research (Skogstad et al., 2009), we have opted for an iterative research design, modeled on Glaser and Strauss's grounded theory approach (Glaser & Strauss, 1967) (Glaser & Strauss, 2007). Thus we commenced without having in mind a set model to be tested. Instead we identified several research questions that we have tackled in succession. Each round of qualitative and experimental probing led to new and refined insights into the existence and workings of the reflective inner design loupes. For our research approach we relied on all three established methods: inductive, deductive and abductive. As such, our methods included observations, surveys, experiments, and expert workshops.

To launch our current analysis, we asked ourselves the following guiding question:

Does reflection help designers?

And more specifically:

How can designers use reflection to support creative ideation?

Based on literature analysis and the elaboration of a working definition of reflection in the context of design, we have opted to break our guiding question into four concrete research questions:

1. *Can we see evidence of reflection in design practices such as idealogging?*
2. *What other reflective practices do designers use?*
3. *With what dimensions can we frame reflective practice?*
4. *Can we identify characteristics of reflective practices that enable creative ideation?*

In the next section we will take a closer look at each research question, using a dictionary definition of reflection as starting point.

3 Exploring Reflective Design Loupes

Merriam-Webster's online dictionary lists a variety of definitions for reflection – entities and actions (transitive and intransitive), physical and mental, and abstract and concrete (Merriam-Webster, 2010). The ones which most resonated with us were:

- the production of an image by or as if by a mirror
- a thought, idea, or opinion formed or a remark made as a result of meditation
- consideration of some subject matter, idea, or purpose
- *obsolete* : turning back : return

We weren't, however, content with Webster's definitions, as they left us with a somewhat nebulous jumble of concepts, and therefore did not help us to understand the characteristics of reflection as it happens in design activity. We therefore chose a purely explorative starting point for our research on the reflective loupe, to search for evidence of reflection and to characterize our findings.

In keeping with the topic of this research, we used an iterative loupes approach, and probed the reflective loupe from several different points of view. We looked at the role this reflective loupe plays in Design Thinking in light of our personal experiences, discussed definitions considering a variety of common meanings of reflection, and prototyped several visual models of reflection. Each probe addressed one of our research questions and helped to refine our understanding of the reflective loupe. They are presented next.

3.1 Can We See Evidence of Reflection in Design Practices such as Idealogging?

The activities around our first research question aim to present evidence for the existence of reflective practices in design. As an exemplary source, we have opted to screen and analyze idealogs. Idealogging is a common design practice, particularly within the inner design loupe, and is most utilized during the fuzzy front end of design projects. We reviewed idealog data to discover evidence of idealogging as a reflective activity. We chose idealogging as a proxy for design practice, since it is a typical output of design projects and it is easily recordable. As a proxy for reflection we examined sketches and notes, mindmaps, doodles, and assigned weekly reflection entries, all of which are present in the idealogs.

Our data was comprised of scans from 22 students who had taken a senior design elective, Ambidextrous Thinking, at the University of Maryland. The review process was iterative, and was initially focused on finding indications of reflection within the idealog artifacts.

First we selected four students' idealogs demonstrating varying styles/techniques. We printed out and displayed digital scans one at a time, on the wall or a table, so that each student's entire idealog could be seen at once (see Fig. 2 for examples).

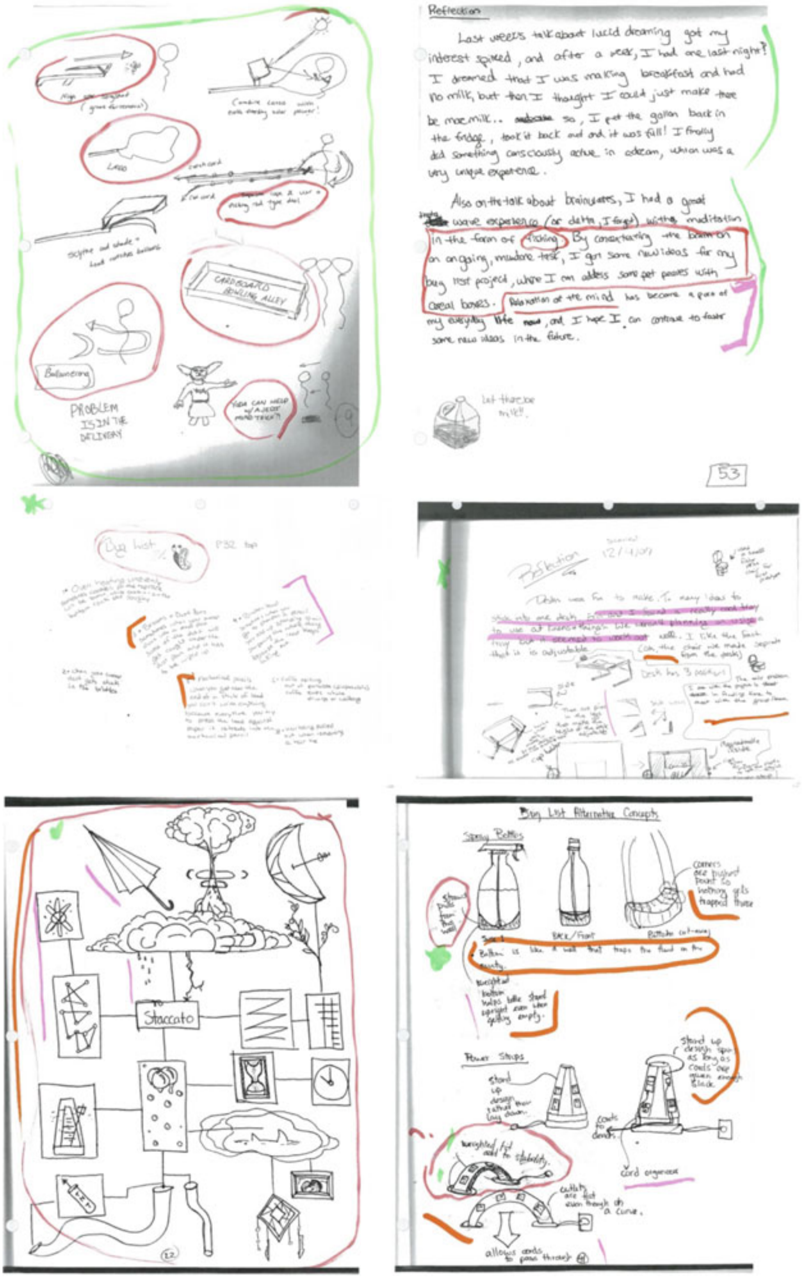


Fig. 2 Example pictures from the students' idealogs, coded by the researchers. From the top left it shows: project design ideas, weekly reflection task, bug list, weekly reflection, mind map and final project ideas

Then three researchers went through them simultaneously, marking them according to our individual interpretations of their idealog material, noting anything that implied evidence of reflection. We discussed each idealog individually and in conjunction with the others, describing what we had marked and justifying why we had identified it as evidence of reflection.

In doing so, we were able to identify different forms of reflection evidence. This led us to distinguish and name a variety of reflection modalities. We compiled them in a comprehensive list of reflection practices observable through the idealogs.

Each idealog we reviewed led us to refine and solidify our understanding of what forms reflection in idealogging could take. We expanded on or refined the list with each idealog. By the fourth idealog we found that our list was not growing significantly, so we moved away from the data for a while and started looking more closely at the reflective practices we had identified.

This list formed the basis for a discussion on the types of reflection and how they might be grouped. This step enabled us to characterize different kinds of reflective practices in early stage design We chose a subset of activities to explore in-depth. We recompiled the full list, and reorganized it into (a) most, (b) moderately, and (c) least interesting sub-lists, based on common agreement between the researchers. This gave us a way to narrow down the set to the following distinctions:

- Imaginative reflection
- Skill-building reflection
- Metaphorizing
- Doodle-flexion
- Surprising reflection
- Reflection ghost
- Background reflection
- Metathinking
- Reflection-out-of-action
- “What I didn’t do”
- Broader meta-categories of verbal vs. pictorial.

At this point we realized that the previously identified reflection ‘types’ were not described consistently with respect to language or level. To better understand these reflections we narrowed our scope and focus on the three particular reflection types defined by each researcher’s favorite (shown circled in the above list).

The three types we selected were “metaphorizing” (reflecting using metaphors), “metathinking” (reflecting on how one thinks), and “reflection-out-of-action” (reflecting outside of the work setting). We divided the remaining 18 students’ idealogs between us and informally coded them for instances of those three kinds of reflective practice. This gave us better consistency in the identification of reflection types in student idealogs. Based on the analysis, two distinct categories emerged from the cluster of different reflection types: those based on internal memories and those based on external observations.

At this point, we redesigned our approach to include this variation for systematically characterizing reflection in the idealogs, which we termed: (i) remembering

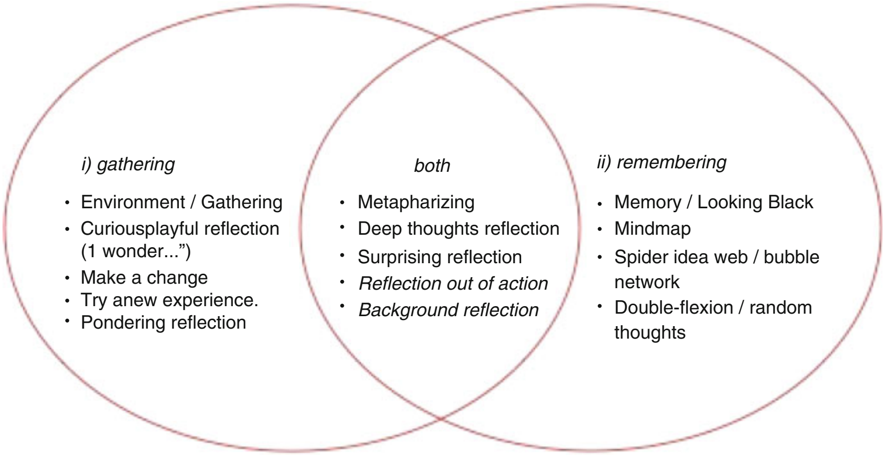


Fig. 3 Reflective practices based on (i) remembering and (ii) gathering. It emerged that some practices stem from both sources. Reflection-out-of-action and background reflection proved to be especially interesting

and (ii) gathering. These distinctions originated from a framing exercise we had conducted prior to the idealog analyses. Taking these expressions, we again classified the reflection types from lists (a), (b) and (c) according to whether they were primarily related to 'looking back at memories' or from 'gathering from the environment'. These different meanings resonated with both the idealog data and our understanding based on personal experience, which pointed to 'gathering' and 'remembering' as two primary sub-modes of reflection (see Fig. 3).

Some of our identified reflection types exemplified both expressions, rather than just one or the other. "metaphorizing", "reflection-out-of-action", and "background reflection" are three reflection types from our most interesting sub-list, which represent broader reflective activity. Many of the other identified types, outside of gathering or remembering, can be viewed as concrete reflection tools stemming from these broader reflection types.

Reflection-out-of-action and background reflection, in particular, caught our attention, as descriptive of many reflective activities, such as taking a shower or talking a walk. They include any activity that removes you from conscious effort and enables non-conscious ideation. By non-conscious, we mean ideation that is happening in the mind of the designer, but as a background thought process, and not actively attended to. Background reflection, more specifically, refers to reflection that happens when the mind is otherwise disengaged from what the person is doing physically. We use the term background for two reasons: (1) because the reflection happens as an incidental, background, rather than a deliberate, foreground thought process; and (2) because it is accompanied by a routine background physical activity that doesn't require mindfulness, but frees up the mind for reflection.

As these reflective practices are among those farther removed from the traditional workplace and work pace, they are largely unrecognized and underappreciated as potential sources of productivity. One can imagine that time spent doing them would not be allowed as billable work-time, and that designers employed by many firms and companies would be discouraged from spending part of their work day engaged in these types of reflective practice. Therefore we decided it to be useful to distinguish reflection-in-action (or -during-action) from reflection-out-of-action.[1]

The first research probe provided evidence in the idealogs of two different conceptual perspectives on reflection:

- remembering vs. gathering
- reflection-in-action vs. reflection-out-of-action

The following section describes our second research probe into the distinction regarding reflection-in-action and reflection-out-of-action, and into background reflection as a sub-category of reflection-out-of-action.

3.2 What Reflective Practices Do Designers Use?

The idealog probe indicated that the dual characteristics of remembering and gathering are a frame that can be used to characterize reflective design practice. It also pointed to the existence of many different ways in which reflection permeates the idealogging practice. It was necessary, however, to step back a bit and look at reflection in design from a broader perspective. We wanted to understand more generally how reflection contributes to ideation and what reflective practices designers employ to produce ideas, so we surveyed design experts from the Stanford Center for Design Research community and the Hasso Plattner Institute – Stanford Design Thinking Research Program.

At this point we started to believe that there is a relationship between background activity and ideation. We sensed that context and mental focus play a part in the reflective processes that underly creative ideation. We asked 20 survey participants in a Stanford design research seminar to complete a survey. We asked them to think of a creative idea that they had recently experienced, to recall where they were, what they were doing, and whether or not they were trying to come up with the idea at the time. While not every participant responded to every question, we compiled the responses that we did receive and coded them according to place, background activity, and intent. Fig. 4 presents and describes the reported reflective practice trends observed:

Eighteen participants answered the first question (Where were they at the ideation instance?). Of these, only three reported being at work when they thought of their idea. Of the rest, six were at home, and the remaining eight researchers were

[1]This conceptual separation must be differentiated from Schon's reflection-in-action and reflection-on-action (Schön, 1983).

	HowYouGetIdeas ME397 Ideation Survey			
	Idea	**Where were you?**	**What were you doing?**	**Were you trying?**
1	watermelon peeler for juicing	home - kitchen	making food	yes
2	unnamed	out - cafe	drinking coffee, chatting with friends	
3		home	walking	no
4	skateboard with novel controls	out - hospital	getting stitches	yes
5	audiobook for friend	work	not working	no
6	unnamed	home - shower	[showering]	no
7	unnamed	work - outside (quad hallways)	walking	yes
8	unnamed	out - outside	jogging	yes
9	Use role preference in videogames to find preferences in design/working teams	class	writing	no
10	unnamed	out - metro	sitting or standing	yes
11	baby bottle maker	home - room	ideating	yes
12	idea for gearbox	home - shower	showering	no
13	unnamed	work	sitting	yes
14	unnamed		I was trying the morphological method	yes
15	unnamed	home - living room	talking	no
16	unnamed	out - between Stanford and home	biking	
17	unnamed	out - car	driving, listening to music	
18	measurement device for blind people to see the level of fill	out - museum	viewing exhibition	yes
19	idea/discovery that my laws map to the hunter-gather mode	work - CDR for HPI workshop	communicating, drawing	no
20	playground cover and bar-baby stool and room separation		brainstorming	yes
	Lightly shaded = not sure	**Green Font = reflection-out-of-Action**		
	Shaded = not at work, not doing work activities, not trying to come up with ideas.	**Blue Font = background reflection**		

Fig. 4 Reflective practices demonstrated by designers in our survey group. Note the activities that take place at work vs. those not at work (equivalent to our previously described out-of-action) and note the activities out-of-action and the background reflection activities. Reflection-out-of-action, in this context, does not always happen physically outside of the workplace, but it does happen while the subject is not specifically engaged in work tasks

engaged in various transportation or recreational activities. One participant was at the hospital, having an injury cared for.

Likewise, for question two (What were they doing?) 14 of the participants reported being engaged in non-work-related activities at the time their ideas came to them. Of those who were at work, one reported not working at the time, and one reported being engaged in a communication-related activity rather than an ideation-related task. Two were engaged in work-related tasks while not at work.

Ten of the 17 participants who responded to question three (Were they trying to come up with an idea?) reported that they did try to come up with an idea at the time, while seven reported that they did not intentionally engage in reflection for ideation. We did not ask participants to describe the specific idea they had, though some offered related information (shown in the last column of Fig. 4).

In most cases, ideation took place outside of the traditional work situation, with respect to location and background activity. Subjects were almost as likely to get ideas when they are not looking for them as when they were looking. One respondent said that he finds various types of physical activity particularly helpful for different kinds of ideation. For example, he finds jogging to be good for situations to see structure, and walking helpful when he's dealing with mathematical problems.

These survey results support of the notion that reflective practices can be meaningfully categorized as "reflection-out-of-action", and "background reflection", in addition to more commonly recognized reflection categories like Schon's "reflection-in-action". These survey results support "reflection-out-of-action" and "background reflection" as meaningful distinctions which capture a greater range of reflective practice than the existing commonly recognized categories like Schon's "reflection-in-action".

Reflection-out-of-action happens out of the workplace and the context of work activities. Examples are when participants said they got their ideas at home while making food, or while chatting with friends over coffee. Background reflection happens during routine physical activities. Examples are when participants noted that their ideas came while jogging, or while in the shower, which enabled them to think mindfully about other things.

Having given evidence of reflective practices based on design artifacts (Sect. 3.1) and having obtained a survey-based list of reflective practices actually in use (Sect. 3.2), we decided to take another step back, with the aim to make further distinctions regarding reflective practices and the impact of context.

3.3 With What Dimensions Can We Frame Reflective Practice?

From research question one, the idealog probe, we came up with a preliminary list of reflection types as seen in common design practice. The survey probe offered support for two of these types, reflection-out-of-action and background reflection, as commonly practiced and applied in productive ideation. The third probe was designed to further our understanding of how the various types of reflective practice differ, and where they fit within a general framework of reflective practice.

Direct reflective practices, such as sketching design ideas, seem to support 'gathering' more, while indirect practices, such as taking a walk, seem to support both 'gathering' and 'remembering' with less discrimination. This is because:

- *Direct* practices are more tightly centered on the immediate design problem at hand, and less readily tempt the mind to wander off into more contextually distant associations or more temporally distant memories.
- *Indirect* practices are more loosely centered on the immediate design task, and more explicitly allow the mind to wander both contextually and temporally, since they expose the senses to stimuli which are more mundane and therefore associated with a broader set of past experiences.

To illustrate this, taking a walk (an indirect reflective practice) encourages the designer to become aware of the surrounding environment, the landscape, people walking by, conversations happening around them, the air, the light, trees, grass, and cityscape. These, or similar environmental aspects, are a common part of past experiences, but are oftentimes not consciously noted. Perceiving them during reflection incites the mind to draw associations between features of the surroundings and the design task at hand, in a gathering reflection mode. Additionally, it may open up pathways to dormant memories, thus serving as links between past experiences and the current task, exemplifying the remembering reflection mode.

Design sketching (a more direct reflective practice), on the other hand, provides a steady inflow of visual stimuli, inciting the cycle of seeing new things, and making changes or moves based on the surprises experienced (Schön, 1983) (Goldschmidt, 1991). Design sketching also keeps the mind focused on the task at hand, and does not readily open pathways to more distant memories. Thus, it favors gathering over remembering.

Defining reflection was the starting point for framing the reflective loupe. Our definitions included: contemplation, looking back and observing mindfully, mirror image, indirect expression, manifestation, similitude, and echo. These various ways of characterizing reflection capture both transitive and intransitive meanings, actions and entities, memory-based and observation-based perspectives, and physical and conceptual references. Figure 5 presents a thesaurus map depicted as a mind map used for capturing a definition of reflection. Its components are devised from several rounds of the researchers' work sessions:

Several iterations on this exercise of defining reflection led us to the determination that reflection in design is perhaps best studied not as a disembodied concept, but in the context of reflective practices. Since we cannot get inside the designer's head (making this a black-box problem), a more practical way of learning about the reflective loupe is through the activities in which reflection takes place, which we

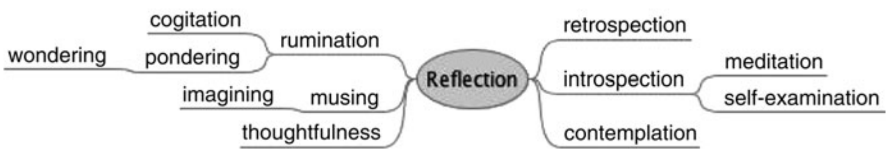


Fig. 5 This mind map represents a thesaurus map of reflection. We compiled this map from related words, as a second route toward developing a working definition of reflection

call reflective practices. We approach the study of reflective practices by observing and analyzing the behavior of designers and the artifacts that they produce.

> A reflective practice is any activity, which leads the designer to reflect, not necessarily on the activity itself, but on the design task, process, or goals through the activity.

The first step in framing our understanding of reflective practice in design was to represent it metaphorically, starting with inputs to reflection: environmental stimuli and recollected memories. These factors aligned with the remembering and gathering factors, which emerged from our definition exercise (in Sect. 3.1). They also resonated with our descriptions of our personal experiences with reflection (in Sect. 3.2).

To reflect in this context is to engage the mind with the goals of the design task, which typically includes creative ideation, by recalling and considering memories of past experiences, and/or observing and considering things in the present environment. As indicated through the survey (in Sect. 3.2), reflective practice can be either intentionally or unintentionally motivated.

We generated a series of diagrams to describe our Remembering-Gathering model of Reflection (see Fig. 6):

The end result of any reflective activity, for example, as depicted in the Remembering-Gathering Model, is the creation of a new design-specific idea.

To provide a space to map different kinds of reflective practice, we constructed a framework of some key elements of reflective practice (shown in Fig. 7). This framework drew on the personal experiences of the researchers involved and the models of the process of reflection we had prototyped. It is based on three dimensions: in-action vs. out-of-action on the y-axis, internal vs. external on the x-axis, and background vs. foreground on the z-axis. The x and y dimensions map well to the 'serious/real vs. playful/dreamlike' states of mind from Fig. 6c, and to the 'remembering, gathering' perspectives discerned through our prior research probes.

This conceptual model allows us to cluster reflective practices according to the dominant features of each practice. After having elaborated a theory-driven framework of reflective practices, we are ready to explore the same experimentally.

3.4 Can We Identify Characteristics of Reflective Practices That Enable Creative Ideation?

Following the idealog data coding and the previous explorations, we explored the reflective loupe in practice through an experiment. This enabled us to observe how designers use built-in reflection time and to find out if inserting specified reflection periods within a design task has any clear impact on the outcome. We ran the experiment with three pairs of designers (p1, p2 and p3), each tasked with designing a creative (novel + useful) product for the design loft at Stanford's Center for Design Research. This experiment was comprised of three explorations into scheduled reflection from three different angles, all utilizing reflective activities that we have termed "reflection-out-of-action".

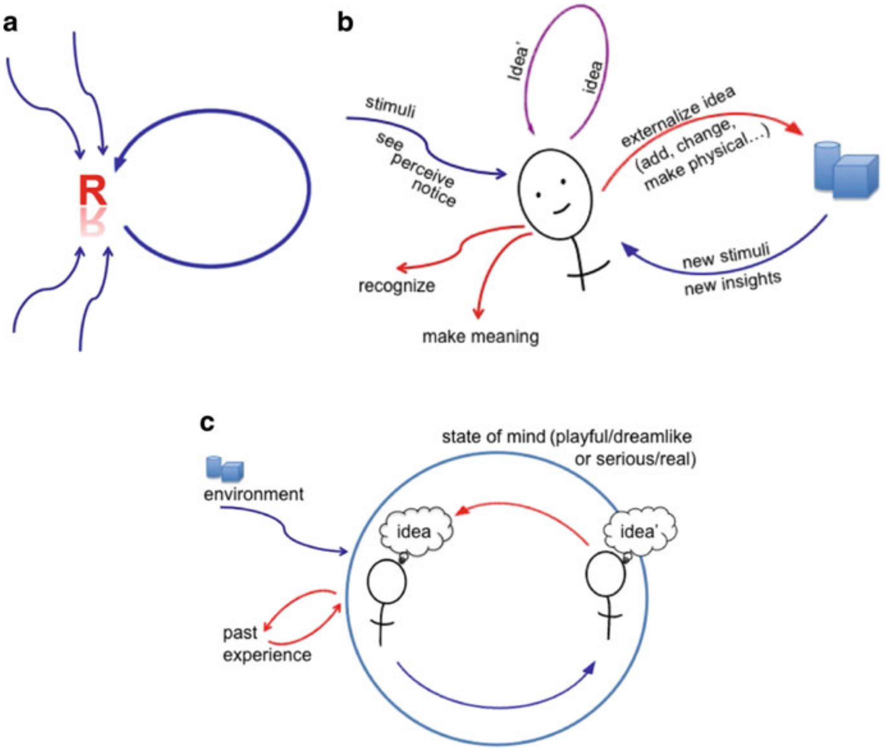


Fig. 6 Remembering-Gathering Model of reflection. This figure shows the interchange between remembering and gathering in the action of reflection. (**a**) shows the elements of gathering from the environment and the remembering. (**b**) expands on this, highlighting the sub-activities such as perceiving, recognizing, externalizing, etc. **c**) places the Remembering-Gathering Model in the context of the individual's state of mind

p1: In the first, we separated the design task into three segments, with two "structured breaks" between them, during which the participants were sent to separate rooms and asked to scan documents. Scanning documents was intended to serve as background background activity, to set the stage for reflective ideation. Two different scanning machines, one highly automated and one more manual and less familiar to the subject were chosen.
p2: In the second experimental setup, we invited the two participants to take a coffee break together, and gave them $10 with which to buy drinks and snacks. They were encouraged to enjoy the break and to chat together and were permitted but not pressured to talk about the project. We considered this setup to be a proxy for out-of-action, yet not necessarily background, reflection.
p3: In the third exploration, the participants were moved to a "play room" set up with Play-doh®, arts and craft materials, and a game and instructed to have fun and not necessarily focus onto the design challenge. The idea was to initiate an out-of-action state and to provide a variety of stimuli in a relaxed, creativity-supporting setting.

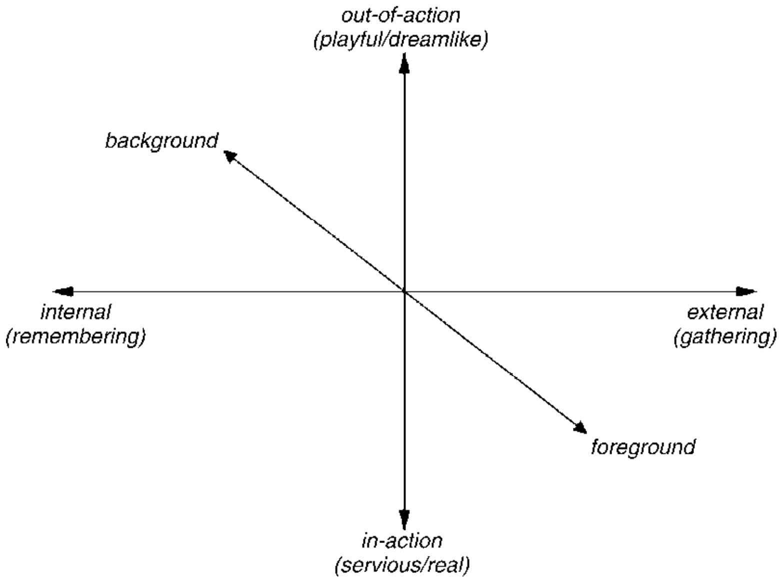


Fig. 7 The resulting general conceptual framework for reflective loupes, showing three dimensions for characterizing reflective practice: (1) internal – external; (2) in-action vs. out-of-action; and (3) background vs. foreground

We gave each group the same design task to complete over the same length of time. They were to work for 20 min, take a break for 10 min, repeat this cycle, and finally journal with a note pad for 5 min to record what they did during their breaks and where their ideas came from. Finally, we interviewed them for approximately 10 min. Interviews were semi-structured, with some straightforward questions and some open-ended questions (besides demographic information):

- Tell us about your design. What is it, what does it do?
- Tell us about your process. How did you come up with this design?
- How did you work together as a team? What roles did each of you play, specifically in the ideation?
- What did you do in the first 10 min, the next 20 min, the last 20 min?
- What did you think about during the structured breaks?
- What did you think about during your journaling time?
- Go through the journals and tell us what you journaled about
- Where did your ideas come from?
- What worked well or not well about the process?
- What did you enjoy about this design task?
- Is there anything you would do differently if you had to do it again?
- How do approach your own research in terms of getting ideas and where have you been successful in getting ideas for your research?
- What do you do when you are stuck in your research or design work? How does it work for you?

We captured the design task on video for all three pairs, and also videotaped the play break for the third pair. Interviews were recorded both on video and in notes taken by the interviewer.

Currently we are coding and processing the data obtained. As a result, we expect to have three in-depth controlled examples of reflection-out-of-action including background information on the subjects, observation of their actual action and resulting artifacts of their design activity. We expect that this data will allow us to make first approximations for how to support ideation-enhancing out-of-action reflective loupes.

4 Reflective Loupes, Conclusion and Future Research

This year's Design Loupes project focused on showing the existence of and obtaining a better understanding of the inner design loupe – the reflective design loupe. Starting out on the hunch that these kind of loupes not only exist, but may also support the individual ideation process, and taking a grounded theory approach, the team decided to design an unstructured iterative and explorative research plan. The objectives of this research were to give evidence of the existence of the phenomenon, to develop a theoretical conceptual framework, to construct a working definition and to conduct experiments, which allow observing and analyzing the reflective loupes in action.

First evidence was provided by means of analyzing idealog files, a comprehensively used design tool (analysis of artifacts). To better understand reflective loupes, we then collected and structured reflective practices from design experts and practitioners by means of a survey. The analysis of this practical input from the 20 experts, in combination with an iterative and continuous framing process, allowed us to conceptually construct the reflective design loupe space. This space appears as a continuum between internal (remembering) vs. external (gathering) thought on the x-axis, in-action (serious/real) vs. out-of-action (playful/imaginative) ideation activities on the y-axis, and background vs. foreground reflection on the z-axis.

Currently we are working to analyze the result of an in-vitro design exercise that was videotaped and accompanied by semi-structured interviews. Results of these three experimental explorations should allow researchers to: (1) set up broad statistical enquiries into reflective practices in actual design processes, using survey and secondary data analyses of broad samples; and (2) create tightly-controlled experiments aimed at uncovering the underlying mechanisms of reflective loupes.

We have found that observing designers' behaviors, and collecting and analyzing the artifacts of their reflective practices, provide insights into the reflective loupe and its role in early design ideation. In the future we will expand upon, and continue to mine the existing data set for richer insights into the benefits of reflection in innovative design. With this as a foundation, we can develop and test more formal hypotheses.

Additionally, the data used in our reflective loupe study may be connected and fed into the analyzer of the d.store currently under development by our partner project team at HPI in Potsdam. The core task will then be to identify digital elements of reflective practices embedded in design communication. Analyzing these jointly with other digital design process artifacts, may, for example, enable us to identify the most creative individuals, as well as systematically introduce support measures and tools to enhance ideation on a group level.

Acknowledgements The authors would like to thank the Hasso Plattner Institute – Stanford Design Thinking Research Program for funding this project. A special thank you goes to the Enterprise Platform and Integration group headed by Prof. Plattner and Dr. Zeier, for a successful and very pleasant second-year collaboration. We also wish to express great appreciation to Mariel Lanas and Tiffany Lau, who contributed greatly to the execution of this research through the Stanford Undergraduate Research Initiative, and who were a joy to work with. We congratulate Dr. Philipp Skogstad, one of the leading researchers on the Stanford side of this project, who graduated in June 2009, and Dr. Matthias Uflacker, from the HPI side of the project, who graduated in January 2011.

References

Brown, T. (2008). Design thinking. *Harvard Business Review*, *86*(6), 84–92.

Cockayne, W. R. (2004). *A Study of the Formation of Innovation Ideas in Informal Networks*. Stanford University.

Cross, N., & Clayburn Cross, A. (1995). Observations of teamwork and social processes in design. *Design Studies*, *16*(2), 143–170.

Dym, C. L., Agogino, A. M., Eris, O., Frey, D. D., & Leifer, L. J. (2006). Engineering design thinking, teaching, and learning. *IEEE Engineering Management Review*, *34*(1), 65–92.

Glaser, B., & Strauss, A. (1967). *Grounded Theory: The Discovery of Grounded Theory*. New York: de Gruyter.

Glaser, B. G., & Strauss, A. L. (2007). *The discovery of grounded theory: Strategies for qualitative research*. Aldine Transaction.

Goldschmidt, G. (1991). The dialectics of sketching. *Creativity research journal*, *4*(2), 123–143.

Kim, J., & Wilemon, D. (2002). Strategic issues in managing innovation's fuzzy front-end. *European Journal of Innovation Management*, *5*(1), 27–39.

Koen, P., Ajamian, G., Burkart, R., Clamen, A., Davidson, J., D'Amore, R., Elkins, C., et al. (2001). Providing clarity and a common language to the" fuzzy front end". *Research-Technology Management*, *44*(2), 46–55.

Merriam-Webster. (2010). *Reflection*. Retrieved December 10, 2010, from http://www.merriam-webster.com/dictionary/reflection

Plattner, H., Meinel, C., & Weinberg, U. (2009). *Design-Thinking*. Finanzbuch Verlag GmbH. Retrieved from http://books.google.com/books?id=9wVSPgAACAAJ

Reinertsen, D. G. (1999). Taking the fuzziness out of the fuzzy front end. *Research-Technology Management*, *42*(6), 25–31.

Schön, D. A. (1983). *The reflective practitioner*. Basic books New York.

Schumpeter, J. A. (2006). *Capitalism, socialism and democracy*. Routledge.

Skogstad, P., Steinert, M., Gumerlock, L., & Leifer, L. (2009). We Need a Universal Design Project Outcome Performance Measurement Metric: A Discussion Based on Empirical Research. In *Proceedings of the 17th International Conference on Engineering Design (ICED'09)* (Vol. 6). Stanford: Design Society.

Skogstad, P. L., Currano, R. M., & Leifer, L. J. (2008). An Experiment in Design Pedagogy Transfer Across Cultures and Disciplines. *life-long learning*, *11*, 12–13.

Skogstad, P. L. S. (2009). *A unified innovation process model for engineering designers and managers*. Stanford University.

Thomke, S., Fujimoto, T., & Research, H. U. G. S. O. B. A. D. O. (2000). The effect of front-loading problem-solving on product development performance. *Journal of Product Innovation Management*, *17*(2), 128–142.

Uflacker, M., & Zeier, A. (2009). A platform for the temporal evaluation of team communication in distributed design environments. In *Computer Supported Cooperative Work in Design, 2009. CSCWD 2009. 13th International Conference on* (pp. 338–343). IEEE.

Van de Ven, A. H. (1999). *The innovation journey*. Oxford University Press, USA.

Towards Next-Generation Design Thinking II: Virtual Multi-user Software Prototypes

Gregor Gabrysiak, Holger Giese*, and Andreas Seibel

Abstract Design thinking benefits from the usage of tangible prototypes to communicate, validate and explore insights and design ideas. For domains dealing with immaterial objects and intangible concepts, however, prototyping is usually not feasible. During the first year of the *Scenario-Based Prototyping* project we conceptualized an approach for creating tangible prototypes of multi-user software systems based on executable formal models. Through simulation and animation, these models can then be experienced and evaluated by end users. In this chapter, we further elaborate on the implementation of our approach and discuss results of an evaluation comparing the usability of our approach with traditional formal and informal modeling approaches.

1 Introduction

Design thinking is a process that is highly interactive and incremental. People with different backgrounds and experiences are crucial to drive it. To create a useful innovative product, it is crucial to understand the real needs of the end users. Therefore, the design thinkers have to gather insights about the end users and their domain. More specifically, *what* do end users do, *how* do they do it and *why* is it done. After design thinkers have gathered assumptions about the end users, it is imperative to validate what was captured and the subsequent design ideas. Producing tangible prototypes has proven to be an essential tool for establishing a common understanding within heterogeneous design teams, but also with and among end

G. Gabrysiak • H. Giese (✉) • A. Seibel
System Analysis and Modeling Group, Hasso-Plattner-Institute for IT Systems Engineering, University of Potsdam, Prof.-Dr.-Helmert-Str. 2–3, 14482 Potsdam, Germany
e-mail: gregor.gabrysiak@hpi.uni-potsdam.de; holger.giese@hpi.uni-potsdam.de; andreas.seibel@hpi.uni-potsdam.de

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_7,

users, about what the design thinkers learned. By using prototypes to elicit or provoke feedback about assumptions gathered beforehand or manifestations of design ideas [17], it becomes possible to validate the assumptions and the resulting design choices.

Tangible prototypes support the design thinkers' need to gather feedback from end users to validate the underlying concepts they embody. In typical design thinking projects these prototypes can be derived straightforwardly which allows to validate concepts quite early, inexpensive, and often. This allows design thinkers to *fail early* and *fail often*.

As pointed out by Andriole [4], prototyping is very effective also for software systems, since it can allow end users to experience something *tangible* and provide feedback about it. Bäumer et al. identify three different kinds of prototypes for software systems [5]. While *Explorative Prototypes* are commonly used at the beginning of a project to test people's reactions on new concepts, *Experimental Prototypes* are produced to evaluate whether a concept fulfills the end users' expectations. On the other hand, *Evolutionary Prototypes* combine both approaches. Through multiple iterations, the prototype matures till the final prototype can be considered as the final result.

Typical design thinking software projects consequently employ prototypes of graphical user interfaces (GUI) where end users are in the center of attention [26]. However, for complex multi-user software systems this is not the case. Since GUI prototypes usually only represent a single user's view on the system, their applicability is limited to validating the design of this individual view (*how* can I do *what*).

Thus, the validation of the underlying rationale (*why* do I do it) is not feasible for all involved end user. Looking at the complete system, however, building a prototype that captures the whole underlying processes and data is prohibitive expensive. These high costs associated with producing such prototypes of complex software systems imply that only few prototypes would be economically feasible, if at all. In software engineering for complex multi-user software systems, end user needs and potential solutions are thus today described at a high level of abstraction using software engineering models such as processes and scenarios, but no prototypes. Still, design thinking strongly depends on insights about the design challenge gathered directly from end users using prototypes. Prototyping permits to obtain further insight due to end users' feedback to end up with a suitable solution for the *right* problem [23]. Problems arise especially when designing solutions involving multiple users as each end users' understanding of the problem and the implicated individual needs may not only be different but even conflicting. Therefore, besides eliciting insights from all end users, it is also required to resolve these conflicts by creating a common understanding of the problem domain, which can be quite complicated and costly.

In addition, not all members of the design team might be familiar with the employed software engineering models and, thus, they can hardly contribute to them in a direct manner. Until today, multiple iterations with tangible prototypes are not feasible for complex multi-user software systems. Thus, design thinking cannot realize its full potential when addressing such problems.

In this chapter, our approach for prototyping multi-user software processes scenario-based in a tangible manner is presented as well as a brief explanation of its prototypical implementation, our *Scenario-Based Tangible Prototypes Environment* (SceB-TaPE[1]). The approach is discussed in Sect. 2. Then, preliminary evaluation results are presented in Sect. 3. Afterwards, Sect. 4 discusses related work of our approach before this chapter closes with a summary and an outline for future work.

2 Approach

Figure 1 illustrates a sequence of activities, which represents our methodology of gathering insights and needs of end users from initial interviews to subsequent validations via virtual prototyping of software engineering models.

Initially, end users are interviewed by design thinkers to gather insights and needs (1). These insights are subsequently externalized into software engineering models by design thinkers (2).[2] Formal models are necessary to cope with the

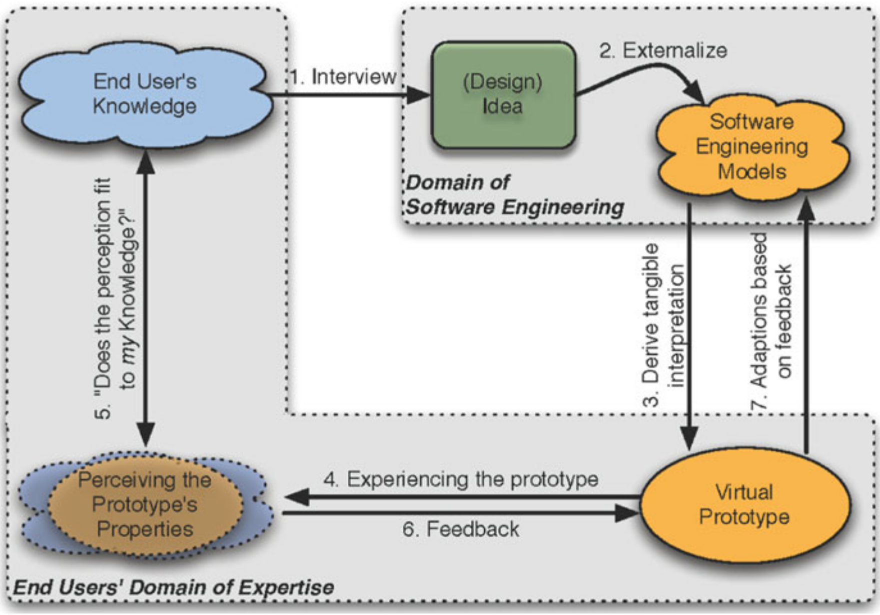


Fig. 1 Our methodology to gather and validate end users' insights and needs

[1]SceB-TaPE is part of the MDE-Lab Tools, *http://mdelab.org/SceB-TaPE/*

[2]We expect that in a multidisciplinary team of working on a typical design thinking software project, there is at least one member who is familiar with software engineering.

inherent complexity of multi-user software systems. While such models support the management of large sets of requirements, using a word document of several hundred pages is not as simple to maintain.

To validate end users' insights and needs we use SceB-TaPE to automatically derive a virtual prototype (3), which is a tangible interpretation of the underlying software engineering models. These virtual prototypes are closer to the domain of expertise of end users and, thus, can be experienced by end users directly (4). During experiencing the virtual prototype, end users perceive concepts embodied in the virtual prototype and compare them to their corresponding understanding of their domain (5). This allows them to appropriately provide feedback about concepts within their domain of expertise (6). Based on the provided feedback during virtual prototyping, the software engineering models can be adapted (7). This sequence can be systematically repeated until there is a common understanding of the problem domain.

In the following, we will explain in more detail the similarity between prototypes and models (software engineering models) (Sect. 2.1), the kind of software engineering models we support (Sect. 2.2), our methodology from a designer thinker's perspective (Sect. 2.3), and from an end user's perspective (Sect. 2.4).

2.1 Prototypes and Models

As mentioned before, design thinkers heavily rely on prototypes to share and iterate their design ideas. It is through prototypes that designers can externalize these ideas to allow the world to speak back [17]. As we have outlined in [10], tangible prototypes are the means of choice when it comes to *failing early* and *failing often* to validate insights and explore design alternatives. By enabling end users to experience and judge a manifestation of a design idea they can evaluate, judge, and provide feedback about. Usually, these manifestations are incomplete, since the construction of a complete prototype, which mimics the envisioned product in every way, is too expensive to be feasible throughout multiple iteration cycles. It is important for the end users to be able to interpret the relevant properties the prototype embodies correctly (cf. Fig. 2). Otherwise they might complain about the unrealistic *size* of the prototype without noticing the relevant property *weight*.

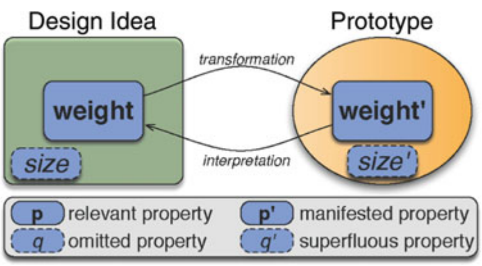


Fig. 2 Designer thinkers can communicate and iterate design ideas using prototypes if the end users can interpret these correctly (adapted from [10])

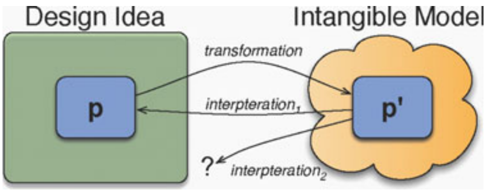


Fig. 3 While modeling experts can share and iterate their ideas through models, everybody unable to read and understand their specification is left out of the loop (adapted from [10])

In domains dealing with intangible concepts or immaterial objects, the ability to create tangible representations to communicate and share ideas is restricted. There are several industry standards of capturing knowledge in certain domains, e.g., the UML[3] or BPMN.[4] While, especially, software engineers have the necessary knowledge to externalize, share and iterate their ideas among each other, end users are usually left out of the loop. This is issue sketched in Fig. 3.

The correct interpretation $interpretation_1$, is shared by software engineers, while end users are left wondering in $interpretation_2$. It is important to note, that such formal models can be considered as prototypes for software engineers. Consequently, restricted only by the modeling notation they use, they can filter properties, evaluate designs and iterate solutions. To enable end users to provide valuable feedback about these ideas requires them to understand the captured knowledge and ideas. This, in turn, introduces overhead for the software engineers. First of all, they have to explain the modeling notation, the used modeling elements and also the implications of the decisions they have already included into their model. Secondly, after the end users have understood the presented model, their feedback has to be translated into changes to the model. Unfortunately, formal modeling languages are usually quite restrictive as far as what they allow the designer to express. Thus, this translation process is quite error-prone.

As mentioned before, this tension between the software engineers who prototype their design ideas using intangible models and the end users who want to be sure that the result is appropriate for them is commonly dealt by graphical user interface (GUI) prototypes. These prototypes allow the end users to comment on the individually perceived usefulness (usability) of the proposed interface. However, the underlying rationale for the design cannot be fully addressed because the interaction of end users is not considered.

[3]Unified Modeling Language, *http://www.omg.org/spec/UML/*

[4]Business Process Modeling Notation, *http://www.omg.org/spec/BPMN/*

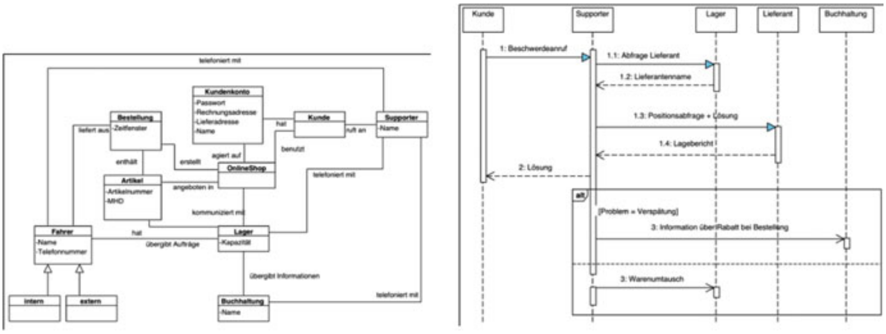


Fig. 4 A class diagram describing domain concepts (*left*) and a specific scenario (*right*) captured by designers in our evaluation

2.2 Software Engineering Models for Multi-user Scenarios

In the domain of software engineering, the information gathered from end users that is subsequently virtually prototyped are actions of end users and interactions between them, embedded into specific scenarios. Information about these scenarios can be used to design a software system that supports the interactions between end users.

In an evaluation with students (cf. Sect. 3), we asked software engineering students to conduct interviews with end users about a similar problem. We had three kinds of groups of students with each using a different technique to externalize their insights. From a specific kind of group, we did not expect them to use certain software engineering models but they should use a specific UML tool. Figure 4 shows the externalized software engineering models of one of these groups of students. They used UML class diagrams to specify domain concepts such as roles, and other concepts that are relevant to the role of the interviews end user. Furthermore, they have specified UML sequence diagrams to define scenarios they gathered during the interview. All groups that have to use UML used similar diagrams.

The software engineering models that we support to automatically derive virtual prototypes are quite similar but to some extent more formal. End users' concepts are defined in a formal ontology, which contains conceptual entities and potential relationships in between (e.g., emails have one sender and potentially multiple receivers). This ontology is similar to UML class diagrams. Interactions between end users are clustered in scenarios. Each scenario can be considered as a set of sequence diagrams where each interaction is related to a behavioral specification that defines the condition of the interaction as well as the side effect.

2.3 *From the Design Thinker's Perspective*

Initially, design thinkers conduct interviews with end users. In these interviews, the design thinkers elicit roles that act as abstractions of end users, scenarios with all actions and interactions between these roles, and finally concepts that occur in all scenarios. Based on this gathered insights, a design thinker creates an initial software engineering model which is the externalization of the conducted interviews.

The initially externalized software engineering model is sufficient to apply SceBTaPE to conduct validation sessions for individually identified scenarios. Subsequent validation sessions are necessary because we have to assume that the initially gathered insights are neither complete nor completely correct. Thus, the goal of the validation session is to maximize the agreed upon behavior of identified scenarios and further resolve conflicting behavior.

To start a validation session, a design thinker starts a virtual prototype for a selected scenario using SceB-TaPE. Afterwards, end users can be invited to participate by enacting specific roles in this scenario. Invited end users can remotely connect to the validation session via web browser from any workplace. Figure 5 shows the principle of virtual prototyping with SceB-TaPE.

The core of a virtual prototype is a simulator which simulates the behavior of specific scenario defined in the software engineering model. The simulator uses the existing concepts and already specified behavior in the selected scenario. Any invited end user is connected via a GUI to the simulator. The GUI is tailored to the domain of expertise of participating end user (cf. Sect. 2.4). The GUI can visualize behavior (play-out) and it provides the ability to trigger already captured behavior or create new behavior (play-in). Thus, the end users are able interact with the virtual prototype.

To increase the efficiency of the validation sessions, the simulator provides two particular simulation strategies. The elicitation of priorly unknown behavior can be

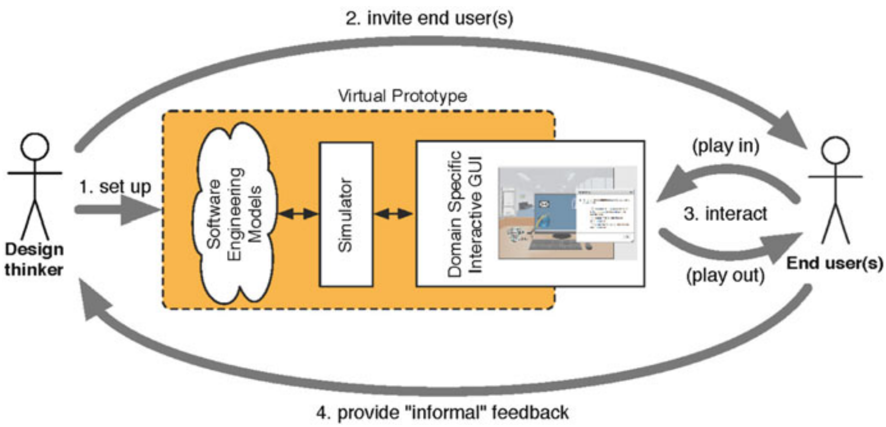


Fig. 5 Conducting a validation session using a virtual Prototype

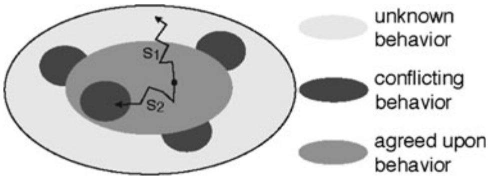


Fig. 6 Kinds of behavior that is explored during validation sessions

achieved, for example, by starting from a specified initial state of the process, replaying the already agreed upon behavior and then requesting end users to continue the simulation (see trace s_1 of Fig. 6). During the replay the behavior is validated. When the yet unexplored behavior begins, the participants *play in* behavior that was not observed yet. This might also lead to the observation of conflicting behavior.

Having identified a conflict between two stakeholders who gave, e.g., inconsistent information, it is necessary to gain insights into why they differ or whether the conflict is simply due to a misunderstanding. For example, leading an end user directly to a conflict during the simulation in order to resolve it, we can enforce that the simulation provides the necessary information (see trace s_2 of Fig. 6).

The common way of dealing with these conflicts is to infer the most likely behavior [15]. If we consider two roles a and b (with specified or played-in partial trace sets A and B, respectively) which have provided different feedback, the typical strategy to interfere behavior is to restrict the system to the intersection of both perspectives (similar to $A \cap B$ concerning the overlap of the perspectives). E.g., while the behavior differed in the way a and b prepared a document, they implicitly agreed on the necessity of sending the result via email to their boss. This intersection is suitable if the design thinkers want to get the agreed upon behavior, however, to learn how things are seen *differently*, it is necessary to discuss possible conflicts as conflicts may also indicate possible alternatives for the design. To understand these differences and their implications, it helps to gather feedback from a on what b does differently and the other way around to entangle the conflicting perspectives. By presenting behavior in B that is not covered by A (similar to $B \setminus A$ concerning the overlap of the perspectives), we enable that a understands what b proposes differently and to judge whether the difference is justified.

After each validation session the design thinkers have to potentially update the software engineering models. This is done automatically for actions as well as interactions but not for adding, removing or updating concepts. This, feedback is provided interactively and further manually integrated by the design thinkers. Because conceptual changes potentially invalidate existing behavioral specifications we have developed a mechanism to automatically update all behavioral specifications that refer to changed or deleted concepts.

In the end, when the design thinkers have conducted sufficient validation sessions there are multiple traces for individual scenarios that can be synthesized into a common scenario reflecting all possible alternatives. The resulting software engineering model can then be used as a stable, commonly agreed upon foundation to ideate and explore design ideas.

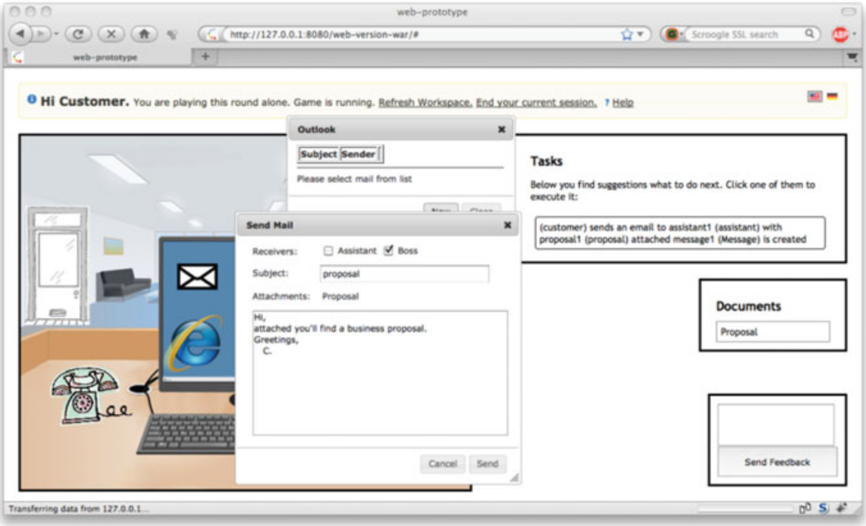


Fig. 7 As part of the multi-user process, the *Customer* sends a *Proposal* to the *Boss*

2.4 From the End User's Perspective

End users interact with SceB-TaPE when designers decide to conduct a validation or elicitation session via using a virtual prototype. Based on the kinds of end users (e.g., the role in the company) different visualizations for the virtual prototypes can be used as interface between end users and the virtual prototype. While abstract models are used to capture and specify concepts and procedures the end users are not only familiar with, but even experts in, the unknown modeling notations and concepts hinder them to validate, correct or even expand the specified content. To tackle this problem, we prototyped a possibility to create a tangible representation for models of multi-user processes in the first year of this project [11]. By presenting end users models in their domain of expertise, we ease their understanding of what is essentially a prototyped description of their domain of expertise.

When end users are invited, they can remotely log into a validation session by choosing one of the identified roles. Roles that are not actively enacted by end users are automatically simulated in the background by SceB-TaPE. This simulation is done based on existing interactions captured either in previous elicitation and validation sessions or by initial interviews. Based on their everyday experience, end users can interact with each other, e.g., via email (Fig. 7) or signatures on contracts (Fig. 8).

As part of investigations during the second year of this project, we also worked on different visualizations for different groups of end users. While end users participating in multi-user processes[5] can validate insights first hand in detailed

[5]*Normal operators* according to Alexander [2].

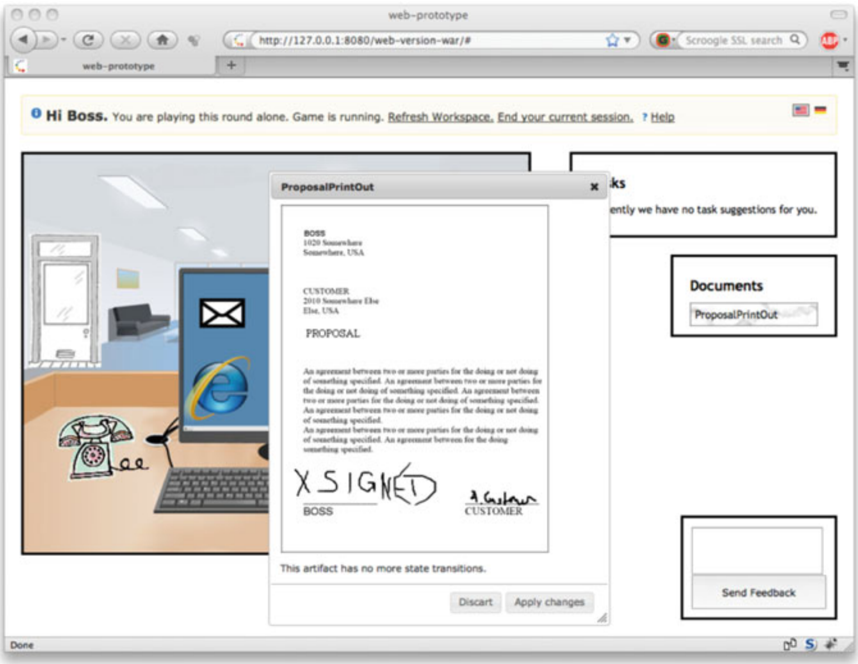


Fig. 8 End users cannot only interact with each other, but also with documents and artifacts, e.g., by signing them

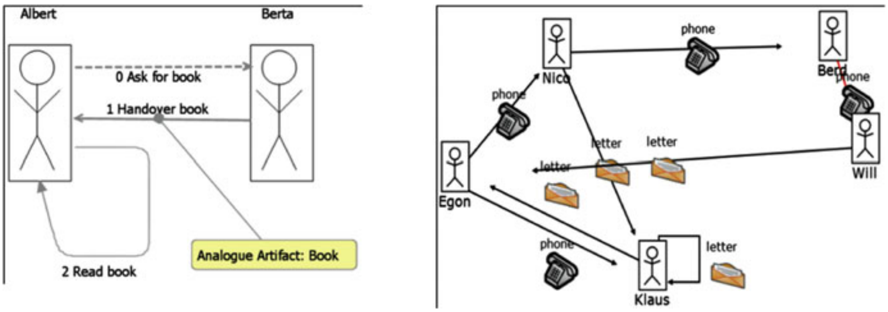


Fig. 9 While end users can only validate interactions they are participating in (*left*), managers need more holistic views on the information gathered (*right*)

simulations and animations, supervisors and managers might be overwhelmed by the amount of details. Thus, they require another view on these models that shows fewer details and is broader. Instead of focusing on one participant only, they are interested in the holistic picture of what is actually done. Figure 9 presents two different broader views on insights gathered. While the view on the left focuses on sequences of interactions, the other one illustrates which end users interact.

Based on the end users' feedback about how they experience their part within the multi-user process, the designer evaluates whether the gathered assumptions or the prototyped ideas are suitable.

3 Evaluation Results

To evaluate the feasibility of our approach, we set up an experiment comparing our prototypical implementation (*SceB-TaPE*) with a formal UML modeling tool (*Formal UML Tool*) and an informal modeling approach based on paper and whiteboards (*Paper*). All groups had to elicit information by interviewing an end user, by modeling their insights and by presenting these insights to the end user. The hypotheses we evaluated are presented in Sect. 3.1. The experiment setup is described in Sect. 3.2, while Sects. 3.3 and 3.4 discuss our preliminary results.

3.1 Hypotheses

To evaluate the effect of SceB-TaPE, we formulated the following hypotheses:

Hypothesis 1. *Using our approach, the end users and designers develop a better understanding of what is presented to them.*

While Hypothesis 1 includes an improved common understanding among designers, it is important for end users to understand the designer's specifications without having to explicitly ask for explanations along the way.

It is important for our evaluation to set up realistic experiments during which designers or software engineers use SceB-TaPE to model and validate findings and ideas. However, access to end users suitable for such experiments is limited, mainly due to the required knowledge only people who participate in a complex multi-user process possess. Only with them, our approach can be tested realistically. Thus, we hypothesized that we would be able to teach people the necessary knowledge:

Hypothesis 2. *We can enable people to gain domain knowledge in form of a multi-user process by playing multiple times through process scenarios with them.*

3.2 Experiment Setup

In order to evaluate our approach of supporting the elicitation and validation of how end users within a complex multi-user process collaborate and achieve their common goal, we set up an experiment within an undergraduate modeling lecture. The students worked in nine groups of four students. They were presented with the assignment to gather information from a support agent who works for an online supermarket. While their lecture assignment specified how goods from this online supermarket are delivered successfully, the potential problems that could be encountered were not considered. Consequently, we presented the students with an end user whose role as participant in the presented company seemed to have been forgotten.

To simulate these support agents in a realistic manner, we cast graduate students from other faculties to enact the role of a corresponding end user within this online supermarket. Details on how we prepared them can be found in [13]. For all tasks, the students had four hours in total. They had to interview the end user (50 min), specify their findings (100 min), and validate their results with the end user (50 min). In between, questionnaires were handed out to the 36 students as well as to the 9 end users. All questionnaires used a 5-point Likert scale (1 being in *agreement*, 5 being in *disagreement*) and free text questions.

3.3 Preliminary Results

3.3.1 Interviews

The interviews between the modeling students and the enacted end users were quite successful. This phase of the experiment was identical for all groups since the students were asked to take notes. Thus, no modeling was involved. The students were told to concentrate solely on the elicitation of the end user's as-is workflow. As it turned out, the duration of the interviews ranged from 35 min up to the maximum of 50 min at which point we ended the interview. Generally, the length depended on how soon the students thought they obtained enough information from the end user. More details about the quite authentic interviews can be found in [13].

3.3.2 Specification of Findings

After the interviews, the students were asked to specify their findings using the approach assigned to them. While all students generally agreed that their results are logical, correct and understandable, we see that students modeling informally (Paper) were significantly more confident that end users would be able to understand their results (Q3 in Fig. 10, $p = 0.05$).

According to the answers shown in Fig. 11, all students were generally quite satisfied with the results they produced during the specification phase. As anybody who ever used a formal modeling tool will understand, we were surprised to see that students relying on the formal modeling tool had slightly more fun than the other groups. Interestingly, students using SceB-TaPE agreed that the 100 min for

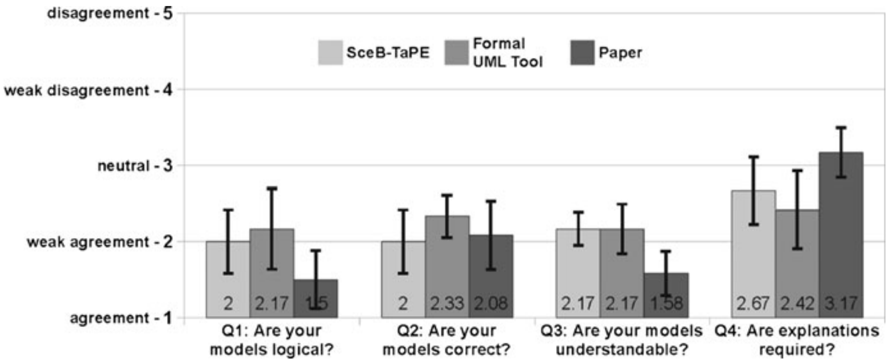


Fig. 10 After the specification phase, the modeling students were quite confident about the quality and understandability of their results (n = 12, p = 0.05)

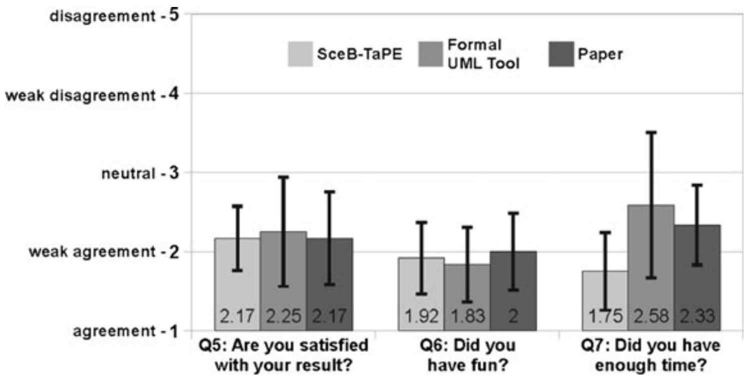


Fig. 11 Questions about the modeling activities of the students (n = 12)

specifying their findings were sufficient. The usage of a formal modeling tool proved to be quite time consuming in comparison. When asked how much more time they felt was necessary, only one of the students using our approach asked for 20 more minutes. While four students relying on informal representations felt they needed 36 min more on average, 50% of the students employing the formal modeling tool stated that they required 80 min more on average.

To evaluate Hypothesis 1, we asked the students *after the modeling phase* about how they perceived the common agreement within their individual groups (cf. Fig. 12). After the 100 min of discussion and specification, 24 students (67%) perceived an improved common agreement of their team.[6] The difference in the individual modeling approaches is illustrated in Fig. 13. Students using SceB-TaPE

[6]It was rated as being the same by 11 students (30.5%). However, one student rated the agreement reached afterwards even worse while the rest of the team did not perceive a change.

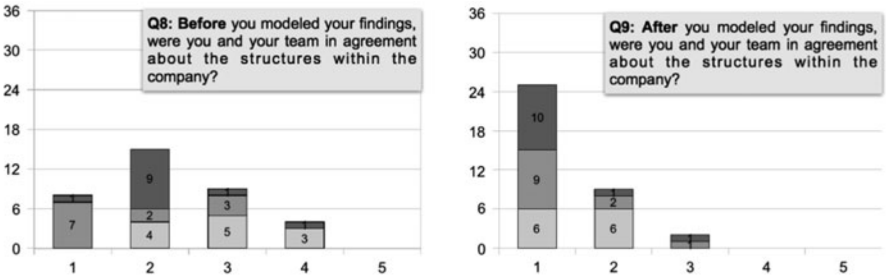
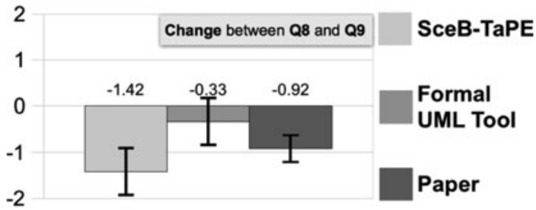


Fig. 12 The students were asked retrospectively about the agreement within their group before (*left*) and after (*right*) the modeling (number of students per rating, n = 36)

Fig. 13 Most of the students perceived that their groups more commonly agreed upon their findings after discussion and specification (n = 12, p = 0.05)



agreed significantly ($p < 0.01$) more positively that their team had improved their common understanding compared to groups relying on a formal modeling approach.

3.3.3 Validation of Findings

The end user's ability to understand the presented specifications is indirectly measured through question 10 and 11 (Fig. 14[7]). The students presenting their results encountered significantly more questions when they relied on the formal modeling tool ($p < 0.05$). Also, the three different modeling approaches all varied significantly concerning the amount of comprehension related questions the students had to answer ($p < 0.02$). While the presentation of the formal UML models raised many questions, the other two approaches were more understandable.

When we asked the students whether the end users provide corrections (Q12 in Fig. 14), the amount the differences were not significant. However, we were astonished to see that the students relying on the formal modeling tool tended to blame themselves for the errors that were made during the specification phase. Our

[7]These results are based on the individual perception of the individual students. The video coding of the experiments still needs to be finished to have an objective comparison for Q10, Q11, and Q12 in Fig. 14.

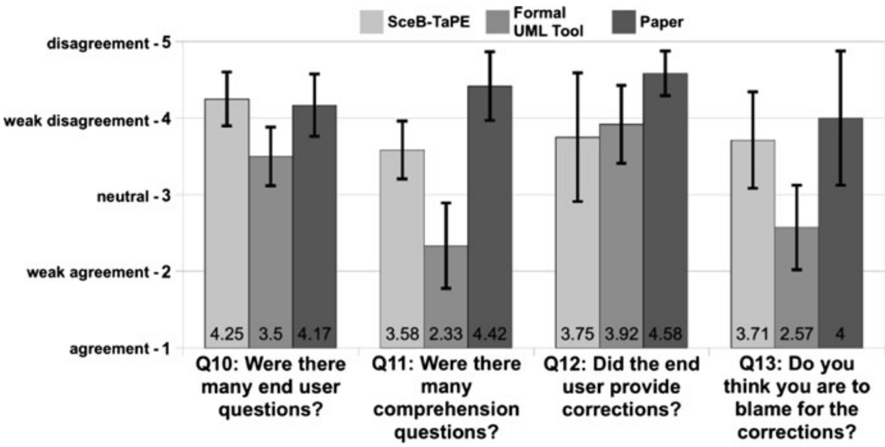


Fig. 14 After the validation, we asked the students about the end users' feedback (n = 12)

working thesis for this phenomenon is based on the inability to *preserve ambiguity* when specifying findings using a formal model. Multiple times the students had to decide to, e.g., either model an interaction as synchronous or asynchronous, since there is no *in between* or *TBD*[8] in a formal specification. So, by deciding for either option without sufficient information, they were at risk to model it wrong. While this might provoke end user feedback, the end users have to understand the models to be able to correct them.

3.4 Discussion of Results

Due to the nature of the lecture which we used to evaluate SceB-TaPE, there were 12 students for each specification and validation method, while we had to rely on only three end users per method. Thus, the results are sufficient concerning the modeler's point of view on our approach, while the end user's perspective can only be considered as exploratory.

For Hypothesis 1, our preliminary results indicate that students using our approach achieved a much better common understanding of what they heard from the end user compared to the other approaches. The end users exposed to SceBTaPE all agreed ($\bar{x} = 1, s^2 = 0$) that what they ended up with after the validation session was created by the students as well as the end user. While end users presented with informal specifications felt similarly involved ($\bar{x} = 1.66, s^2 = 0.58$), groups exposing the end users to UML models did not involve them as much ($\bar{x} = 3, s^2 = 2$). In summary for this hypothesis, our preliminary results already yield strong indications

[8]*To be determined.*

that this hypothesis holds in comparison to the formal modeling approach, although not compared to the informal specifications. The results of Q13 in Fig. 14 give us a strong indication that the ability to preserve ambiguity in models is quite important.

The ability to specify insights without ambiguity leaves the designers more confident in their models. Without any distinction on how sure a part of the model is, looking back at it enforces the effect that the decision was correct, as for the rest of the model. This also explains why students using the UML tool were in agreement about their model but still felt rather guilty about the errors that were corrected. While the students relying on the UML modeling tool already had experience in using it, the students using SceB-TaPE were initially introduced to it during the experiment. Hence, we expect better results in a setting where the designers have used it before.

For Hypothesis 2, we found that it is possible to teach multiple people the required multi-user process knowledge to have access to as many end users as necessary. More details about the corresponding results and how the end users were prepared and instructed can be found in [13].

4 Related Work

The problem of how to communicate with end users either to elicit information or validate insights and ideas has been explored for quite some time. While structured interviews seem to be most suitable to elicit information from end users [8], the validation strongly depends on how formally these insights were specified and presented afterwards. For the area of requirements engineering, Al-Rawas and Easterbrook report that practitioners rely on (informal) natural language to convey their findings to end users either by annotating their formal models or by solely specifying their results in natural language [1]. This inability to communicate formally specified results and ideas effectively might also be one of the reasons of the survey results of Neill and Laplante [18], who found that 51% of practitioners create their specifications using informal approaches. Approaches trying to close the gap between informal and formal modeling also exist. Usually, these approaches allow the designer to model informally. Afterwards, a formal specification is created, e.g., via means of pattern recognition and transformation [3]. Our approach takes the opposite direction by creating a tangible, rather informal representation of formal models [12].

As Andriole argued, prototyping already be used during requirements elicitation to enable users to provide feedback to something tangible [4]. Brown defines prototypes as “anything tangible that lets us explore an idea, evaluate it and push it forward” [7]. In case of *things you cannot pick up*, he proposes the usage of scenario descriptions as a way to evaluate ideas. On the other hand, some approaches rely on tangible artifacts representing non-material objects to improve the communication and to create a common understanding, e.g., CRC cards [6] or even recent approaches such as Tangible Business Process Modeling [9]. While these approaches are important, they require all participants to be in one location at the same time. Additionally, these approaches usually require the software engineer to formally specify the findings later on manually, which then in turn can lead to errors during the transformation.

As explained in Sect. 2.1, end users need to be able to understand the content of the specifications. To enable them to validate their requirements, the area of *Requirements Animation* emerged [25]. In [14], Gemino presents an empirical comparison of requirements animation and narration. The preliminary findings include that the presentation of information can be as important as the content itself.

Sellen et al. presented similar results concerning the format of prototypes and how this influences the perception of different groups [20]. For our approach, the two main groups of users that were taken into account are designers and end users. However, since end users in different domains or even companies perceive tangibility of virtual prototypes differently, it is possible to adjust the domain-specific GUI on top of the simulation accordingly (cf. Fig. 5). Other approaches of requirements animation aim at the software engineers rather than the end users, e.g., by simply visualizing the state of the simulation of requirements in their formal notation [21]. In [19], domain specific control panels visualize the state of the simulation in the stakeholders' domain of expertise, thereby easing the stakeholders' understanding. However, it focuses rather on single user control systems by monitoring the input of stakeholders without allowing them to extend the model during the simulation.

Harel and Marelly [16] present an approach to not only simulate (*play-out*) a formal requirements model, but also to enrich it with new details (*play-in*) Besides the formal model in form of live sequence charts, also a prototypical GUI of the software system can be used to animate the simulation as well as capturing the user's feedback to enrich the formal models by playing in new additional scenarios. Generally, scenario-based approaches such as [22] or [24] emphasize on synthesizing requirements, either from multiple play-in sessions or records of valid system behavior.

However, while partially using requirements animation, these approaches are aimed rather at requirements engineers than at end users. Both, formal requirements models and prototyping offer many advantages. Still, existing work either focuses on the one or the other. Existing requirements modeling approaches relying on animation and/or play-in result in severe limitations for the interactions with stakeholders. Either no focus on particular stakeholders can be set (cf. [25]) or the approaches are limited to animation only.

5 Summary and Future Work

During the first year of the project, we investigated possibilities of how the validation of insights could benefit from more direct end user involvement in the design of multi-user software systems. The concept we prototyped relied on executable formal models combined with interactive animation. For other end users in other domains, different visualizations are necessary to animate the concepts of these domains correspondingly to enable their end users to understand them. Allowing end users to directly provide feedback about the model during such a simulation can be used iteratively to evolve the underlying model till all end users and designers share a common understanding on the validated model. In the second year, we enriched the capabilities of the SceB-TaPE and its simulator.

Most importantly, everything that is observed from end users is captured in formal models and can immediately be synthesized heuristically to be reused in succeeding simulation sessions. For the end users, new visualizations have been explored and created. Also, SceB-TaPE can now be adjusted to new domains quite easily. As for the designers, SceB-TaPE now includes different visualizations, e.g., for simulation trace management and to provide an overview of which parts of the multi-user process have already been covered.

For the forthcoming, third year of our project, we will focus on exploration strategies to guide end users through the simulation in a more goal-oriented fashion. By executing the models and monitoring their effects, possible conflicts can be spotted. Then, if a corresponding end user joins the simulation, the identified conflict can be resolved by guiding the simulation and the user to the corresponding situation in which the conflict occurs. By choosing a certain path through the process execution, we should be able to support the end users in resolving conflicts quicker than before. Overall, through such a guided simulation, the exploration and validation of *who* does *what* and *why* should become faster, since the designers can actively set the priorities for the simulation instead of the participating end users.

Also, our implementation SceB-TaPE will be enriched with new features along the way, such as increased flexibility when modeling insights, and an interactive visualization for at least one other domain on top of the simulation to assess the effort necessary to employ SceB-TaPE in other domains.

Acknowledgements The authors are grateful for the input of Alexander Renneberg (D-LABS GmbH), Nico Rehwaldt, Alexander Lüders, Henrik Steudel, Stefan Kleff, Stefan Richter, Ralf Teusner, Christoph Kühnl and Jonathan A. Edelman.

References

1. A. Al-Rawas and S. Easterbrook. Communication Problems in Requirements Engineering: A Field Study. In *Proceedings of the First Westminster Conference on Professional Awareness in Software Engineering*. Royal Society, London, February 1996.
2. I. F. Alexander. A Taxonomy of Stakeholders: Human Roles in System Development. *International Journal of Technology and Human Interaction*, 1(1):23 – 59, 2005.
3. A. Anaby-Tavor, D. Amid, A. Fisher, H. Ossher, R. Bellamy, M. Callery, M. Desmond, S. Krasikov, T. Roth, I. Simmonds, and J. de Vries. An algorithm for identifying the abstract syntax of graph-based diagrams. In *VLHCC'09: Proceedings of the 2009 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 193–196, Washington, DC, USA, 2009. IEEE Computer Society.

4. S. Andriole. Fast, cheap requirements: Prototype, or else! *IEEE Software*, 11(2):85–87, 1994.
5. D. Bäumer, W. R. Bischofberger, H. Lichter, and H. Züllighoven. User interface prototyping—concepts, tools, and experience. In *ICSE'96: Proceedings of the 18th international conference on Software engineering*, pages 532–541, Washington, DC, USA, 1996. IEEE Computer Society.
6. R. Biddle, J. Noble, and E. Tempero. Reflections on CRC cards and OO design. In *CRPIT'02: Proceedings of the Fortieth International Conference on Tools Pacific*, pages 201–205, Darlinghurst, Australia, Australia, 2002. Australian Computer Society, Inc.
7. T. Brown. *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*. HarperBusiness, September 2009.
8. A. Davis, O. Dieste, A. Hickey, N. Juristo, and A. M. Moreno. Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review. *Requirements Engineering, IEEE International Conference on*, 0:179–188, 2006.
9. J. Edelman, A. Großkopf, and M. Weske. Tangible Business Process Modeling: A New Approach. In *Proceedings of the 17th International Conference on Engineering Design, ICED'09*. STANFORD UNIVERSITY, STANFORD, CA, USA, August 2009.
10. G. Gabrysiak, J. A. Edelman, H. Giese, and A. Seibel. How tangible can virtual prototypes be? In *Proceedings of the 8th Design Thinking Research Symposium*, 2010.
11. G. Gabrysiak, H. Giese, and A. Seibel. Towards Next Generation Design Thinking: Scenario-Based Prototyping for Designing Complex Software Systems with Multiple Users. In H. Plattner, C. Meinel, and L. Leifer (editors), *Design Thinking – Understand, Improve, Apply*. Springer-Verlag, 2010.
12. G. Gabrysiak, H. Giese, and A. Seibel. Using Ontologies for Flexibly Specifying Multi-User Processes. In *Proc. of FlexiTools Workshop at ICSE 2010, Cape Town, South Africa*, 2 May 2010.
13. G. Gabrysiak, H. Giese, A. Seibel, and S. Neumann. Teaching requirements engineering with virtual stakeholders without software engineering knowledge. In *Requirements Engineering Education and Training, 2010. 5th International Workshop on*, 2010.
14. A. Gemino. Empirical comparisons of animation and narration in requirements validation. *Requir. Eng.*, 9(3):153–168, 2004.
15. D. Harel, H. Kugler, R. Marelly, and A. Pnueli. Smart play-out of behavioral requirements. In *FMCAD'02: Proceedings of the 4th International Conference on Formal Methods in Computer-Aided Design*, pages 378–398, London, UK, 2002. Springer-Verlag.
16. D. Harel and R. Marelly. *Come, Let's Play: Scenario-Based Programming Using LSC's and the Play-Engine*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.
17. Y.-K. Lim, E. Stolterman, and J. Tenenberg. The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Trans. Comput.-Hum. Interact.*, 15 (2):1–27, 2008.
18. C. J. Neill and P. A. Laplante. Requirements Engineering: The State of the Practice. *IEEE Software*, 20(6):40–45, 2003.
19. C. Ponsard, N. Balych, P. Massonet, J. Vanderdonckt, and A. van Lamsweerde. Goal-Oriented Design of Domain Control Panels. In S. W. Gilroy and M. D. Harrison, editors, *DSV-IS*, volume 3941 of *Lecture Notes in Computer Science*, pages 249–260. Springer, 2005.
20. K. M. Sellen, M. A. Massimi, D. M. Lottridge, K. N. Truong, and S. A. Bittle. The people-prototype problem: understanding the interaction between prototype format and user group. In *CHI'09: Proceedings of the 27th international conference on Human factors in computing systems*, pages 635–638, New York, NY, USA, 2009. ACM.
21. C. Seybold, S. Meier, and M. Glinz. Evolution of requirements models by simulation. *Principles of Software Evolution, International Workshop on*, 0:43–48, 2004.
22. C. Seybold, S. Meier, and M. Glinz. Scenario-driven modeling and validation of requirements models. In *SCESM'06: Proceedings of the 2006 international workshop on Scenarios and state machines: models, algorithms, and tools*, pages 83–89, New York, NY, USA, 2006. ACM.

23. M. Tohidi, W. Buxton, R. Baecker, and A. Sellen. Getting the Right Design and the Design Right: Testing Many Is Better Than One. In *CHI'06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 1243–1252, New York, NY, USA, 2006. ACM.
24. S. Uchitel, G. Brunet, and M. Chechik. Synthesis of Partial Behavior Models from Properties and Scenarios. *IEEE Transactions on Software Engineering*, 35(3):384–406, 2009.
25. H. T. Van, A. van Lamsweerde, P. Massonet, and C. Ponsard. Goal-oriented requirements animation. *Requirements Engineering, IEEE International Conference on*, 0:218–228, 2004.
26. T. Winograd, editor. *Bringing design to software*. ACM, New York, NY, USA, 1996.

Parallel Prototyping Leads to Better Design Results, More Divergence, and Increased Self-efficacy

Steven P. Dow, Alana Glassco, Jonathan Kass, Melissa Schwarz, Daniel L. Schwartz, and Scott R. Klemmer*

Abstract Iteration can help people improve ideas. It can also give rise to fixation—continuously refining one option without considering others. Does creating and receiving feedback on multiple prototypes in parallel—as opposed to serially—affect learning, self-efficacy, and design exploration? An experiment manipulated whether independent novice designers created graphic Web advertisements in parallel or in series. Serial participants received descriptive critique directly after each prototype. Parallel participants created multiple prototypes before receiving feedback. As measured by click-through data and expert ratings, ads created in the Parallel condition significantly outperformed those from the Serial condition. Moreover, independent raters found Parallel prototypes to be more diverse. Parallel participants also reported a larger increase in task-specific self-confidence. This paper outlines a theoretical foundation for why parallel prototyping produces better design results and discusses the implications for design education.

1 Introduction

Iteration is central to learning and motivation in design (Dow et al. 2009; Hartmann et al. 2006; Schon 1995; Schrage 1999). Yet, its primary virtue—incremental, situated feedback—can also blind designers to other alternatives, steering them to local, rather than global, optima (Buxton 2007; Dow et al. 2009). To combat this, creating multiple alternatives in parallel may encourage people to more effectively discover unseen constraints and opportunities (Cross 2006), enumerate more diverse solutions (Buxton 2007), and obtain more authentic and diverse feedback

S.P. Dow • A. Glassco • J. Kass • M. Schwarz • D.L. Schwartz • S.R. Klemmer (✉)
Human-Computer Interaction Group, Stanford University, Gates Computer Science Building, 353 Serra Mall, Stanford, CA 94305, USA
e-mail: spdow@stanford.edu; srk@cs.stanford.edu

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*, Understanding Innovation, DOI 10.1007/978-3-642-21643-5_8,

from potential users (Tohidi et al. 2006). While a parallel approach has potential benefits, it can take time away from refinement.

Effectively educating a more creative workforce requires understanding how and why design practices affect results. Towards that goal, this paper investigates the relative merits of parallel and serial prototyping under time constraints. In a between-subjects experiment, 33 participants designed Web banner advertisements for a magazine. In both conditions, participants created five prototype ads and then a final ad (see Fig. 1). They received descriptive critique on each proto-type. Participants worked independently and were given equal time to create each proto-type and read each critique; the structure of the process differed across conditions. In the Serial condition, participants received feedback after creating each prototype. Participants in the Parallel condition created three prototypes, received feedback on all three, then made two more prototypes, and received feed-back again before creating a final ad design.

The study measured design performance by running a MySpace.com advertising campaign with all participants' final ad creations and measuring click-through analytics. Independent experts also rated ad quality. To measure the diversity of each participant's ad creations, independent online raters judged pair-wise similarity between each of the participants' six ad prototypes. A self-report assessment

Fig. 1 The experiment manipulates when participants receive feedback during a design process: in serial after each design (*top*) versus in parallel on three, then two (*bottom*)

measured participants' pre- and post-task view of task-specific self-efficacy (Fredrickson 2001; Hall 2008) (see Appendix B). The study concluded with an open-ended interview (see Appendix C).

Parallel participants outperformed Serial participants by all performance measures: click-through rates, time spent on the target client website, and ratings by the clients and ad professionals. Further, independent raters found that the diversity of each participant's prototypes was greater in the Parallel condition. Parallel participants reported a significant gain in self-efficacy, a measure of task-oriented confidence. Serial participants did not. In post-task interviews, nearly half of serial participants reported negative reactions to critique of their prototypes; no Parallel participants reported this. About half the participants had prior graphic or ad design experience. Participants with prior experience outperformed novices.

The study found that a parallel prototyping approach yields better results, more divergent ideas, and that parallel prototypes react more positively to critique. The results could significantly impact both how people approach creative problems and how educators teach design.

2 Theoretical Benefits of Parallel Design

Research on human problem solving traditionally examines problems with an optimal solution and a single path to reach that solution (Newell 1972). In design, problems and solutions co-evolve (Dorst and Cross 2001), constraints are often negotiable (Schon 1995), sub-problems are interconnected (Goel and Pirolli 1992), and solutions are not right or wrong, only better or worse (Rittel and Webber 1973). How and when to explore or refine solutions to open-ended problems remains an active debate in design research and education (Ball et al. 1997; Cross 2006; Nielsen and Faber 1996). Without exploration, designers may choose a design concept too early and fail to identify a valuable direction (Cross 2004). Without refinement, ideas may not reach their full potential (Ball and Ormerod 1995). Navigating a design space may come easier as designers develop intuition, however even experts can exhibit fixation (Cross 2004) and groupthink behaviors (Janis 1982). The architect Laseau posits an idealized conceptual model for exploring and refining, where designers iteratively diverge and converge on ideas, eventually narrowing to a best-fit concept (Laseau 1988). This paper investigates the hypothesis that parallel prototyping increases learning, exploration, and design task confidence. More broadly, this research seeks a richer theoretical understanding of creative work to help practitioners and students design more effectively.

2.1 Parallel Prototyping Promotes Comparison

Throughout life, people learn interactively, trying different actions and observing their effect in the world (Gopnik et al. 2001; Piaget 2001). Life provides a corpus of experiences from which to draw comparisons in new learning situations

(Kolodner 1993; Simon 1996). Examples can aid problem solving (Alexander, Ishikawa, and Silverstein 1977; Herring et al. 2009; Lee et al. 2010; Smith et al. 2008), especially when people explicitly extract principles (Gick and Holyoak 1983; Thompson et al. 2000). Comparison helps people focus on key relations (Gentner and Markman 1997), aiding the acquisition of underlying principles (Colhoun et al. 2008; Gentner et al. 2003) and sharpening categorical boundaries (Boroditsky 2007). This paper hypothesizes that parallel prototyping better enables people to compare feedback on multiple prototypes, leading to a better understanding of how key variables interrelate.

Hypothesis 1: Parallel prototyping leads to feedback comparison and produces higher quality designs.

In the ad design study, quality is measured with click-through analytics and expert ratings.

2.2 Parallel Prototyping Encourages Exploration

The open-ended nature of design problems often requires designers to imagine and try out alternative solutions (Buxton 2007; Kelley 2002). Without sufficient exploration, design teams may fixate on potential solutions (Duncker 1945; Jansson and Smith 1991), overlook key insights (Kershaw and Ohlsson 2004), make poor choices to justify prior investments in money or time (Arkes and Blumer 1985), and exhibit groupthink, a "deterioration of mental efficiency, reality testing, and moral judgment that results from in-group pressures" (Janis 1982). Numerous interventions have been proposed to help designers think divergently, laterally, or "outside the box" (de Bono 1999; Dym et al. 2005; Torrance 1974).

Osborn posited premature evaluation as a major block to organizational creativity and proposed "rules" for brainstorming: think broadly early on and save critique for later (Osborn 1963). Immediate feedback sets the focus on refinement, whereas postponing critique until after creating multiple designs encourages more divergence.

Hypothesis 2: Parallel prototyping results in more divergent concepts.

In the ad design study, independent raters judge the diversity/similarity of participants' sets of prototypes, providing a measure of design divergence.

2.3 Parallel Prototyping Fosters Design Confidence

Self-efficacy is a person's belief about their capabilities to perform towards a specific goal (Bandura 1997). High self-efficacy improves one's ability to learn (Dweck 2007), perform tasks (Bandura 1997), exert agency and persist (Mele 2005), and find enjoyment in challenges (Csikszentmihalyi 1991). People with strong self-efficacy respond less negatively to failure and focus on strengths

(Dodgson and Wood 1998). Critique, setback, and risks make creative work extremely challenging (Schrage 1999), and high self-efficacy provides an important robustness. With low self-efficacy, people are more likely to construe critique as an assessment of them, rather than as an assessment of the concept (Kosara 2007). Recognizing this, the studio model of art and design education emphasizes critiquing the work, rather than the person (Schon 1990).

Tohidi et al. revealed that potential users of interactive systems withhold critique when presented with a single prototype; the users were concerned about offending the designer (Tohidi et al. 2006). More importantly, Tohidi et al. showed that the presence of multiple alternative concepts gave users license to be more critical with their comments. This paper explores the other side of the coin: how critiquing designer-generated alternatives affects the designer's self-efficacy. This paper hypothesizes that parallel prototyping changes the investment mindset: it encourages investment in a creative process rather than in a particular idea. Serial prototyping may lead people to fixate on a single concept, causing them to construe critique as a rebuke of their only option.

Hypothesis 3: Parallel prototyping leads to a greater increase in design task-specific self-efficacy.

In the ad study, self-efficacy is measured with a multi-question self-report assessment, administered before and after the design task.

3 Method

The study described in this paper manipulates the structure of the prototyping process. Web advertising was chosen because it fulfilled the following criteria:

- Quality can be measured objectively and subjectively;
- Participants need minimal artistic or engineering ability;
- Individuals can complete tasks within a single lab session;
- Solutions demonstrate creative diversity and a range of performance quality;
- The study procedure can generate consistent and useful feedback during iteration.

3.1 Study Design

The experiment employed a between-subjects design with one independent variable: the structure of the prototyping process. The study held constant the number of prototypes created, the amount of feedback provided, and the overall time allotted. In the Parallel condition, participants created three prototypes and then got feedback, then made two more and got more feedback, then a final version. In the Serial condition, participants create five prototypes in series, receiving feedback after each prototype, then a final version. Parallel participants were instructed to start subsequent prototypes at the same intervals as serial participants.

3.2 Participants

Thirty-three participants were recruited with fliers and assigned to one of two conditions. (Of 36 recruited, 3 dropped out before the end.) Participants' average age was 22; three-fourths were students. Using a stratified randomization approach, the study balanced gender (19 females) and prior design experience across conditions. Fourteen participants reported some prior experience in ad or graphic design; none were professional designers.

3.3 Materials

3.3.1 Graphic Design Tool

Participants designed a 160 × 600 pixel banner advertisement to be hosted on the social networking site MySpace.com. Ads were created using MySpace's Flash-based AdBuilder tool (see Fig. 2). This simple graphic design tool was easy to learn, and no participants had used it before. Selecting a novel tool removes the confound

Fig. 2 The ad design study used MySpace's AdBuilder, a browser-based graphic design tool

of fluency with particular software. To insure a base level of competence, all participants had to successfully replicate a sample graphic in less than 10 min.

3.3.2 Advertising Client

Participants all created ads for the same client, Ambidextrous magazine, a student-led design publication. A design brief described the magazine's purpose and the kind of advertising desired by the client (Appendix D).

3.3.3 Prototype Critique System

Prior to the experiment, a team of three advertising and graphic design professionals developed a list of about 50 statements that could serve as critique for banner ads (see Appendix A). The list included three categories of statements—overall theme, composition and layout, and surface elements. Each category contained 12–20 statements, intended to provide high-level direction, without using explicitly positive or negative language. These statements express basic graphic design principles. During the study, the experimenter chose three statements—one from each category—to attach to each ad prototype (see Fig. 3).

The experimenter chose critiques relevant to each prototype and never repeated statements for the same participant. This process was identical for both conditions. Neither condition explicitly compared a participant's ads, such as, "The color in this ad is better than that one." In parallel, the experimenter reviewed each ad

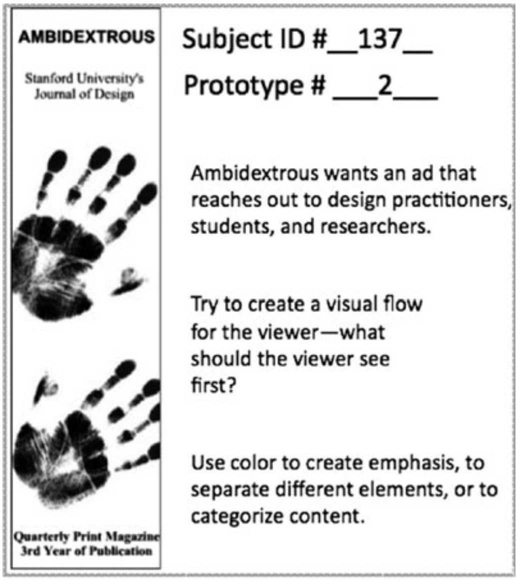


Fig. 3 Example critique

sequentially so that the process was equivalent in both conditions. The discussion section provides an in-depth treatment of the potential for bias in the study critique system.

3.4 Dependent Measures

3.4.1 Performance

After the experiment, all 33 final ad designs were uploaded to MySpace for a 15-day campaign targeted to users interested in design-related activities. This study's total advertising costs were under $200. Design performance was determined through two objective measures:

- MySpace click-through rates (CTR): daily number of clicks divided by the number of impressions (number of appearances on MySpace), and
- Google Analytics on the target client Website: number of visitors, time spent, and number of pages visited daily from each ad.

Moreover, ads were independently judged by the magazine editors and by ad professionals. Editorial staff and ad professionals represent two important—and different—stakeholder perspectives. Four magazine editors and three advertising professionals rated the participants' ad designs from 0 to 10 along five dimensions: adherence to the client's theme, creativity/originality, visual appeal, tastefulness, and adherence to graphic design principles. Raters were blind to condition and rated ads individually, with no knowledge of other raters' scores.

3.4.2 Divergence

Creating a diverse set of ideas helps people understand the space of designs and their relative merits (Buxton 2007). To obtain a measure of idea diversity, independent raters assessed pair-wise similarity of all combinations of each participant's ads (see Fig. 4). Raters were recruited from Amazon Mechanical Turk, a crowdsourcing system for paying workers for short online tasks. For each ad, raters assessed similarity on a scale from 1 to 7 (not similar to very similar). Each rater assessed a randomly ordered set of at least 50 ads. Rating a large number of ads helped raters calibrate their assessments. This measure generated 14,850 judgments (30 worker assessments on each of the 15 pair-wise comparisons for 33 participants).

3.4.3 Self-efficacy

Questions on self-efficacy assessed participants' views of their graphic design ability (adopted from self-efficacy assessments in education (Fredrickson 2001;

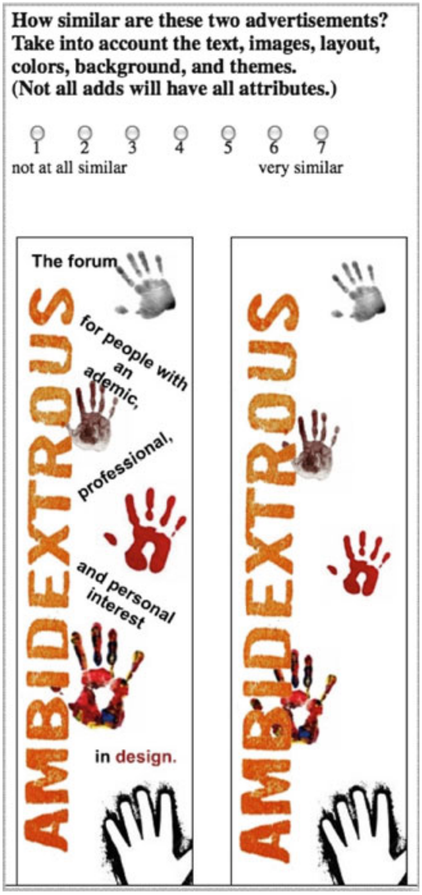


Fig. 4 Example pair-wise ad similarity rating, a measure of design divergence

Hall 2008)). The assessment asks participants to rate their ability to: create advertisements, understand design problems, detect problems in a design idea, and incorporate feedback into a design idea (see Appendix B). Each question solicited a seven-point Likert scale response. The same questions were administered before and after the design task, creating a difference measure (the time between the pre and post test was 1 h). Comparing the change in self-efficacy measures how the process manipulation (Parallel/Serial) influenced an individuals' belief in their design abilities.

3.5 Procedure

The experiment had the following steps (see Fig. 5): consent form, pre-task questions and demographics, practice task, design brief/instructions, prototyping periods (10 min per prototype), critique reports (5 min per prototype), final design

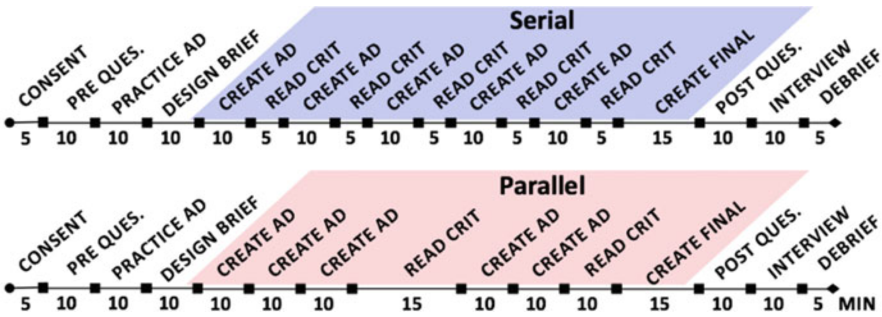


Fig. 5 Procedure for *Serial* and *Parallel* conditions, with timing

period (15 min), post-task questions, an open-ended interview, and a final debriefing to reiterate the consent details. The practice task required participants to replicate a graphic (unrelated to the main task). The design brief detailed the ad campaign's client, Ambidextrous magazine and outlined three goals: increase traffic to the Ambidextrous Web site, impress the editors, and create ads with effective graphic design.

Participants were instructed they would receive critique from an ad expert on each prototype. As experimenters prepared critique reports in a separate room, participants were allowed to navigate the client Web site, search for images, or sketch on paper. After a set amount of time (2 min per ad), participants received an envelope containing the printed ad prototype with feedback statements. As part of the final questionnaire, participants filled out the "Creativity Achievement Questionnaire" developed by Carson et al. to assess creative achievement across ten domains (visual arts, music, dance, architecture, writing, humor, inventions, scientific discovery, theater, and culinary arts) (Carson et al. 2005).

For 150 min of participation, subjects received $30 cash. Experiment proctors only entered the participant room to introduce the tool and task, to deliver feedback envelopes, and to conduct the open-ended interview.

4 Results

Participants generated a wide variety of ad concepts. The most successful ads (high click-through rates and ratings) tended to be simple, visually balanced, professional, creative, matched the theme of the magazine and contained some sort of intriguing hook, such as the face made of hands in the highest click-through performer (see Fig. 6).

The study supported all three hypotheses. Participants in the Parallel condition produced higher quality designs (better click-through rates and higher subjective ratings) and more divergent prototypes. They also reported a greater increase in

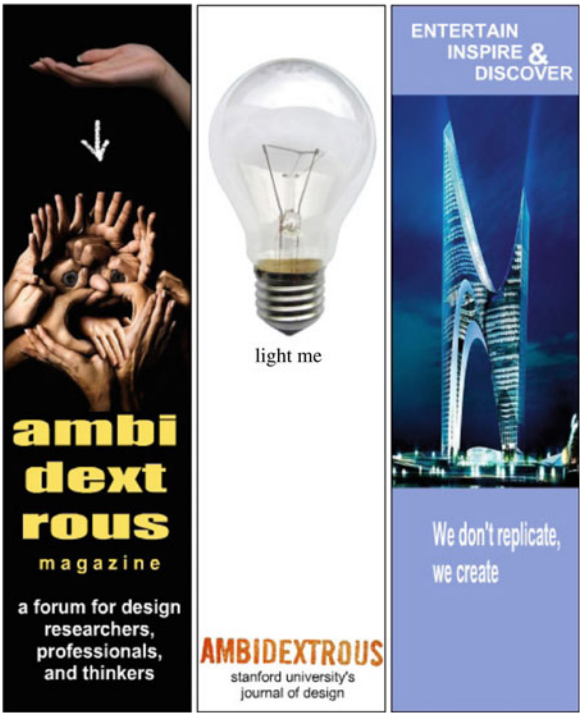


Fig. 6 Example ads: (*Left*) Parallel ad, 1st in click-through rate, 6th in expert rating; (*Middle*), Parallel ad, 9th in CTR, 1st in expert rating; (*Right*) Serial ad, 4th in CTR, 32nd in expert rating

task-specific self-efficacy. Participants with prior experience in ad or graphic design outperformed complete novices, however the prototypes created by experienced participants were less diverse than novices.

4.1 Parallel Ads Outperformed Serial Ads

4.1.1 Online Click-Through Rates

Performance data on each ad was extracted from MySpace and Google Analytics on the Ambidextrous Web site (see Table 1). MySpace reports that over the 15-day campaign, the 33 participant ads received 501 total clicks on 1,180,320 total impressions (i.e., number of ad appearances), giving an overall average click-through rate (CTR) of 0.0424% or 424 clicks per million impressions. The top two click-through rates were both Parallel ads, with 735 and 578 clicks per million impressions, respectively. The bottom two ads were both from the Serial condition; neither received any clicks.

MySpace users clicked Parallel ads more than Serial ads. Counting clicks can be misleading because some ads are shown more than others: when an ad performs

Table 1 Summary of campaign data from MySpace and Google Analytics (standard deviation in parentheses)

	Parallel	Serial
Performance data from advertising host (MySpace.com)		
Total impressions	665,133 (43,968)	515,187 (36,373)
Total clicks	296 (22.8)	205 (19.1)
Clicks per million impressions	445.0 (18.3)	397.9 (19.6)
Performance data on client site (Google Analytics reports)		
Total visitors	264 (19.9)	158 (15.3)
Average time (sec) per visitor	31.3 (143)	12.9 (79.9)
Pages visited on site	394 (31.6)	198 (21.1)
Pages visited per visitor	1.49 (0.48)	1.25 (0.41)

well, the host often shows it more.[1] There are two approaches for measuring performance comparably. The first is to measure clicks per impression. The second is to hold impressions constant and compare clicks. In this study, ads received an approximately equal number of impressions for the first 5 days. A chi-squared analysis examines performance through day 5. Parallel ads had 79,800 impressions with 44 clicks and Serial ads had 79,658 impressions with 26 clicks (see Fig. 7); at this early stage, Parallel ads had a significantly higher click-through rate ($\chi^2 = 4.60$, $p < 0.05$).

Over the entire campaign, an analysis of variances was performed with condition (Serial/Parallel) and Creativity test scores (high/low) as factors and final click-through rates for each ad as dependent variable. Parallel outperformed Serial, 445.0 and 397.9 clicks per million impressions respectively ($F(1,30) = 4.227$, $p < 0.05$) (see Table 1).[2] Also, high Creativity scorers had a higher average click-through rate (352 clicks per million) than low scorers (305); this difference is not significant ($F(1,30) = 3.812$, $p = 0.06$).

[1]Like many advertising hosts, MySpace varies the number of impressions based on prior performance of the ad. MySpace does not publish their algorithm for determining the frequency of impressions, but a repeated measures general linear model with the Day 5 CTR as a factor and impressions on each subsequent day as dependent measure shows the CTR for days 1–5 to be a significant predictor of the number of impressions for the final 10 days of the campaign ($F(1,29) = 23.2$ and $p < 0.01$). MySpace receives payment on each click; intuitively, it is in their interest to show high-CTR ads more often.

[2]According to Google Analytics, the Ambidextrous Web site received 422 total visitors during the 15-day campaign, 79 less than the number of clicks reported by MySpace. One explanation for the disparity could be that users clicked the ad and then hit "back" before the browser loaded the client site. The 264 visitors from Parallel ads are significantly more than the 158 visitors from Serial when compared against impressions ($\chi^2 = 6.61$, $p < 0.05$).

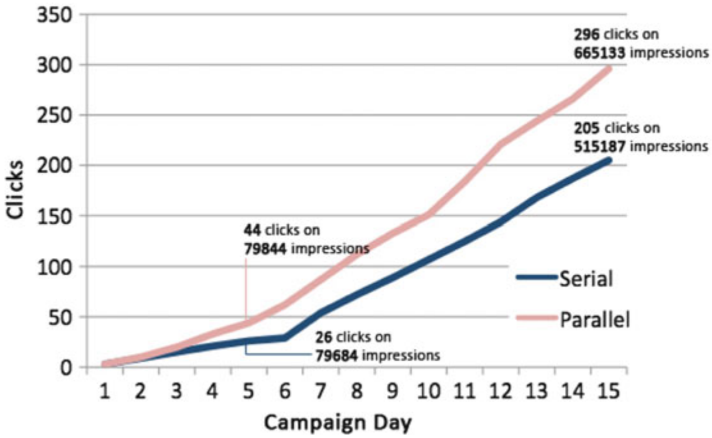


Fig. 7 Parallel ads received more clicks—and more clicks per impression—than serial ads during a 15-day campaign

4.1.2 Visitor Behavior on Client Site

One common measure of ad effectiveness is time on site (Sterne 2002). The average time on site for Parallel ads (31.3 s) was greater than Serial ads (12.9 s) ($t(493) = 1.781$, $p < 0.05$). The result suggests that Parallel ads were more likely to reach people genuinely interested in the product offered by the clients. The number of pages visited per visitor was about the same: 1.49 for Parallel and 1.25 for Serial. Visitor's navigation behavior did not show a statistical difference: 71 of 264 visitors from Parallel ads and 35 of 158 visitors from Serial ads visited pages beyond the front page of Ambidextrous' website ($\chi^2 = 1.18$, $p > 0.05$).

4.1.3 Independent Expert Ratings

Overall rating contained five 10-point rating scales: adherence to the client's theme, creativity/originality, visual appeal, tastefulness, and adherence to graphic design principles. The average expert rating across all ads was 23.0 out of 50 (35.6 high and 15.0 low). The three top-rated ads were all from the Parallel condition. An analysis of variances was performed with condition (Parallel/Serial), prior design experience (some/none), rater (seven independent raters), and rater type (client or professional) as factors and overall rating as the dependent variable. Parallel ads were rated higher ($\mu = 24.4$, $SD = 9.7$) than Serial ads ($\mu = 21.7$, $SD = 8.8$) ($F(1,203) = 3.871$, $p < 0.05$) (see Fig. 8). Experienced participants created higher-rated ads ($\mu = 25.7$, $SD = 9.6$) than novices ($\mu = 21.0$, $SD = 8.6$) ($F(1,203) = 20.98$, $p < 0.05$) (see Fig. 8). There was no interaction effect between condition and prior experience.

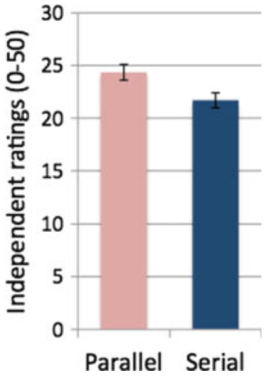


Fig. 8 Parallel ads received higher average independent ratings (0–50 scale) than serial ads

Some raters had higher average ratings than others ($F(5,203) = 18.88$, $p < 0.05$). There was no interaction between rater and condition; raters generally agreed that parallel ads outperformed serial ads. Analyses of variances were conducted separately for all five dimensions with condition and experience as factors. All dimensions skewed towards Parallel ads, but only two—tastefulness ($F(1,227) = 7.527$, $p < 0.05$) and adherence to graphic design principles ($F(1,227) = 4.188$, $p < 0.05$)—were independently significant in favor of Parallel ads. The ratings provided by the clients were higher on average ($\mu = 24.3$, $SD = 9.5$) than those provided by external ad professionals ($\mu = 22.0$, $SD = 9.1$) ($F(1,203) = 4.376$, $p < 0.05$). There was no interaction effect between rater type and condition.

Ads that performed well online generally also received high ratings by the clients and ad professionals. The ad with the best overall click-through rate received the 6th highest rating by the clients and ad professionals (see Fig. 6, *left*). Likewise, the highest rated ad achieved the 4th highest click-through performance (see Fig. 6, *middle*). There were anomalies, such as the top two ads in the Serial condition. These two ads were ranked 25th and 32nd (out of 33) by the expert raters, but received the 3rd and 4th best overall click-through rates. The latter of those designs does not even mention the client (see Fig. 6, *right*). Statistically speaking, online click performance was not a predictor of overall expert rating ($R^2 = 0.057$, $F(1,31) = 1.858$, $p > 0.05$, $b = 0.192$).

4.2 Parallel Ads Were Rated More Diverse Than Serial Ads

Workers on Amazon Mechanical Turk rated Parallel ads as more divergent than Serial ads. Raters performed pair-wise similarity comparisons on a scale of 0–7 within each participant's set of six prototype ads. An analysis of variances was performed with condition (Serial/Parallel) and prior design experience as factors and pair-wise similarity rating as the dependent variable. Serial ads were deemed significantly more similar ($\mu = 3.25$, $SD = 1.96$) than Parallel ads ($\mu = 2.78$, SD = 1.66) ($F(114,816) = 239.3$, $p < 0.05$). Parallel ads were rated more divergent.[3]

Similarity ratings were not predictive (or inversely predictive) of online click performance ($R^2 = 0.032$, $F(1,31) = 0.030$, $p > 0.05$, $b = 0.009$) or overall independent ratings ($R^2 = 0.030$, $F(1,31) = 1.999$, $p > 0.05$, $b = 0.246$).

4.3 Parallel Participants' Ad Design Self-efficacy Increased

A self-efficacy assessment measured participants' belief in their ability to perform the design task. The difference between the pre- and post-task scores provides an indication of how participants' beliefs change. Across all participants, self-efficacy rose from 10.85 to 12.12 (out of 20); a paired-samples *T*-test shows a significant difference ($t(32) = 2.355$, $p < 0.05$). Examining inter-question effects, each question independently resulted in a significant rise from pre to post-task, except for question four ("rate your ability to incorporate feedback into a design idea") ($t(32) = 0.154$, $p > 0.05$). This rise is consistent with prior findings that show individual self-efficacy beliefs increase with practice (Bandura 1997; Hall 2008).

An analysis of variances was performed with condition (Serial/Parallel) and prior task experience (Experienced/Novice). Participants in the Parallel condition reported a significant increase in self-efficacy scores (see Fig. 9), a net gain of

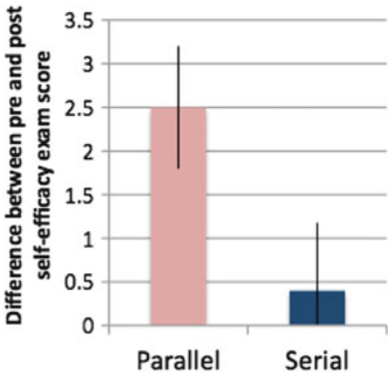


Fig. 9 Participants in the Parallel condition reported a greater increase in self-efficacy from pre to post design task

[3]Similarity ratings changed depending on whether they were generated early or late in the process. Pair-wise comparison of pairs 1-2, 2-3, and 1-3 were labeled "Early" designs; pairs 4-5, 5-6, and 4-6 were labeled "Late" designs. An analysis of variances was performed with condition (Serial/Parallel) and design-stage pairs (Early/Late/Other) as factors and similarity rating as the dependent variable. Across conditions, ads created *later* were deemed more similar ($\mu = 3.41$, SD = 2.03) than *early* ads ($\mu = 2.97$, SD = 1.77) ($F(1,14,814) = 107.835$, $p < 0.05$). The interaction between condition and design stage was marginally significant ($F(114,814) = 2.460$, $p = 0.085$). Serial ads were rated more similar than Parallel ads, both for early and late pairs, but the similarity is greater for later ads.

2.5 points ($F(1,29) = 4.210$, $p < 0.05$), while the Serial condition essentially remained even (net gain $\mu = 0.4$).

4.4 Experienced Participants Outperformed Novices

4.4.1 Online Click-Through Rates

The fourteen participants with prior experience in ad or graphic design significantly outperformed novices. Ads by participants with prior experience received 350 clicks on 752,424 impressions, compared to 151 clicks on 427,896 impressions by novices ($\chi^2 = 8.10$, $p < 0.05$). There was no interaction effect between condition and prior participant experience.

4.4.2 Visitor Behavior on Client Site

Visitors spent more time on the client's site after clicking ads created by experienced participants (38.0 s/visitor) compared to those created by novices (7.6 s/visitor) ($F(1,491) = 8.456$, $p < 0.05$). An interaction between condition and prior experience showed that having prior experience in the Parallel condition led to more time on site than prior experience in the Serial condition, 57.0–18.9 s/visitor ($F(1,491) = 4.045$, $p < 0.05$). Visitors from experienced ads were also more active navigators; 88 of 296 visitors from experienced ads and 12 of 126 visitors from novice ads visited pages beyond the front page of Ambidextrous' website ($\chi^2 = 19.96$, $p < 0.05$).

4.4.3 Divergence Ratings

Participants with prior experience created significantly more similar ads ($\mu = 3.15$, $SD = 1.86$) than novices ($\mu = 2.88$, $SD = 1.80$) ($F(1,14,816) = 76.70$, $p < 0.05$). Ads created by novices were rated more divergent. There was also an interaction effect indicating that experienced participants in the serial condition created the most similar ads ($F(1,14,816) = 36.45$, $p < 0.05$).

4.4.4 Self-efficacy Assessment

Participants with prior ad design experience reported a similar gain in self-efficacy ($\mu = 1.93$) as novices ($\mu = 0.79$) ($F(1,29) = 1.088$, $p > 0.05$). There was an interaction effect between condition and prior experience: novices reported a 2.9 increase in self-efficacy in Parallel, but a 0.73 decrease in Serial

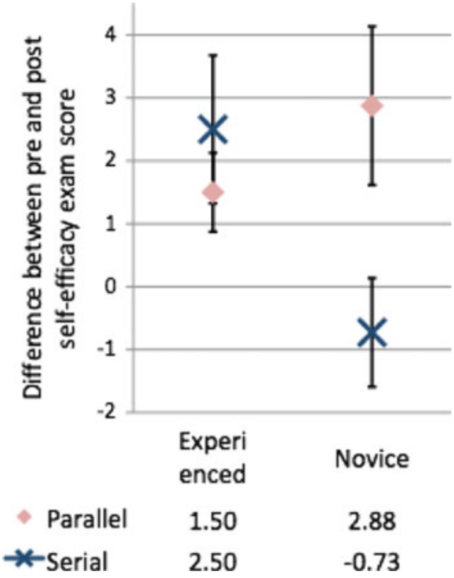


Fig. 10 Novice participants in the Parallel condition reported an increase in self-efficacy from pre to post design task; self-efficacy for novices in serial decreased

($F(1,29) = 6.331$, $p < 0.05$) (see Fig. 10). In short, parallel prototyping positive affected an individual's belief in their ad design ability, especially for novices.

5 Analysis

A parallel approach led to ad creations with better performance by every measure: higher independent ratings, more impressions served up by MySpace, better click-through rates, more visitors to the client Web site, and more site interaction per visitor. Participants created the same number of prototypes and received equivalent feedback in the same time period. The only difference between conditions was a matter of when participants received critique on their ideas—after each concept or after multiple creations.

Why did the process manipulation affect performance? This section offers three explanations for the differential results: comparison helped the parallel participants learn ad design principles, parallel participants better explored the design space, and serial participants perceived the critique as negative and thus gained no confidence at ad design.

Did parallel feedback impact how participants learned? Comparison processes can facilitate inductive reasoning on rival observations (Colhoun et al. 2008; Thompson et al. 2000). Since Parallel participants received feedback on multiple ideas simultaneously, they were more likely to read and analyze critique statements side-by-side. Direct comparison perhaps helped Parallel participants better understand key design principles and lead to more principled choices for subsequent prototypes. In Serial prototyping, participants' ideas tended to follow directly from the feedback. This serial approach may implicitly encourage refinement at the expense of exploration. Performance likely improves in Parallel because people exercise their comparative abilities to learn contextual constraints, malleable variables, and their interrelations.

Learning a Parallel approach may change future behavior. When asked to describe their process for future design projects, 11 of 16 Parallel participants said they would create more than one prototype and obtain copious feedback; only 5 of 17 Serial participants made similar claims ($\chi^2 = 2.63$, $p > 0.05$). As one Parallel participant said, "not spending too much time on any single prototype is useful because then you don't go into details too much."

Did a parallel process impact how participants explored concepts? The study showed Parallel participants created significantly more divergent prototypes; Serial participants tended to create more similar designs. The interviews revealed the role of critique, as one Serial participant explained, "I think the feedback helped. I kept repeating the same mistakes, but maybe less and less each time… the feedback reiterated that." Another Serial participant said:

> I would try to find a good idea, and then use that idea and keep improving it and getting feedback. So I pretty much stuck with the same idea.

This notion of "sticking" with an idea or using the feedback to decide where to go next did not surface in the Parallel condition. As one Parallel participants reported: "I didn't really try to copy off of the ads that I did before…I just made new ideas." Both the divergence measure and the qualitative interviews suggest the parallel structure supports more generative thinking and reduces fixation.

Parallel prototyping may encourage both a broad enumeration stage and a subsequent reflection stage. By contrast, Serial's immediate feedback implicitly encourages refinement. On this view, the fact that Parallel delays feedback is actually an advantage. From a behaviorist perspective, this can seem counterintuitive because immediate feedback highlights the connection between cause and effect. However, delay helps learners reflect: readily available, immediate feedback can be a crutch that discourages building a deep understanding (Anderson and Schooler 1990; Schmidt et al. 1989).

There are countless ways to combine text, images, and backgrounds in a 160×600 pixel ad design; some combinations perform better than others. To use an analogy, exploring design possibilities is like simulated annealing (Granville 1994). Creative work often benefits from broadly exploring a design space with high entropy before optimizing in one direction. Perhaps serial participants hill-climbed to local, rather than global optima.

Experienced participants created a less diverse set of designs than novices; they also outperformed novices. In general, experts may know a priori which areas are promising and which to avoid. By contrast, novices have to learn what is effective through trial and error.

Did Parallel participants gain more confidence in their ad-design ability? Parallel participants reported self-efficacy gains, while the Serial participants reported no change. This effect was more pronounced for novices. Serial participants also perceived the expert feedback more negatively. In open-ended interviews, 13 of 16 Parallel participants said the feedback was helpful or intuitive compared to 6 of 17 in Serial ($\chi^2 = 7.13$, $p < 0.05$). More notably, 8 of 17 of the Serial participants reported the feedback as negative, compared to no such reports in the Parallel condition ($\chi^2 = 9.94$, $p < 0.05$). One participant in the Serial condition said:

> I received really negative comments saying [the clients] are looking for a creative and clever ad, which in other words is saying that this is stupid or ridiculous.

Moreover, participants were asked to leave their email if they wanted to later volunteer for Ambidextrous magazine. Twelve out of 16 Parallel participants provided their email, while only 5 of 17 did the same in Serial ($\chi^2 = 6.86$, $p < 0.05$), which suggests the Parallel process may have helped motivate future action.

Perhaps having multiple alternative designs encourages investment in a creative process rather than a particular idea. Consequently, the parallel process encourages viewing critique as an opportunity for improvement. In contrast, the fixation engendered by serial prototyping may cause people to take critique as a catastrophic rebuke of their only option. With only one option there is no separation between designer and design. Parallel offers people distance between ego and object; Serial conflates them.

6 Follow-Up Studies

Two short follow-up experiments examined questions raised by the main study.

Did the experimenters (possibly subconsciously) provide better critique to the parallel participants? To assess bias, two ad professionals unfamiliar with the experimental manipulation provided blind-to-condition independent assessments of the critique statements. The expert judges performed a selection task resembling the task performed by the experimenter. After reading about the client's advertising needs, the judge viewed an ad prototype and two triads of critique statements; one triad contained the three statements chosen during the actual experiment and the other triad was a random selection from the critique statement corpus. Judges were instructed to select a triad that "provides the most appropriate critique for the advertisement."

An intra-class correlation (ICC) with a two-way mixed model (Shrout and Fleiss 1979) calculated the reliability between the experimenter's choice of statements and each expert judge's choice. The ICC(3,1) single measure correlation among raters on Parallel ads is 0.434, and 0.496 for Serial ads. There is no significant difference between these numbers and both represent a moderate level of inter-rater reliability (Landis and Koch 1977). In short, experts generally agreed with the feedback provided, and the level of agreement was comparable across conditions.

Did the critique statements help participants produce better ads? A follow-up study examined the value of the scripted ad critique statements in Appendix A. Thirty participants followed a serial process to create three prototypes and one final advertisement. The final ads were launched in an online ad campaign and rated by experts. Participants were randomly assigned to one of three conditions: targeted, random, and none. In the targeted condition, an experimenter selected three critique statements intended to help the participant improve their design. In the random condition, a random algorithm selected three critique statements. Participants in the none condition received no critique; rather, they viewed the client Web site during an equivalent critique interval.

In a 3-week campaign, ads that received targeted critique had 49,414 impressions with 203 clicks, ads with no feedback had 49,663 impressions with 179 clicks, and ads that received randomly selected critique statements received 49,543 impressions with 157 clicks ($\chi^2 = 6.01$, $p < 0.05$). Moreover, twenty independent experts rated ads (on a 0–30 scale) with targeted critique higher $\mu = 15.9$ (SD = 5.4) than ads with random critique $\mu = 15.1$ (SD = 5.2) and ads with no critique $\mu = 14.4$ (SD = 6.2) ($F(2,597) = 3.287$, $p < 0.05$). The study found that targeted critique helped participants learn basic principles of graphics design and produce better ads.

7 Discussion About Experimental Approach

The experimental paradigm introduced in this paper provides several important benefits for studying design. First, design results can be objectively measured through real-world analytics, and subjectively assessed through crowdsourced and stakeholder ratings. Second, solutions demonstrate creative diversity and exhibit a broad range of performance. Third, it offers a mechanism for presenting feedback interactively and studying its effects. The advertising domain achieves theses goals particularly well. Hopefully this paradigm will prove useful in additional domains.

Web analytics can be tremendously valuable for experimental work; it also presents several challenges. Web hosts often show ads differentially based on performance. Poor performing ads must have a large enough number of impressions to yield a robust measure of click-through rate. Additionally, click-through rate can vary over time. Fair comparison requires holding the number of impression constant, analyzing data from a time interval with roughly balanced impression rate, or using more sophisticated statistical analysis to factor out time effects.

8 Conclusions and Future Work

This paper found that when people create multiple alternatives in parallel they produce higher-quality, more-diverse work, and experience a greater increase in self-efficacy. Many excellent designers practice this approach already; their successes inspired this research. Hopefully, these results will encourage many more practitioners and teachers to adopt a parallel approach. Integrating the parallel approach into design practicum can inculcate healthy prototyping habits and help foster a positive outlook toward critique. In the future, software tools and infrastructure providers could provide a powerful benefit by enabling people to rapidly design alternatives and experimentally compare them. More broadly, this research seeks to develop a theoretical understanding of creative work to help practitioners and students solve design problems more effectively. An important direction for future work is to study the impact of parallel design in other contexts, especially at longer time scales and for design teams.

Acknowledgements We thank Ugochi Acholonu, Lera Boroditsky, Jodi Forlizzi, Jeffrey Heer, Erik Stolterman, Ewart Thomas, and Terry Winograd for valuable discussions about this research. We thank Rachel Lopatin for analyzing the targeted critique data. The Hasso Plattner Design Thinking Research Program financially supported this work; Intel Corporation donated computers.

Appendix A: Expert Critique Statements

Overall/Thematic

- Ambidextrous seeks an ad with a single clear message that matches the theme of their journal.
- Ambidextrous wants an ad that clarifies the product: a journal about design and design process.
- Ambidextrous desires an ad that is simple, readable, consistent, and deliberate.
- Ambidextrous does not want the ad to sound exclusive; they are open to anyone with interest.
- Ambidextrous is looking for a creative and clever ad.
- Ambidextrous is looking for a professional and tasteful ad.
- Ambidextrous wants an exciting and visually appealing ad.
- Ambidextrous wants an ad that matches the journal's style.
- Ambidextrous wants an ad that reaches out to design practitioners, students, and researchers.
- Use graphics/images that support the overall message. What message are you trying to convey?
- Use colors/fonts that support the overall message. What message are you trying to convey?
- Remember that the ad is a link; the URL does not necessarily have to be on the ad design.

Composition and Layout

Visual Flow and Balance

- Try to create a balanced layout where the graphics don't tilt to one side or the other.
- Try to create a visual flow for the viewer—what should the viewer see first?
- Think about the proximity of different elements. How close together or far apart elements are placed suggests a relationship (or lack thereof) between otherwise disparate parts.
- To help balance the ad, leave slightly more space at the bottom relative to the top of the ad.
- Contrast the position of elements to draw the viewer's attention to the most important parts.
- To create consistency for the viewer, create a consistent and balanced look using repetition.

Spacing and Alignment

- Align text and graphics to create more interesting, dynamic, and appropriate layouts.
- Use alignment to create a clean and organized look.
- It's ok to break alignment only to draw the viewer's attention to important elements in the ad.
- Use white around text and images to help frame the content.
- Use space—the absence of text and graphics—to provide visual breathing room for the eye.
- Try to balance the spacing around the border of the ad design.
- These visual elements in the ad don't line up.
- Consider playing around with different ways to justify the text (e.g., center, left, or right-justified).

Emphasis and Hierarchy

- Be conscious of competing elements in the ad. Think about what should have emphasis.
- Draw the viewer's attention to elements by contrasting size (scale).
- Think about the visual hierarchy of the different elements (texts, images, colors, etc.) of the ad. What is the most important?
- Help the viewer recognize, identify and comprehend the most important information in the ad.
- Use elements with visual intensity or color for emphasis.

Fonts, Colors, Images

Font Type

- Try not to distort the font so that it becomes hard to read.
- Use large, bold font/graphics to create focus or emphasis on the ad design.
- If using text over an image, make the text bigger and darker than normal; make sure it is readable.
- For text to stand out it has to be substantially different than other text.
- Try not to mix serif and sans serif fonts.
- Avoid using two different fonts that are too similar.
- Try not to over emphasize text elements. (ex. a font does not need to be large, bold, and italic).

Images

- Use large, bold graphics to create the focus of the ad design.
- Consider using images for more visual impact.
- Consider using fewer images.
- Try not to over-rotate images, as it often distorts the content.

Color

- Use color to create emphasis, to separate different elements, or to categorize content.
- Avoid really light, bright colors.
- Avoid colors together that look too similar (ex. brown and grey).
- Try to use different colors that go well together.
- Avoid complicated backgrounds.
- Try to create a good visual separation between the text and the background

Appendix B: Self-efficacy Questions (Pre and Post Task)

One a scale from "Not confident at all" (1) to "Very confident" (7), how confident are you:

1. With your ability to design advertisements?
2. At understanding design problems?
3. With detecting problems in your design?
4. With incorporating expert feedback in your design?

Appendix C: Post Interview Guide

These questions provided guidance for the final interview; the exact order and phrasing varied.

- Please describe the final design you came up with.
- What do you think are the strengths and weaknesses of your ad design?
- Describe your design process and why you created each design.
- How did the feedback affect you? Was it helpful? What did you learn about graphic ad design?
- If you created another ad, how would you approach it? Describe your design process. Would you seek feedback? How many prototypes would you create?

Appendix D: Advertising Design Brief

Assignment

You have been hired to design a graphic advertisement for Ambidextrous, Stanford University's Journal of Design. You will learn a new graphic design tool, prototype a number of example ads, receive feedback from an ad design expert, and then create a final ad for MySpace.com.

Goals

Keep in mind the following goals as you create your ads:

- Increase traffic to the Ambidextrous website: http://ambidextrousmag.org/
- Reach out to the target audience: designers, researchers, practitioners, and students who are interested in stories about the process of design.
- Impress the editors of Ambidextrous. The client wants an ad that fits their overall aesthetic and theme (see below).
- Create ads with effective graphic design.

What Is Ambidextrous?

Ambidextrous is Stanford University's Journal of Design. In its 3rd year of publication, Ambidextrous is a quarterly subscription-based print magazine that features people and processes involved in design and design thinking. Ambidextrous is a forum for the cross-disciplinary, cross-market community of people with an academic, professional and personal interest in design. Each issue focuses on one general topic. Previous topics have included Space, Secrecy, Food, The Future, Danger, Developing, Borders and Interfaces, etc. Articles are written by the community at large, and edited, illustrated, and photographed entirely by volunteers.

Theme and Aesthetic for the Ambidextrous Ad

The Ambidextrous editors would like an ad that embodies the theme and general aesthetic of the journal. The journal tells stories about people who do design and the process of getting there, not just final products. Readers of the journal are not an exclusive club—it's intended to be accessible to folks without formal design training. In general they are looking for an ad that is tasteful, creative, professional, visually appealing, and conveys a clear message about the product.

Rules/Requirements

- You may download and use graphics, images, text etc. as you see fit.
- You may not use another company's logo, copyrighted images, profanity, obscenity or nudity. Unacceptable ads will be rejected by the research team.
- Do not include the magazine's URL on the ad. Clicking the ad will direct the user to the site.

References

"Amazon Mechanical Turk." https://www.mturk.com/mturk
"Ambidextrous Magazine." http://ambidextrousmag.org/
"Google Analytics." http://www.google.com/analytics
Alexander, Christopher, Sara Ishikawa, and Murray Silverstein. 1977. A Pattern Language: Towns, Buildings, Construction. Oxford University Press.
Anderson, John, and Lael Schooler. 1990. "The Disruptive Potential of Immediate Feedback." Pp. 702–708 in CMU Department of Psychology. Cambridge, MA.
Arkes, Hal R., and Catherine Blumer. 1985. "The psychology of sunk cost." Organizational Behavior and Human Decision Processes 35:124–140.
Ball, Linden J, Jonathan St.B.T., Evans Ian Dennis, and C. Ormerod. 1997. "Problem-solving Strategies and Expertise in Engineering Design." Thinking & Reasoning 3:247.
Ball, Linden J., and Thomas C. Ormerod. 1995. "Structured and opportunistic processing in design: a critical discussion." International Journal Human-Computer Studies 43:131–151.
Bandura, Albert. 1997. Self-Efficacy: The Exercise of Control. 1st ed. Worth Publishers.
de Bono, Edward. 1999. Six Thinking Hats. 2nd ed. Back Bay Books.
Boroditsky, Lera. 2007. "Comparison and the development of knowledge." Cognition 102:118–128.

Buxton, Bill. 2007. Sketching User Experiences: Getting the Design Right and the Right Design. Morgan Kaufmann.
Carson, Shelly H., Jordan B. Peterson, and Daniel M. Higgens. 2005. "Reliability, Validity, and Factor Structure of the Creative Achievement Questionnaire." Creativity Research Journal 17:37–50.
Colhoun, J., Dedre Gentner, and J. Loewenstein. 2008. "Learning abstract principles through principle-case comparison." Pp. 1659–1664 in Proceedings of Cognitive Science Society.
Cross, Nigel. 2006. Designerly Ways of Knowing. 1st ed. Springer.
Cross, Nigel. 2004. "Expertise in design: an overview." Design Studies 25:427–441.
Csikszentmihalyi, Mihaly. 1991. Flow: The Psychology of Optimal Experience. Harper Perennial.
Dodgson, P.G., and J.V. Wood. 1998. "Self-esteem and the cognitive accessibility of strengths and weaknesses after failure." Journal of Personality and Social Psychology 75:178–197.
Dorst, Kees, and Nigel Cross. 2001. "Creativity in the design process: co-evolution of problem-solution." Design Studies 22:425–437.
Dow, Steven P., Kate Heddleston, and Scott R. Klemmer. 2009. "The Efficacy of Prototyping Under Time Constraints." Pp. 165–174 in Proceeding of the seventh ACM conference on Creativity and cognition.
Duncker, Karl. 1945. On Problem-solving. Greenwood Press Reprint.
Dweck, Carol. 2007. Mindset: The New Psychology of Success. Ballantine Books.
Dym, Clive L, Alice M Agogino, Daniel D Frey, and Larry J Leifer. 2005. "Engineering Design Thinking, Teaching, and Learning." Journal of Engineering Education 94:103–120.
Fredrickson, Barbara L. 2001. "The role of positive emotions in positive psychology: The broaden-and-build theory of positive emotions." American Psychologist 56:218–226.
Gentner, Dedre, Jeffrey Loewenstein, and Leigh Thompson. 2003. "Learning and transfer: A general role for analogical encoding." Journal of Educational Psychology 95:408, 393.
Gentner, Dedre, and Arthur B Markman. 1997. "Structure mapping in analogy and similarity." American Psychologist 52:45–56.
Gick, Mary L., and Keith J. Holyoak. 1983. "Schema induction and analogical transfer." Cognitive Psychology 15:1–38.
Goel, Vinod, and Peter Pirolli. 1992. "The structure of design problem spaces." Cognitive Science 16:395–429.
Gopnik, Alison, Andrew N. Meltzoff, and Patricia K. Kuhl. 2001. The Scientist in the Crib: What Early Learning Tells Us About the Mind. Harper Paperbacks.
Granville, V. 1994. "Simulated annealing: A proof of convergence." IEEE Transactions on Pattern Analysis and Machine Intelligence 16:652–656.
Hall, T. Simin. 2008. "Improving Self-Efficacy in Problem Solving: Learning from Errors and Feedback." Text, University of North Carolina at Greensboro.
Hartmann, Björn et al. 2006. "Reflective physical prototyping through integrated design, test, and analysis." Pp. 299–308 in Proceedings of the 19th annual ACM symposium on User interface software and technology.
Herring, Scarlett R., Chia-Chen Chang, Jesse Krantzler, and Brian P. Bailey. 2009. "Getting inspired!: understanding how and why examples are used in creative design practice." Pp. 87–96 in Proceedings of the 27th international conference on Human factors in computing systems.
Janis, Irving L. 1982. Groupthink: Psychological Studies of Policy Decisions and Fiascoes. 2nd ed. Wadsworth Publishing.
Jansson, David, and Steven Smith. 1991. "Design Fixation." Design Studies 12:3–11.
Kelley, Thomas. 2002. The Art of Innovation. Broadway Business.
Kershaw, Trina C, and Stellan Ohlsson. 2004. "Multiple causes of difficulty in insight: the case of the nine-dot problem." Journal of Experimental. Psychology Learning, Memory, and Cognition 30:3–13.
Kolodner, Janet. 1993. Case-Based Reasoning. Morgan Kaufmann.
Kosara, Robert. 2007. "Visualization Criticism - The Missing Link Between Information Visualization and Art." Pp. 631–636 in Proc of the Conf on Information Visualization. IEEE Computer Society.

Landis, J R, and G G Koch. 1977. “The measurement of observer agreement for categorical data.” Biometrics 33:159–174.

Laseau, Paul. 1988. Graphic Thinking for Architects and Designers, 2nd Edition. 2nd ed. John Wiley & Sons Inc.

Lee, Brian, Savil Srivastava, Ranjitha Kumar, Ronen Brafman, and Scott R Klemmer, 2010. “Designing with Interactive Example Galleries.” In Proceedings of the 28th international conference on Human factors in computing systems.

Mele, Alfred R. 2005. Motivation and Agency. Oxford University Press US.

Newell, Allen. 1972. Human Problem Solving. Sixth Printing. Prentice Hall.

Nielsen, Jakob, and Jan Maurits Faber. 1996. “Improving System Usability Through Parallel Design.” Computer 29:29–35.

Osborn, Alex F. 1963. Applied Imagination: Principles and Procedures of Creative Problem Solving. Charles Scribner’s Sons.

Piaget, Jean. 2001. The Psychology of Intelligence. Routledge.

Rittel, Horst, and Melvin Webber. 1973. “Dilemmas in a general theory of planning.” Policy Sciences 4:169, 155.

Schmidt, R.A., D.E. Young, S. Swinnen, and D.C. Shapiro. 1989. “Summary Knowledge of Results for Skill Acquisition: Support for the Guidance Hypothesis.” Journal of Experimental Psychology: Learning, Memory, and Cognition 15:352–359.

Schon, Donald A. 1990. Educating the Reflective Practitioner: Toward a New Design for Teaching and Learning in the Professions. 1st ed. Jossey-Bass.

Schon, Donald A. 1995. The Reflective Practitioner: How Professionals Think in Action. Ashgate Publishing.

Schrage, Michael. 1999. Serious Play: How the World’s Best Companies Simulate to Innovate. Harvard Business School Press.

Shrout, P E, and J L Fleiss. 1979. “Intraclass correlations: uses in assessing rater reliability.” Psychological Bulletin 86:420–428.

Simon, Herbert A. 1996. The Sciences of the Artificial - 3 rd Edition. 3 rd ed. The MIT Press.

Smith, Steve, Nicholas Kohn, and Jami Shah. 2008. “What You See Is What You Get: Effects of Provocative Stimuli on Creative Invention.” in NSF International Workshop on Studying Design Creativity.

Sterne, Jim. 2002. Web Metrics: Proven Methods for Measuring Web Site Success. 1st ed. Wiley.

Thompson, Leigh, Dedre Gentner, and Jeffrey Loewenstein. 2000. “Avoiding Missed Opportunities in Managerial Life: Analogical Training More Powerful Than Individual Case Training.” Organizational Behavior and Human Decision Processes 82:60–75.

Tohidi, Maryam, William Buxton, Ronald Baecker, and Abigail Sellen. 2006. “Getting the right design and the design right.” Pp. 1243–1252 in Proceedings of the SIGCHI conference on Human Factors in computing systems.

Torrance, E. Paul. 1974. Torrance tests of creative thinking. Personnel Press, Ginn and Co., Xerox Education Co.

Part III
Distributed Design Collaboration and Teamwork in Design Thinking

Towards a Shared Platform for Virtual Collaboration Monitoring in Design Research

Thomas Kowark, Matthias Uflacker, and Alexander Zeier*

Abstract Prior applications of a system to monitor IT-mediated communication activities of design teams provided new insights into the collaboration behavior during the early phases of concept creation and prototyping. We now take our approach to the next level by sketching an architecture for a platform that aims to establish 'out- of-the-box' monitoring capabilities for virtual team environments and to facilitate the sharing and evaluation of recorded activities within a larger research community. To further demonstrate the flexibility and applicability of our instrument, we present results and experiences gained from a recently conducted observation of software engineering teams. Our vision is a common service for capturing and analyzing virtual collaboration activities that promotes comparative research and team diagnostics in engineering design.

1 Introduction

Virtual collaboration has become an integral part of the daily work of engineers. Information sharing, communication, and coordination activities carried out through email, groupware, and online services increasingly determine the way in which engineering teams design and prototype new products, software, or services. With the critical importance of effective and efficient team communication being generally acknowledged, the question of how virtual collaboration behavior affects the quality of engineering processes still remains largely unanswered. One reason for this lack of understanding is the difficulty to systematically observe, analyze, and compare such processes. While recent research has begun to examine virtual

T. Kowark • M. Uflacker • A. Zeier (✉)
Hasso Plattner Institute, University of Potsdam, August-Bebel-Str. 88, 14482 Potsdam, Germany
e-mail: thomas.kowark@hpi.uni-potsdam.de; matthias.uflacker@hpi.uni-potsdam.de; alexander.zeier@hpi.uni-potsdam.de

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*, Understanding Innovation, DOI 10.1007/978-3-642-21643-5_9,

collaboration in co-located and distributed design teams more closely, the lack of generally applicable instruments for monitoring the broad range of online team activities hinders in-depth investigations. Existing approaches commonly rely on tailor-made tools that work well in a specific work scenario or with certain collaboration tools, but are not applicable in different observation contexts. Transferring those instruments to other collaboration scenarios is often impossible or involves the rewriting of software or interfering with the process under study. Repetitive efforts and high costs of implementing customized solutions to observe virtual collaboration processes are the result. Furthermore, the use of isolated instruments and data formats prevents other researchers from replicating or verifying previous findings, an important criterion for relevance and rigor in empirical design research (Dixon 1987). A broadly applicable technological foundation for monitoring and studying virtual collaboration in the field is needed. It has to minimize the efforts for data collection and analysis and facilitate comparative research in design.

At HPI, we have developed *d.store*, a customizable service platform to collect and analyze virtual collaboration activities during project runtime and in a non-interfering manner (Uflacker et al. 2010). The platform is configurable and can be utilized to capture collaboration activities from heterogeneous groupware systems, generating a single record of temporal and semantic relationships between identified actors and resources. A service interface provides the functionality that is needed to explore trends and to analyze detailed characteristics in the collaboration behavior of the observed teams. A first application of the instrument in the conceptual design phases of 11 small-group distributed engineering teams has demonstrated the feasibility of our approach as well as collaboration metrics that correlate with the performance of the teams (Uflacker 2010).

In order to make the *d.store* services, and hence the data that has been collected during observations, available to a larger research community, our next iteration of the platform seeks to incorporate functionality that facilitates its integration and application in 3rd party project environments and which allows for easy sharing of anonymized activity records. In this chapter, we introduce the targeted system architecture and motivate a common approach to virtual collaboration monitoring. First, a brief overview of the *d.store* platform and its extensive application in engineering design projects is given. Then, a recent observation of virtual collaboration activities in software engineering teams is presented. In the last section, we give an outlook on the next steps towards a shared platform for analyzing and evaluating virtual collaboration activities in engineering design. We believe that this approach will stimulate relevant and rigorous findings in empirical design research and help scientists and practitioners to understand relevant factors in virtual team collaboration.

2 d.Store – Monitoring Team Collaboration

The technical foundation for this common monitoring approach is the *d.store* platform (Uflacker and Zeier 2009). It uses Semantic Web technologies to represent concepts of collaboration artifacts, such as emails or wiki pages, as ontologies.

The concepts are linked through associations, e.g. a person is linked to an email by being its sender. They are also time-annotated and, accordingly, can be put in correlation with the project timeline.

The platform relies on a set of sensor clients for data collection. Those clients are programs that can extract collaboration data from the respective tools and transform it into a format that can be processed by the *d.store*. Depending on the sensor implementation, the data collection can happen at a single point in time or continuously throughout the project.

Regardless of the sensor client implementation, all uploaded collaboration artifacts will be combined into a single Team Collaboration Network. This unified view of the data has the advantage of enabling progressive data exploration. This means that starting from simple questions regarding the usage of one collaboration channel (e.g., How many emails have been sent?) more data dimensions can be added to the queries (e.g. How many emails have been sent by Person A to Person B and contained a link to resource X?) without further adoptions of the database, since all collaboration artifacts are already present within the graph (Figs. 1 and 2).

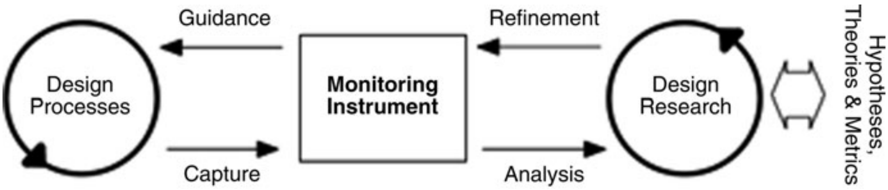


Fig. 1 The instrumentation of virtual collaboration enables “in-flight” monitoring of engineering design processes by means of computational capture and analysis of collaboration activities. New insights stimulate design research and may lead to the refinement of the instrument and its application to guide the work of engineering teams

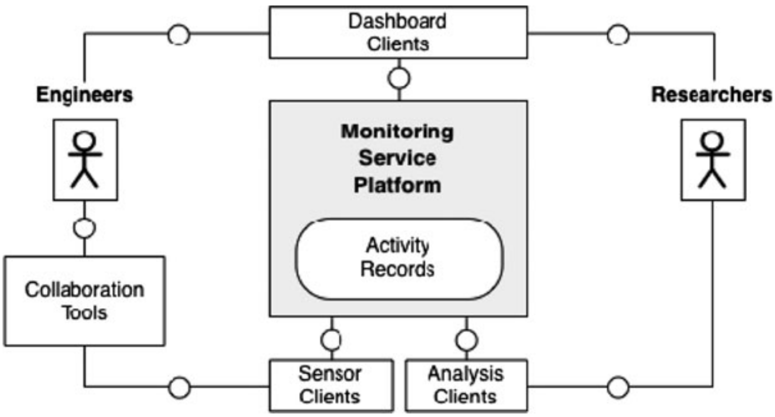


Fig. 2 A monitoring service platform provides a common interface to distributed clients that are specialized in the capture, analysis, and monitoring of collaboration activities

3 Application in Conceptual Engineering Design

Hypothesizing that high-performance design teams produce different collaboration patterns than lower-performing ones, we have applied *d.store* in the analysis of 11 engineering projects during an 8-month period of early stage concept creation and prototyping. The activities scanned from email archives, Wiki pages, and shared document folders are represented as Team Collaboration Networks and provide the basis for a detailed inspection and comparison of the teams' collaboration patterns. In particular, the system has been used to test whether the occurrences of specific patterns correlate with independent measures of team effectiveness. The teams have been ranked based on different performance criteria: the average satisfaction of team members as determined by a team diagnostic survey based on (Wageman et al. 2005), judges reviewing the project outcome, and the number of explored design alternatives.

The findings from this pilot application give first indications that performance conclusions can be drawn from virtual collaboration patterns. Patterns that correlate with independent team performance metrics can be interpreted as surrogates for 'outside-in'-driven design and team-internal information sharing. For example, a positive and significant correlation existed between the self-reported satisfaction of team members and a team's tendency to contact external process participants (e.g., end-users, customers, domain experts). This suggests that a close involvement of team-external process stakeholders has beneficial effects on a project. The results indicate that high-performance design teams share different collaboration patterns than low-performance teams, endorsing a continued utilization of the instrument to evaluate relevant performance indicators and new opportunities in the conducting of real-time team diagnostics (Uflacker and Zeier 2010).

4 A Case Study in Computer Science

To understand what differentiates Design Thinking from other methodologies, it does not suffice to solely examine Design Thinking projects. Other techniques have been used for creating innovative products, too. What sets them apart from Design Thinking? Where can we find commonalities?

In order to answer these questions, it is necessary to first take a step back and thoroughly analyze projects that applied other methodologies. The domain of agile software development is especially interesting for comparison purposes, as it shares a set of common values and principles with Design Thinking. Both approaches deal with problems that are commonly not very well understood and rather difficult to describe, so-called wicked problems (DeGrace and Stahl 1990). Constant creation of prototypes to gain feedback from end-users (Larman and Basili 2003) and a close collaboration within teams and with customers (Beck 2000), (Schwaber and Beedle 2001) are also shared practices used in either methodology. Because of those similarities in approaching comparably tough problems, it is interesting to see if and how the working styles manifest within the digital collaboration traces of the teams.

During the winter term 2009/2010, we conducted a first case study that used the previously presented toolset for monitoring the IT-mediated collaboration within an agile software development process. The project under investigation was part of the exercise of a lecture in software engineering. This educational background enabled us to control a variety of the parameters in the project, such as the development process or the utilized collaboration tools, upfront and also allowed for corrective measures later in the project.

4.1 Exercise Setup

The exercise was embedded within a third year undergraduate software engineering lecture. Due to the restricted number of students that our institute admits each year, every class consists of approximately 80 students. Participation in the course and the corresponding exercise was mandatory. Accordingly, observation results were less likely to be biased by an artificially increased degree of motivation and skill such as is usually shown by voluntary participants (Berry and Tichy 2003). This setup also ensured a comparable standard of knowledge with regards to university education. Previous working and project experience, however, varied, as shown by an upfront assessment.

Due to these differences in prior working experience as well as knowledge of the problem domain and the used programming languages or environments, the project plan accounted for a 2 week period of initial training before the beginning of the actual development phases.

Students participating in the course were close to finishing their undergraduate studies. They were required to participate in so-called bachelor projects. The project teams consisted of four to eight members. This ensured ideal team sizes for effective collaboration (Beck 2000). Students were assigned to the teams by university administration based on preference lists. Hence, the impact of friendship relations was reduced in comparison to teams formed by the students themselves. Furthermore, each project team was assigned its own working space with adequate technical equipment. Thus, problems regarding team-internal scheduling or finding suitable working areas were solved upfront.

4.2 Constraints

Dedicated workspaces and team composition were important factors since they closely resembled conditions in an industry setup. Other factors, however, could not properly be reenacted. It was, for example, not feasible to provide the students with a salary. Grades served as a substitute for this kind of compensation.

Secondly, the project was missing the pressure of having to ship to a certain target market within a predefined amount of time and with a required minimal set of functionality. Even though the project duration was limited by the time constraints of the university term, the impact of a failed project is hard to compare with the implications it would have in a real life scenario. On the other hand, grades were tied to project success and a set of external partners was present and had genuine interest in the project outcome. This should have resulted in equal pressure to succeed.

Working time was also a constraint of the study. University rules limited the time spent on project work to 8 h per week. This induced periods of vocational adjustment on a weekly basis – an effect that would not have occurred in a full-time project. Overall project duration was limited to 13 weeks between early November and mid February.

4.3 Project Outline

The task of the project was the implementation of an enterprise resource planning (ERP) system targeted at small startup companies. Their needs are very diverse, thus the system had to include the most common modules of ERP systems: Financials, Human Resource Management, Customer Relationship Management, and Reporting. Additionally, the system had to provide a means for role management, authentication, configuration, and connectivity to external systems. Three real startup companies served as potential customers for the software.

The problem domain was unknown to the majority of the students, as their previous studies did not include a lecture on this topic. This was a deliberate choice, since it ensured that the students were not able to validate their implementation by themselves, but were forced to perform user research interviews with real domain experts.

4.4 Development Process Structure

Main ideas of Scrum (Schwaber and Beedle 2001) were used as the foundation for the team setup and infrastructure of the case study. The 78 students of the project formed 13 sub-teams that conducted their work over the course of four equally long (i.e., approximately 3 weeks) sprint cycles. Each team chose a Scrum Master that was responsible for moderating the weekly Scrum Meetings with the tutors as well as for controlling the collaboration with other sub-teams.

Six students were chosen as Product Owners (POs). Each PO was assigned to two teams, one being the team of his own bachelor project. One PO was required to be responsible for a third team due to the odd total number of sub-teams.

The main responsibility of the PO team was the requirements analysis for the system by performing on-site user research at the aforementioned startup companies. The gathered requirements subsequently were split up into 13 topics and broken down into various user stories. Upon the start of each sprint, each PO presented the prioritized list of user stories to the teams. During this sprint-planning meeting, the teams estimated the required effort for each story using "planning poker". By that, they also decided which stories were considered for development during the upcoming sprint. At the end of the sprint, POs and teams met again for a sprint review meeting to assess which requirements were successfully implemented and which needed to be carried further into the next sprint.

The students decided to conduct a weekly meeting of the Scrum Masters, or other representatives of each team, to solve the following problems:

- Coordinating the collaboration between teams that implemented mutually dependent requirements,
- Creating an institution that was able to solve problems, which affected the entire project,
- Maintaining an overview of the other teams' current work in progress.

This so-called "Scrum-of-Scrums" was performed in a fashion similar to the one used in the team-internal weekly scrum meetings.

In addition to the high-level user research conducted by the POs, the teams were required to validate their implementations with fictional stakeholders that were impersonated by members of our research group. The point in time at which the interviews should take place was not defined. Most teams performed their user research interviews either at the end of the third, or during the fourth sprint. Furthermore, the teams had to perform at least one code review and refactoring session with a programming language and framework expert from our teaching staff.

4.5 Project Analysis

The focused mainly on teaching how to collaboratively work on larger programming tasks that involve multiple development teams. Therefore, the developed product was not expected to be a measure for project success. Instead, analysis concentrated on understanding how the students used the provided infrastructure and which effects certain aspects of the development process had on the collaboration behavior.

The observations were carried out on two levels. On the one hand, a team of tutors was assigned to the teams and closely monitored their weekly meetings, as well as all other appointments prescribed by the development process. On the other hand, we used the *d.store* for analysis of the digital collaboration traces created by the students.

The original implementation of the platform provided support for emails, wiki pages, and shared document folders. Thus, in order to account for the special requirements of the project, ontologies for the domains of source code management systems and bug tracking items were created. In conjunction with the respective sensor clients, this enabled us to capture:

- 2,488 source code revisions
- 1,068 emails (excluding messages from the CI system)
- 303 tickets (with a total of 1,048 status changes)
- 92 wiki pages

These artifacts were uploaded into the *d.store* and aggregated into the overall collaboration network that reflects the collaboration behavior of the development teams with regards to digital tool usage.

It must be noted that the following statements cannot be generalized and have no direct, statistically significant correlation to project success. The purpose of this discussion is to demonstrate that digital collaboration artifacts can provide indicators for certain developments within the project teams that otherwise could only be detected by an extensive manual observation process.

4.5.1 Version Control System

Most students reported that the version control system was a major source of problems that interfered with the progress of the project. Because team sprints were synchronized and the project only used a single trunk in addition to the 13 distinct branches created for the teams, merging data back and forth between the branches was often the source of conflicts.

Data analysis revealed that over 95% of all commits to the trunk that contained the word ‘merge’ happened in a time span of 2 days around the end of the sprints and 64.8% of those commits resulted in an unstable system that could not be deployed to an integration server. During the meetings with the students, we tried to analyze those situations and it became apparent that bridges to decentralized source code management systems (e.g. Git[1]) for team-internal development were secretly used and caused many of those issues. Additionally, many students felt insufficiently prepared to work with Subversion[2], even though a designated tutor for that topic was available.

Interestingly, the suggestion of a continuous workflow with more frequent merges to the trunk was acknowledged by the students to be a possible solution

[1] http://www.git-scm.com

[2] http://subversion.apache.org/

for most of the problems. However, in comments made during a variety of retrospection meetings, it became clear that even though this style of working would be true to the textbook, it would collide with the working reality of the students. Multiple parallel assignments in different courses forced them to prioritize their tasks very rigorously and mostly postpone work until the last possible point in time. This led us to the conclusion that instead of trying to force project members into working styles that they are not going to adopt, anyway, we should try to optimize the toolset in order to suit their needs.

4.5.2 Bug Tracking System

The purpose of the bug tracking system used within the project was to reflect the current state of implementation work and give developers, POs, and the teaching team feedback which, in turn, could be used for sprint planning. In order to live up to this standard, continuous upgrades of tickets statuses (i.e., opening, closing, or updating a ticket with latest development progress) are required. Figure 3 shows that these updates only occurred in clearly visible time clusters, mainly near the start and end dates of sprints. This could either indicate that development was not continuous or the tickets were simply not updated on a regular basis. By adding the dimension of source code revisions to the data analysis, the latter assumption can be explored further.

In the beginning, the curves mostly align, with the exception of some bursts in the revision count and a nearly constant deviation between both lines. Both phenomena are results of the aforementioned problems with the version control system and reflect merge revisions or fixes of merge errors. Therefore, more revisions are needed to implement certain functionality merging, or revisions are not related to

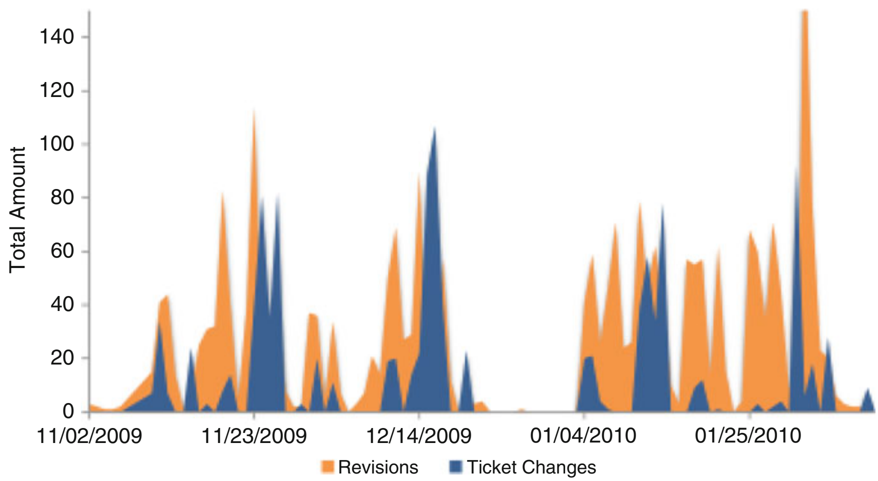


Fig. 3 Number of ticket changes and source code revisions per day

any development task but only fix errors created through merges. As the project continues, however, the alignment between the graphs decreases substantially. Asked about this particular pattern, students indicated that near the end of the project, they did not pay close attention to the ticket system anymore, but were only focused on delivering the required functionality on time.

Still, this does not explain the clearly visible clustering in the beginning of the project. When asked about the reasons for this behavior, the students stated that the bug tracking system was too cumbersome to use and, hence, they only updated the tickets in a bulk procedure.

This often resulted in a discrepancy between the state of the tickets and the actual development progress. Thus, POs and the teaching team had to find other means to obtain this information. Some statements in the final questionnaire indicated that a simple and easy to use bug tracking system would have been utilized much more frequently, at least by some of the students.

These two small examples show that the collected data is a viable means to gain non-trivial insights into the collaboration behavior of the teams and can be used as a starting point for corrective interventions. Interestingly, a lack of data is also an indicator for possibly unanticipated aspects of team collaboration. The relatively low amount of emails generated by the teams – less than one per day per team member – turned out to be such an indicator. Surprised by the lack of data, further investigations revealed that most communication was performed through other channels, such as Facebook, IRC, telephone, or in direct conversations.

4.5.3 Summary

In summary, it can be noted that this initial experiment illustrated the possibilities offered by the analysis of digital collaboration traces. Just by analyzing the ways in which the students used the given toolset, we were able to identify that some tools are obsolete, some require further assistance, and that some who were initially intended to only act as supportive measures turned out to be central elements of team collaboration. Furthermore, the analysis gave insights into detrimental working processes without these necessarily being addressed or project members being consciously aware of them.

5 Next Steps Towards a Shared Platform for Virtual Collaboration Monitoring

As demonstrated in the presented case studies, the analysis of digital collaboration traces is a valuable method for creating insights into the working behavior of project teams in different settings. In conjunction with success metrics it is even

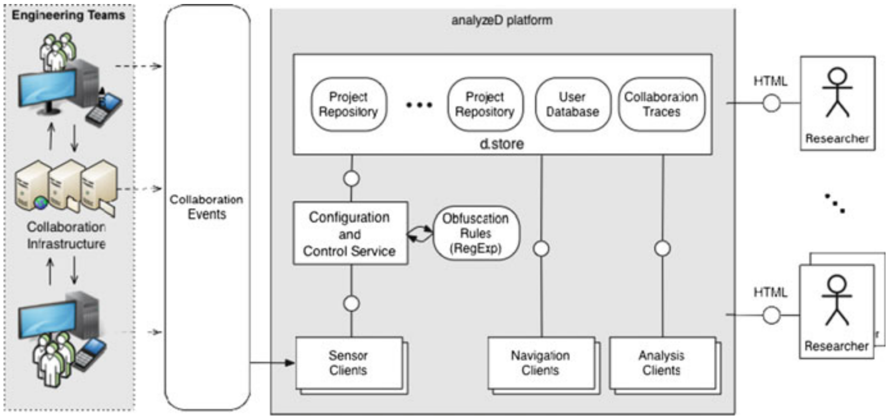


Fig. 4 Basic architecture of the proposed platform

possible to identify patterns of collaboration that indicate project success very early in the process.

However, without having access to similar data from comparable projects it is impossible to generalize the findings beyond the scope of the analyzed case studies. This leads to an important question: **How can researchers be enabled to compare virtual collaboration in different project settings and share the data and analysis results?**

We are trying to answer this question by creating a shared platform for the collection, analysis, and sharing of digital collaboration data and results from their analysis. The basic building blocks of the platform are depicted in Fig. 4.

The system will enable researchers to use a set of readily accessible services to convert and upload the collaboration traces of their respective projects. This data can, in turn, be scrutinized through a set of analysis clients. While those use-cases were also possible with the *d.store* platform, the newly created system will also build up a database of projects that can be used for verifying the conclusions drawn within single project analysis. Furthermore, the system will allow the sharing of analysis results in a way, that the occurrence of certain collaboration patterns can be automatically detected in newly uploaded collaboration networks. Thereby, it can serve as a project management dashboard.

5.1 Platform Requirements

The following requirements emerged while using the existing tools in the previously described case studies, as well as through feedback we received from other researchers in the field of virtual collaboration analysis.

5.1.1 Data Upload

Simple data upload is an essential feature of the platform as it marks the first point of contact that researchers have with it. If setting up the respective sensor clients for their project environment is more cumbersome than creating small capturing tools by themselves, researchers are less likely to use the platform and thereby contribute to the intended database of collaboration traces.

We have created a basic set of sensor clients that covers a variety of collaboration tools. For example, support is already available for two different email archive formats, three types of wiki solutions, two source code management systems, and two bug tracking systems. This set, however, can only be a starting point. Digital collaboration tools are available from numerous different vendors and new products are being developed continuously. Therefore, we have created a very simplistic programming interface that enables potential users to extend the platform with new sensor clients. Instead of creating such tools only for their own use, they are capable of developing services that can also be used by other researchers. Thus, the system is able to remain up-to-date with the evolution of new groupware tools.

5.1.2 Data Analysis

Concerning the analysis of the created Team Collaboration Networks, it became apparent that the existing SPARQL interface of the *d.store* platform is very well suited to perform arbitrary queries on the data and, thereby, test hypotheses one could have about the collaboration behavior of the examined teams. Yet, with regards to data exploration, this interface has proven to be rather unintuitive. Other solutions, such as natural language queries based on the available ontologies (Embley and Kimbrell 1985) might simplify data evaluation for researchers without a technical background. Furthermore, graphical data representations, as for example shown in Fig. 3, could lead to the detection of interesting collaboration behavior through visual perception of anomalies in the graphs.

5.1.3 A Shared Project Database

As stated earlier, core functionality of the proposed system is the creation of a database of collaboration traces. In a nutshell, this means that all collaboration data uploaded to the platform is stored permanently along with additional information about the projects that it was recorded at. If other researchers want to verify findings they made within their own datasets, they can use this database to select a suitable sample set of projects and run the corresponding queries against all stored team collaboration networks.

Research and implementation effort for this requirement has to cover various aspects. In order to be able to select suitable sample projects for the questions at hand, ample data about the projects needs to be available. Problem domain, team size, team structure, project duration, or success metrics are just some examples of potential parameters. More might come to the forefront once the system is in use. Thus, the implementation needs to be flexible with regards to the collected information, yet enforce the input of a minimal set of parameters.

Also data privacy is an issue that requires close attention. The stored collaboration data is of sensitive nature as it allows making assumptions about the working behavior of individuals. However, without further context those assumptions might not properly reflect project reality and lead to wrong conclusions. Therefore, only the original uploader of certain data can see the original names, addresses, and contents, while all other people accessing the content will only be able to see obfuscated data.

5.1.4 Outlook

The proposed system is intended to be a support for researchers as it gives them access to collaboration data from a variety of projects without requiring them to perform the underlying case studies or experiments by themselves. By that, hypotheses about possible detrimental or beneficial collaboration behavior can easily be tested on large sets of independent projects.

Beyond that obvious benefit, the platform can be extended to become a real-time project management dashboard. Therefore, it is necessary to conceive ways for defining certain collaboration patterns that had negative impacts on the success of multiple projects in the past. Those patterns, in turn, can be detected within collaboration networks during project runtime and provide project members with real-time feedback about their current collaboration behavior.

5.2 Initial Applications

To test the conceived system within real projects, two major case studies are currently planned. Both will test different aspects of the platform and provide valuable feedback for further developments. Both case studies do not explicitly require the platform to be run ‘as-a-service’. Nevertheless, development efforts in this direction will continue simultaneously.

Building on experiences gained during the first case study in a software engineering lecture, the platform will be used again in such a setting. Contrary to the first installment, the exercise will feature two development teams that compete with each other in developing a customer relationship management system. Both teams are split into eight sub-teams and equipped with the same groupware tools. By that, comparability of the respective development teams is greatly enhanced, and the case study provides us with the possibility to perform similarity analysis for the resulting team collaboration networks.

A second application is intended to explore the design workflows of users of computer aided design (CAD) systems and put Design Thinking principles to the test. Previous approaches to capture the workflow of CAD designers have struggled with a lack of semantics in the rare sequence of steps contained within the logs (Casotto et al. 1990). By adding additional information about change severity and a rating for the respective iteration to our model, we are able to not only analyze what is being done, but also how it is being done and how successful certain styles of working are. The metrics for those measurements will be developed in close collaboration with CAD-designers of prospective partner companies, as well as engineers of CAD software vendors. A test bed is provided by the CAD installations at Stanford University Product Realization Laboratory, part of Mechanical Engineering. This case study will create Team Collaboration Networks for a multitude of potentially different projects. Thereby, the platform's capability for project parameter encoding and the retrieval of sample projects for a given set of parameters will be tested.

6 Conclusion

"Every Design is Redesign" is one of the basic principles of Design Thinking. It implies that all new designs take the best of previous work, overcome its flaws, and, finally, add new aspects that provide additional value. In this chapter, we have shown that this principle is not only valuable within Design Thinking itself, but also in the research about this topic.

The foundation for our work is the *d.store*. This platform allows real-time capture and analysis of multi-modal team interactions in technology-enabled design spaces. It was successfully applied in multiple case studies in the field of engineering design and revealed correlations between the digital collaboration behavior of project participants and the overall project outcome. The extensible platform architecture also allowed for applications in other domains, such as software engineering.

While both case studies provided insights into the dynamics of virtual team collaboration in the respective domains, they shared one common limitation: It remained unclear how to confirm or even generalize the gathered results. Based on this problem, the concept for a shared platform for virtual collaboration monitoring in design research emerged. This platform is intended to provide even researchers without a strong computer science background with simple tools for capturing and analyzing digital collaboration artifacts in almost arbitrary problem domains. The real benefit of the platform, however, stems from its deployment as a service. It ensures that the collaboration data of every analyzed project is not only stored in a local database of the respective research institutions, but globally available for all researchers using the platform. Thus, each project extends the overall set of available data for testing assumptions about beneficial or detrimental collaboration behavior.

We have presented two applications that are currently planned for the platform and that will help us to verify the technical feasibility of certain aspects of the implementation. Beyond that, however, it is crucial that researchers within and beyond the Design Thinking Research community will also use the platform. With every newly added project, the platform transforms from just being an aid for data collection and evaluation purposes to a shared database that allows to reinforce assertions about Design Thinking principles by testing them on sample set sizes that were previously unattainable for individual researchers.

References

Beck, K. (2000). *Extreme programming explained: embrace change*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

Berry, D. M., & Tichy, W. F. (2003). Comments on "Formal Methods Application: An Empirical Tale of Software Development". *IEEE Transactions on Software Engineering, 29*, 567–571.

Casotto, A., Newton, A. R., & Sangiovanni-Vincentelli, A. (1990). Design management based on design traces. *DAC '90: Proceedings of the 27th ACM/IEEE Design Automation Conference*, (pp. 136–141). New York, NY, USA.

DeGrace, P., & Stahl, L. H. (1990). *Wicked problems, righteous solutions*. Upper Saddle River, NJ, USA: Yourdon Press.

Dixon, J. R. (1987). On research methodology towards a scientific theory of engineering design. *AI EDAM, 1* (03), 145–157.

Embley, D. W., & Kimbrell, R. E. (1985). A scheme-driven natural language query translator. *Proceedings of the 1985 ACM thirteenth annual conference on Computer Science* (pp. 292–297). New Orleans, Louisiana, United States: ACM.

Larman, C., & Basili, V. R. (2003). Iterative and Incremental Development: A Brief History. *Computer, 36*, 47–56.

Schwaber, K., & Beedle, M. (2001). *Agile Software Development with Scrum*. Upper Saddle River, NJ, USA: Prentice Hall PTR.

Uflacker, M. (2010). *Monitoring Virtual Team Collaboration: Methods, Applications, and Experiences in Engineering Design*. Hasso Plattner Institute for IT Systems Engineering, Potsdam, Germany.

Uflacker, M., & Zeier, A. (2009). A platform for the temporal evaluation of team communication in distributed design environments. *CSCWD '09: Proceedings of the 2009 13th International Conference on Computer Supported Cooperative Work in Design* (pp. 338–343). Washington, DC, USA: IEEE Computer Society.

Uflacker, M., & Zeier, A. (2010). A semantic network approach to analyzing virtual team interactions in the early stages of conceptual design. *Future Generation Computer Systems , 27* (1), 88–99.

Uflacker, M., Kowark, T., & Zeier, A. (2010). An Instrument for Real-Time Design Interaction Capture and Analysis. In C. a. Meinel (Ed.), *Design Thinking: Understand – Improve – Apply*. Springer (in print).

Wageman, R., Hackmann, J. R., & Lehman, E. V. (2005). Team Diagnostic Survey: Development of an Instrument. *The Journal of Applied Behavioral Science, 41* (4), 373.

Communicating Meaning and Role in Distributed Design Collaboration: How Crowdsourced Users Help Inform the Design of Telepresence Robotics

David Sirkin, Wendy Ju, and Mark Cutkosky*

Abstract Design has been described as a conversation: with the problem that is being addressed, with materials and artifacts, with our colleagues and ourselves. The language of this conversation is made up of words and images, actions and behaviors. Focusing on the role of gesture in design collaboration, we ran two studies to explore how embodied telepresence robots, or *physical avatars*, can support better communication in distributed teams. The studies drew upon crowdsourced study participants to provide their impressions of: (1) the meaning of individual gestures, and (2) the social roles of design team partners. Distant collaborators were better understood when their telepresence intermediaries portrayed relevant gestures in concert with their facial expressions. When the avatars displayed such physical motions, teammates on both sides of the interaction were perceived as more involved in the conversation, more composed in demeanor, and more equal in stature. Our next step is to apply these requirements to the design of our next generation of field-robust communication avatar.

1 Introduction

Most of us live in a world of words. Yet we communicate with each other in ways that draw upon a cultural *language of behavior* [1] that frequently takes place outside of our conscious awareness [2]. This language includes behaviors such as gesture and movement, posture, physical proximity and body orientation, eye gaze, facial expressions and even non-verbal vocalizations. These actions support the spoken word, and influence our ability to communicate to a greater extent than is

D. Sirkin • W. Ju • M. Cutkosky (✉)
Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA
e-mail: sirkin@cdr.stanford.edu; wendyju@cdr.stanford.edu; cutkosky@stanford.edu

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_10,

commonly recognized. We draw upon them every day, without even thinking about it, to ease our abilities to think improvisationally and to speak expressively.

1.1 Scenario: A Language of Behavior

Imagine that you are at work one day and notice that two colleagues, Jonathan and Tina, are standing at the doorway engrossed in conversation. It is an animated exchange, but neither of them seems inaccessible. In fact, they both appear quite composed and approachable. Jonathan seems to have the floor most of the time.

Although you are not close enough to hear Jonathan or Tina distinctly, you are still able to assess a great deal about their discussion by reading the implicit, non-verbal, behavioral cues that they are both sending and receiving. They stand closer than they might during a casual exchange, and they face each other at just about a 90° angle, almost shoulder to shoulder. They wave their arms around with dynamic, but careful movements, to emphasize what they are saying.

Then Jonathan hands a small object to Tina. It must be fragile, given the way that they each hold and support it. Tina starts to point at one side, and then the other. You cannot make out their words above the din of your design loft, but you soon realize that they are discussing an addition to the prototype that you all worked on yesterday.

You decide to join the conversation, as you may have an opinion or advice to contribute. But how do you wheedle your way in? You walk over, place yourself alongside both of them, and lean slightly inward. No response yet. You cough softly, alternate your gaze between whoever is speaking at the moment, and wait for one of them to look over, or to pause for just the right amount of time, so that you can speak up. After a few moments, Tina does look over, and asks what you think about the prototype.

1.2 Embodied Communication at a Distance

Until very recently, video has been the only way to convey such non-verbal information between each other at a distance. Video chat and conferences—some at very high resolution—are the tools of real-time face-to-face distributed collaboration. But chat does not provide the full breadth of rich, gestural content that we exchange in our collocated interactions, because the images typically only display someone's head and shoulders. And while videoconferences do capture a greater range of behaviors, remote collaborators become both literally and figuratively *two-dimensional*. That is, they cannot move around the physical space where their distant team is located, so their presence may readily be forgotten. They cannot reach through their screens to touch people, or to reference and manipulate objects within that space, so their ability to co-design is compromised. Participants on both

Fig. 1 An embodied communication avatar. The *left panel* shows the avatar in use during a distributed team working session. The *right panel* shows the avatar's screen, which carries the video feed of a remote participant, and its articulating neck, which is mounted to a stable base. These components represents a person's head, neck and upper torso, respectively. An expressive robotic arm, which the remote participant can control to point, wave, touch, and even tap against the table, is attached to one side

sides of the conversation lose the ability to quickly glance over to understand where their distant partner's attention is focused.

To address these shortcomings, we have been developing physically embodied, remotely actuated, robotic prototypes for use during distributed team collaboration [3]. These hybrids draw upon the advantages of both live video, which communicates individual facial expressions and vocal characteristics, and personal robotics, which manifest an embodied, physical presence. The devices allow team members who cannot be physically present to extend themselves into the group's workspace (shown in Fig. 1). Remote participants can control components that represent their heads, bodies and arms—using motion-sensitive interfaces—so that they can move around, gesture expressively and point at shared artifacts [4]. They can therefore be considered *physical avatars*, personifications that represent particular individuals in another place.

We are employing these prototypes in two related studies that explore the role of avatar-enabled non-verbal communication in distributed design activities. The first study focuses on the basics of how people interpret gestures presented by a remote participant who is communicating by robotic avatar during a conversation. The second study shifts focus to how people infer the personalities of remote and local participants whom they observe interacting during a distributed design task.

Our particular interest during these explorations is communication within *hub-and-satellite* teams. These teams are characterized by an individual, or a small number of individuals (the satellites), at locations that are distinct from each other, as well as distinct from a single group of collocated team members (the hub). This form of communication is becoming more common as workgroup members increasingly travel away from their usual office, or work from home, and it is the form that we find most often in our own (globally distributed) design teams. As such, it has informed the designs of our studies and prototypes.

The language of behavior carries with it culturally shared expectations about how people interpret each other's embodied social cues. For example, a shift in body position to lean inward can represent interest and a willingness to engage in conversation. But even with a shared awareness, we do not yet know if the gestural messages that are so familiar during face-to-face interaction will be as powerful and recognizable across a technology-mediated channel. Combining collaborators' personal video feeds with distinct, moving, embodied forms may significantly alter the way that other people perceive their intentions and actions.

2 Crowdsourcing

To extend the reach of our studies to a broad base of participants, we leveraged Web-based crowdsourcing, an expression that combines the two concepts of *crowd* and *outsourcing* [5]. Crowdsourcing makes use of collaboration software to draw upon and organize the collective, but distributed, contributions of members of a particular community. It should be a familiar idea to someone who has purchased a book at Amazon or read an article at Wikipedia. In the case of Amazon, fellow purchasers create the community; members freely share reviews and opinions of available products among themselves. For Wikipedia, registered readers with expert knowledge form the community; they coordinate with each other in an informal organization to author, challenge and revise the encyclopedia's articles [6].

2.1 The Crowd in Research

Crowdsourcing is increasingly being applied to the conduct of research as well [7], in particular, for user evaluation studies such as surveys, usability comparisons and quantitative or qualitative evaluations [8]. These methods can further be applied to design—and to design research—during prototype development [9]. In the context of our current studies, crowdsourcing permits us to expose our embodied avatar prototypes to a pool of potential evaluators having a vastly greater range of demographics—including age, cultural background, location in the world and experience with conferencing technology—than would be possible by recruiting only local participants at our university research lab.

Other potential advantages of a crowdsourcing approach include the ability to reach a great number of participants in parallel, reducing the time that it takes to run a study, as well as the expense of administering that study. Such advantages do still depend on the alternative methods used to recruit participants, the payments that are made to them in either case, and differences in the infrastructure required to run the study, including supporting technology, paid assistants and physical laboratory space.

But with these advantages come potential drawbacks. Among these is the possible lack of uniformity and quality in responses from participants who have varied backgrounds and motivations for participating [10]. The same diversity in participant makeup can also include individuals with a poor grasp of the primary language of the study. Even some who speak the language well may fail to recognize or understand salient cultural references within the study's content.

Because they are unsupervised and contribute from a distance, crowdsourced study participants are also more likely to remain anonymous than traditional, local participants, which can be both an advantage and a drawback. On one hand, participants are more likely to be open and honest in their responses to a study's stimuli, because they are less concerned about being judged by someone (such as study administrators or reviewers of findings) from within their local community. On the other hand, they may be better able to "game the system" (for example, through repeated participation) if their goal of financial reward overrides their interest in scientific honesty [11].

2.2 Mechanical Turk

We employed the Mechanical Turk (Mturk) service provided by Amazon to engage participants in our avatar gesture evaluations. MTurk is an online market where *requesters*, who most often have small individual tasks to be performed, connect with *workers*, who have the resources—including time, attention and labor—to perform them. Tasks are generally assignments that humans are better able than computers to perform, such as labeling images, providing opinions or—as in our case—evaluating prototypes. Requesters assign their tasks a (typically small) monetary value that they are willing to pay; workers then select which tasks to work on based on those values, along with their interest and suitability given any particular task's worker qualifications. These qualifications may exclude workers who are located within certain countries, fall outside of a particular age range, have already completed the task, or have a low rating, which is based on the their task performance quality over time.

Using Mturk meant that we had to carefully manage the ways that we both issued tasks to workers and analyzed the responses that we received. We learned early on through two pilot studies that it was important to strike a balance in the payment amount: if it was too high, responses came in so quickly that only one small geographic region had the opportunity to see the study; if it was too low, responses came in so slowly that the allotted time expired before the response set was completed. For example, a task that takes approximately 10 min to complete was originally priced at $2US, which equates to a $12 hourly wage. Depending on where in the world the Turkers who complete the task reside, the same $2 may be considered a relatively high or low payment. In our experience, depending on the time of day that we released tasks, a batch of 50 $2-payment-for-10-min tasks completed in 3–4 h.

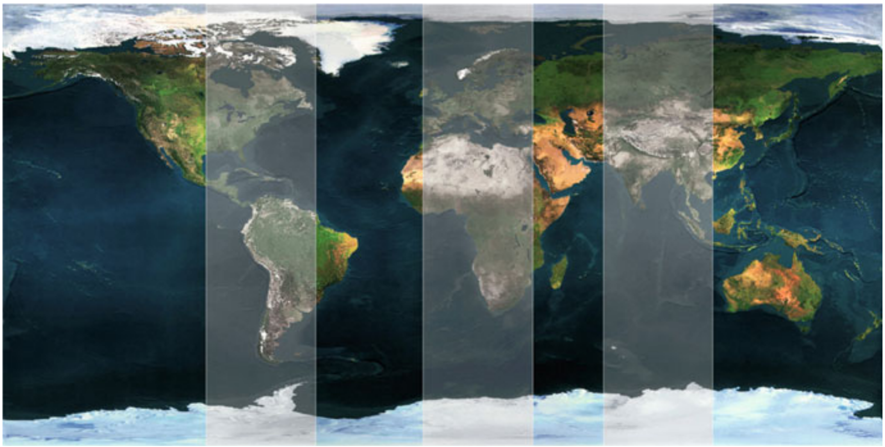


Fig. 2 An attractively priced ($12US/hour) batch of 50 Mturk tasks took 3–4 h to complete on average. We therefore balanced the number, size, payment and timing of response sets to expose our study to a selection of Turkers from around the world. This map shows the geographic coverage provided by issuing several 3-h batches. Most of our responses originated from within these time slices, plus the west coast of the United States and Australia. Of the 200 participants in both studies, one third were located in each of the United States, India and the rest of the world

We also had to manage the day and time that we released batches. Due to the relatively brief window of time to complete response sets, if a high-paying task went out during early weekday evening in India, almost all of our responses came from Mumbai, Delhi or Bangalore; similarly, if it went out on a weekend afternoon in the east coast of the United States, almost all of the responses came from the Boston to New York corridor and eastern Canada (shown in Fig. 2). We eventually found a balance between several batch sizes, prices and times of release, and recruited several hundred participants. Averaging between our two studies, about one third of our participants were located in the United States, with another third in India. The remaining third were distributed throughout 30 countries around the globe.

3 Explorations in Gesture and Team Role

3.1 Study 1: Interpreting Gestural Meaning

The goal of our initial study was to explore how people react to a range of human-avatar gestures that represent messages and expressions that are likely to occur during a typical conversation or design session. For example, leaning inward to engage someone in a new conversation, then leaning backward later to politely excuse oneself from that same conversation. But note how in another context, the same leaning inward motion could be used to focus on the words and facial expressions of one person in particular within a larger group, while leaning backward could be used to take in a broader visual view of all of the action and personalities that comprise that group. Other examples include glancing over to one side or the other, or even up or down, such as when a sketch or prototype is placed on the table in front of the team; recoiling upward and back when surprised by some unexpected update; or even glancing into the unfocused distance as one thinks carefully about a new way to approach a familiar problem. With this understanding, we can create mappings between gestures and expressions, laying out the interpretive landscape of avatar-mediated communication, and come to understand how and when certain types of robotic gestures should be used support human language or avoided because they confound our understanding.

Our approach for these behavioral studies was to create relatively simple, yet functional, prototypes, each of which allowed us to probe a single aspect of this landscape of mediated design communication with precision. A single probe hopefully provides narrow, yet deep, understanding of the questions we are trying to answer. Questions such as: do avatar motions increase the clarity or impact that the remote collaborator's message carries; are these motions familiar and welcome in a conversation, or are they mechanical and seemingly random; do they inspire more or less confident interpretations of the messages? Several of these probes, when taken together, begin to provide the understanding that a new field requires to be broadly applicable to a variety of real-world, everyday problems [12]. Problems such as: how to uniquely address one or two teammates without having to call them out by name at every turn in the conversation; or how to point out a particular Post-It note on a whiteboard spattered with dozens of others with the same color and size; or even how to look out the window to see if it is day or night at the hub team's location.

3.1.1 Video Prototypes

Designers often rely on prototypes to evaluate products or processes that they are designing without having to develop fully realized versions, especially when doing so would require more resources than are necessary at the current, perhaps early, stage of a project. Prototypes are also valuable as thinking tools, to provide insights into chosen forms, to converse with materials, or to inspire new questions to ask.

To leverage these same benefits, we employed online video prototypes during our experiments. A video prototype is a brief movie clip that demonstrates how an interactive technology—a physical avatar in our case—would perform [13]. A video prototype of an avatar-mediated conversation provides many benefits over in-person trials. It permits us to: (a) tune the avatar's motions to be as subtle or obvious, coarse or refined as we intend; (b) reproduce precise timings between the remote collaborator's onscreen gestures and the avatar's motions; and (c) repeat the exact same scenario(s) for study participants as many times as the study requires. This last point is particularly relevant, as we recruited more than 100 participants for each study, and the setup, execution and teardown of a lab experiment can be laborious and fraught with intermittent breakdowns.

This kind of prototype also has drawbacks, including that study participants lose the immediacy of direct interaction with the avatar, and instead must draw upon their prior experience and expectations to interpret the onscreen action presented to them. However, studies that compared user assessments of real-world interactive prototypes with online video studies of those same prototypes found that users responded in similar ways to the social scenarios presented, and that trends in results agreed with each other, although the two methods did not always uncover the same aspects of a design or have the strengths of effect [13–15].

3.1.2 Approach and Design Requirements

We presented a number of gestures to study participants, accompanied by a questionnaire about their impressions. Participants were asked to assume the role of partner in a conversation with the remote collaborator. Nine gestures were then shown to convey the following: uncertainty, surprise, laughter, agreement (nod yes), disagreement (shake no), thinking carefully, leaning in to look closer, glancing over to one side, and looking downward at the table. Each of these was included in three different forms: through the collaborator's facial expressions alone; through the avatar's physical motion alone; and through combined expression and motion. For the expression alone condition, the collaborator portrayed alternate emotions but the screen remained idle. For the motion alone condition, the screen moved but the collaborator retained a neutral expression.

As our study findings are still preliminary, we present early insights in the form of design requirements.

1. *Design physical gestures to complement onscreen expressions.*
   The first row of images in Fig. 3 shows Eric, a remote collaborator, peering forward closely, with the avatar's screen remaining idle. It is clear that he has heightened his attention, but is he looking *through* the screen toward us, or *at* an open window on his local computer? The sequence is ambiguous, because it can be read both ways. The second row of images shows the avatar's screen moving forward and down, but Eric does not show any change in attention or emotional state. It is difficult for us to interpret this action, because we want to read some kind of meaning from Eric's facial expressions, but he provides no visible cues. The third row of images is a simple overlay of the two preceding sequences, but in combining the facial expression with the physical movement so that that they complement each other, the action *makes sense*. Participants were more correct in their interpretations of the latter expression-plus-movement condition than they were for either of the two preceding actions. They were also more confident in their interpretations, and felt that the intended message had a greater impact upon them.

   However, these benefits came at a cost: participants found that the addition of avatar movement to the collaborator's expressions also made the gesture appear

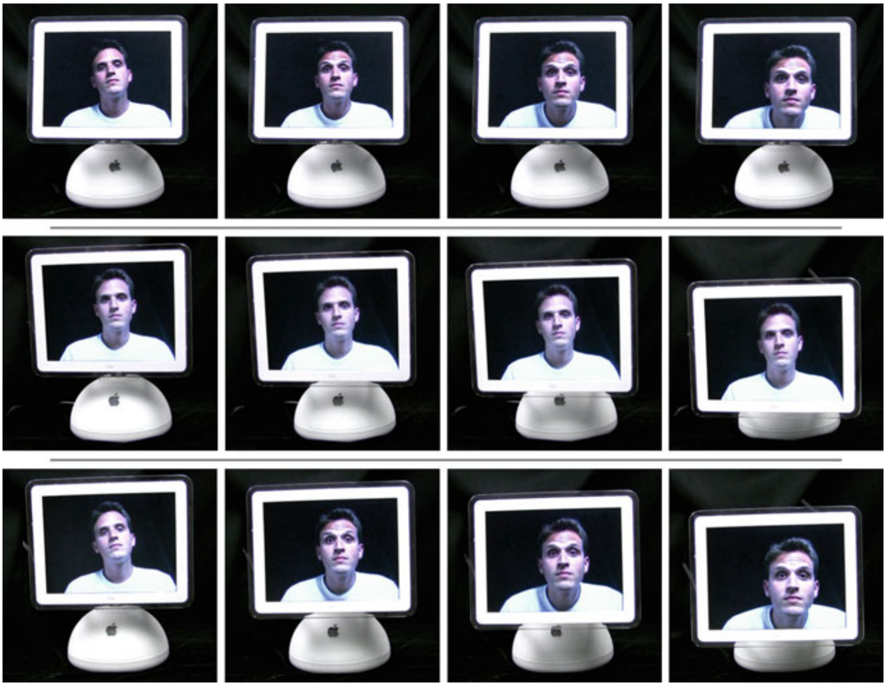


Fig. 3 Key frames taken from clips of the Study 1 video prototype shown to study participants. Images progressing across the page from *left* to *right* span about 5 s of the gesture *lean in to look closer*, showing the male actor. Images progressing down the page represent the same frames, but for each of the three different versions of the gesture. From *top* to *bottom* are: facial expression alone (the actor portrays emotions but the screen remains idle); avatar motion alone (the screen moves but the actor retains a neutral expression); and combined expression and motion (the actor portrays emotions and the screen moves) conditions

less natural. For our current avatar design, there is a tradeoff between clarity and impact on one hand, and naturalness on the other. Alternative designs may exhibit this tradeoff to greater or lesser degrees, so developers of embodied telepresence systems should be aware of its potential to influence effective communication.

2. *Physical gestures should not accompany all onscreen expressions.*
   Our study participants' abilities to interpret the meanings of gestures varied not only from one physical motion condition to another, but also from one gesture to another. That is, certain forms of body language that the remote collaborator can invoke are more readily understandable than others. For example, nodding one's head up and down, with short movements and in quick succession, is a clear indication of agreement in most western cultures. Such gestures, by their nature, are outwardly communicative, and participants can recognize them even when no face at all is displayed on the avatar's screen. But other gestures are more difficult to interpret. For example, unconsciously moving one's head to the side,

or facing down toward the floor, while immersed deep in thought. Such head motion is more reflective of internal, thoughtful states, and participants can infer many alternative meanings that are equally plausible. In contrast to the more communicative gestures such as nodding, adding physical movement to the onscreen facial expressions in these cases can further detract from participants' understanding. Therefore, some form of manual control—perhaps operated by the remote collaborator—or intelligent interface is required, to recognize the collaborator's state or actions, and determine whether or not the avatar should be set into motion.

3.2 Study 2: Observing Team Roles and Interaction

The goal of our follow-up study was to explore how people interpret the roles and personalities of distant collaborators who are interacting through a proxy with their colleagues in an active, team-based scenario. For example, the person speaking during a face-to-face group meeting can usually pose a question to someone in particular without having to call her out by name, just by turning to face her, perhaps also making eye contact. In doing so, the flow and context of the meeting, as well as the characteristics of that gesture, influence both what the speaker intends and what the person addressed infers. A sharp turn, accompanied by a pitch of the head downward, may imply a more dominant affect, whereas a slower turn, with a slight tilt of the head to the side, may indicate a more inquisitive, friendly demeanor. Can these means to direct the flow of conversation, and to convey cues about team role, be communicated as effectively through an embodied avatar? If so, with this understanding, we can design avatar motions that relay someone's current emotional state, or the state that they prefer to reveal, even if it does not reflect their true current condition.

As before, we turn to video prototypes as a means to reach a culturally diverse group of study participants. In this case, the videos depict a distributed design team interaction scenario with several hub teammates and a single remote collaborator, who joins the session through an embodied proxy. Given our focus on peer-to-peer interaction, we need measures of how well individual team members are perceived as collaborators, and whether the roles that they assume support or weaken relations with their colleagues. Examples include whether individuals can express an actively engaged or detached personality through an avatar, or if they can communicate leader or follower roles. And if so, do these traits appear more pronounced when they are paired in different ways?

3.2.1 Relational Messaging Scale

For these measures, we turn to the Relational Messaging Scale [16]. This scale is used by social researchers to describe how people relate to each other during an interaction, rather than the content of that interaction. It introduces two indicators, dominance and affiliation, and the balance between them. Dominance can be considered the extent to which one person in an interaction tries—or appears to try—to control another, so its scale runs from dominant to submissive. Affiliation is the extent to which one person regards the other favorably, so its scale runs from liking to disliking.

But these are rather broad categories, so the two scales are broken down into smaller, more manageable concepts that we can measure using a study questionnaire. These concepts include: immediacy, the extent to which a person is involved in a conversation; affect, the sense that the person is interested in continuing the conversation or deepening the relationship; similarity, whether the conversational partners are friends, or members of the same social group; receptivity, the degree of sincerity, openness and trust between the partners; composure, how calm and poised a person appears; formality, which ranges from formal to casual; dominance, the degree to which one person persuades or influences another; and equality, which describes whether one person esteems another at a higher, equivalent or lower level.

3.2.2 Approach and Design Requirements

We created four different versions of a collaborative design session and recorded a video of each of them. The videos depicted the same scene, dialogue, participants and overall sequence of action. We showed (only) one of the videos to each study participant, accompanied by a questionnaire about the action that took place. The storyline had a remote collaborator asking one of his local hub teammates for help re-thinking the form of remote control that they had recently designed together. After some friendly banter, one of them calls over another hub teammate for further input. A short while later, they all come up with an improved design by working together. We changed only two characteristics of the participants between versions, which we chose because of their relevance to the relational scale. The first was whether the remote collaborator's physical avatar moved along with his actions, or remained idle for the duration of the session. The second was the degree to which either the remote collaborator, or the hub teammate, took a dominant role in leading the session.

Building upon the insights from the initial study, we present findings from our follow-up study as another set of design requirements.

1. *Include physical gestures to convey greater involvement.*
   The first and second frames in Fig. 4 show Eric and Becky during the same exchange within the course of the scenario, with the difference that the avatar remains idle in the first frame, and able to move in the second. Note how in the first, Becky must lean over to both see and be seen by Eric, whereas in the second, she can interact more naturally because he has turned around to face her. Actions such as this, as well as the ability to wave and point (shown in the third frame, which now includes all three participants), led study participants to rate

Fig. 4 Key frames taken from clips of the Study 2 video prototypes shown to study participants. The scene depicts a hub-and-satellite team collaborating on the design of a remote control. At the *top*, the collaborators (humorously) compare hand sizes during the facial expression alone condition; in the *middle*, they also compare hand sizes, but with embodied motion during the combined expression and motion condition; at the *bottom*, the remote participant points to certain features of the prototype

the remote collaborator as more involved and less dominant than in the case where his avatar could not physically move. This makes sense, as the ability to enact body language influences one's own sense of involvement in a conversation. Participants also rated the local teammate as more equal to her partner when he could perform such gestures.

2. *Include physical gestures to equalize the roles of both participants.*
   Study participants rated the remote collaborator as more involved and composed when he took a leadership role in the discussion, compared to when he took a follower role. Examples of this difference include that as leader, he would call over a third team member for further help, ask for opinions, and make a decision about which design he preferred; as follower, he would defer to the local teammate to take these actions. But participants also rated this local teammate as more composed and equal when the remote collaborator led the discussion. These responses, along with the preceding finding of greater teammate equality during avatar motion, suggest that the addition of physical gestures serves as a moderating influence on the perception of *both* participants. One interpretation is that using an avatar that does not move—which can be considered similar to current forms of video chat—introduces a sense of inequality between the partners, and this inequality can be ameliorated by re-introducing the non-verbal cues that avatar motion enables.

4 Plans and Issues for the Future

Our next step is to continue analyzing and making sense of the extensive data series resulting from Studies 1 and 2. We plan to isolate the responses within each study by the geographic region and gender of participants, to determine if their interpretations or confidence levels differ within or between distinct groups. We will also compare the results of both studies against each other, looking for trends that may contradict or reinforce our expectations and findings so far.

Afterward, our focus will shift to applying the lessons and requirements of the two current studies to the design of the next generation embodied communication prototype. These efforts will be divided between the local and remote sides of the interaction. For the local side (the avatar itself), we will generate and compare alternative physical motions for each of our expressions, to assess which are most effective in certain contexts. Doing so will provide greater depth to the broad understanding that we now have about robot-mediated gestural communication. We are thus transitioning from our initial, coarser concerns over *what* makes an interaction successful, to subsequent, more refined questions about *how* to do so most effectively. For the remote side (the control interface), we are building a platform that will track the collaborator's gestural actions and transpose them onto the avatar. The interface will also provide a manual tracking mode, to operate during those instances where less physical motion can be more understandable.

To these ends, we plan two further studies. First, we will re-run studies 1 and 2 with a set of collocated study participants, to confirm that impressions of the avatar, and of the collaborators' gestures and roles, are similar between both video and physical presence prototypes. This requires that the next generation of avatar be robust enough to operate reliably, repeatedly and safely in the field. Second, we will conduct a new study of small, collocated teams performing a design task with a remote collaborator interacting through an embodied avatar. The collaborator will have alternative forms of gestural control—including tracking and manual settings—allowing us to explore and compare the differences between them in a real-world working session context.

5 Conclusion

Gottman states that most individuals only attend to a fraction of the communicative social cues that are available to them at any time [17]. Among these are the pace of a conversation, or someone's gestures, body postures and facial expressions. Although we are generally inattentive to much of this information, it still influences our behaviors, and our behaviors in turn influence those with whom we interact. By introducing expressive, embodied avatars that are capable of displaying such social cues, we hope to re-establish this channel of social feedback in distributed design communication and thinking.

Acknowledgments The authors would like to thank the Hasso Plattner Institute—Stanford Design Thinking Research Program for funding this project; Professors Mark Cutkosky and Larry Leifer for supervising the research; and collaborators Eric Kent for co-designing and constructing robot avatars and contributing to the content of the studies; Rebecca Currano for contributing to the content of the video prototypes and studies; and Samson Phan for guiding the design and safety of our robot avatars.

References

1. Hall, E. (1959) *The silent language*. Garden City, NY: Doubleday.
2. Argyle, M. (1988) *Bodily communication*. London, England: Methuen.
3. Sirkin, D. (2010) Physicality in distributed design collaboration: How embodiment and gesture can re-establish rapport and support better design. In H. Plattner, C. Meinel and L. Leifer (Eds.) *Design thinking: Understand, improve, apply*, 165–178. Berlin, Germany: Springer.
4. Clark, H. (2003) Pointing and placing. In S. Kita (Ed.) *Pointing: Where language, culture and cognition meet*, 243–268. Hillsdale, NJ: Erlbaum.
5. Howe, J. (2006) The rise of crowdsourcing. *Wired Magazine* 14(6).
6. Kittur, A. and Kraut, R. (2008) Harnessing the wisdom of crowds in Wikipedia: Quality through coordination. *Proceedings of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, 37–46.
7. Ross, J., Irani, L., Silberman, M., Zaldivar, A. and Tomlinson, B. (2010) Who are the crowdworkers?: Shifting demographics in Mechanical Turk. *Extended Abstracts of the*

28th Annual SIGCHI Conference on Human Factors in Computing Systems, ACM Press, 2863–2872.

8. Kittur, A., Chi, E. and Suh, B. (2008) Crowdsourcing user studies with Mechanical Turk. *Proceedings of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, 453–456.
9. Dow, S., Glassco, A., Kass, J., Schwarz, M., Schwartz, D. and Klemmer, S. (2010) Parallel prototyping leads to better design results, more divergence, and increased self-efficacy. In *ACM Transactions on Computer-Human Interaction* 17(4), ACM Press, 1–24.
10. Feng, D., Besana, S. and Zajac, R. (2009) Acquiring high quality non-expert knowledge from on-demand workforce. *Proceedings on the 2009 Workshop on the People's Web Meets NLP*, ACL and AFNLP, 51–56.
11. Downs, J., Holbrook, M., Sheng, S. and Cranor, L. (2010) Are your participants gaming the system? *Proceedings of the 28th Annual SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, 2399–2402.
12. Gaver, B., Dunne, T. and Pacenti, E. (1999) Design: Cultural probes. *Interactions* 6(1), 21–29.
13. Ju, W. and Takayama, L. (2009) Approachability: How people interpret automatic door movement as gesture. *International Journal of Design* 3(2), 77–86.
14. Kidd, C. (2003) Sociable robots: The role of presence and task in human robot interaction. Master's Thesis, MIT.
15. Woods, S., Walters, M., Koay, K. and Dautenhahn, K. (2006) Methodological issues in HRI: A comparison of live and video-based methods in robot to human approach and direction trials. *Proceedings of the 15th IEEE International Symposium on Robot and Human Interactive Communication*, IEEE Press, 51–58.
16. Dillard, J., Solomon, D. and Palmer, M. Structuring the concept of relational communication. *Communication Monographs* 66, 49–65.
17. Coan, J. and Gottman, J. (2007) The specific affect (SPAFF) coding system. In J. Coan and J. Allen (Eds.) *Handbook of emotion elicitation and assessment*, 106–123. New York, NY: Oxford University Press.

Teamology – The Art and Science of Design Team Formation*

Greg L. Kress and Mark Schar

Abstract Nearly all design work is collaborative work. The phenomenon of the "design team" is increasingly common in both industry and project-based education. Existing organizational behavior research has shown that diversity on a team has mixed and frequently negative effects, particularly when outward indicators such as gender, ethnicity, age and experience measure diversity. However, relatively little research has been conducted on the problem solving capabilities and preferences of individual team members, or "team cognitive diversity." This study examines 14 measures of cognitive diversity and 3 measures of project performance for 15 design teams comprised of 97 masters-level engineering students from nine universities in eight countries who collaborated over a period of 8 months. We find that students with similar backgrounds and experience level reveal a wide variety of cognitive problem solving preferences.

We also find that overall cognitive diversity does not appear to correlate with overall team project performance. However, team project performance positively correlates with team level "social sensitivity," the cognitive ability to relate to other team members problem solving preferences. Finally, cognitive diversity does not correlate with either individual and team level satisfaction, indicating that cognitive differences may be successfully accommodated over the life of the project. The implications of these findings are discussed.

G.L. Kress (✉) • M. Schar
Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA
e-mail: glkress@stanford.edu; mfschar@stanford.edu

* Dr. Rich Shavelson is principle investigator of this project.

H. Plattner et al. (eds.), *Design Thinking Research*, Understanding Innovation, DOI 10.1007/978-3-642-21643-5_11,

1 Introduction

As teams practice the process of design, there is a growing awareness among researchers that team diversity may play a significant role in the quality of the design output. However, there is a significant body of research that indicates team diversity has both a positive and a negative impact on team performance. On the positive side, team diversity can provide a variety of perspectives and problem-solving approaches, and the belief that this will lead to greater creativity and quality of team performance. However, most research on diversity and team performance of the evidence favors a more pessimistic view: that diversity creates social divisions, which in turn create negative performance outcomes for the group [1].

When demographic markers of diversity have been used to predict team outcomes the results of team diversity have been non-existent to negative [1]. The need for examining the underlying psychological and cognitive mechanisms linking diversity to team outcomes has been emphasized by several researchers and provides motivation for the current study [2]. Cognitive style, as described in the following section, is a factor that is closely associated with educational and functional diversity. Cognitive style can be analyzed both in terms of *variety* and *variation* within a team. To the extent that cognitive style is the underlying driver of both personal and team behavior, it becomes the signal in the otherwise noisy world of team performance.

Several recent studies have shown the power of both variety and variation in cognitive style on team performance. As a measure of variety, Baer et al. [3] used the Big Five (B5) personality type indicator to identify individuals with characteristics conducive to team creativity (high extraversion, high openness, high emotional stability, low conscientiousness, low agreeableness), creating triad teams and a time-phased creative task. They found that a team's collective high creative confidence boosted overall team creative performance.

As a measure of variation, Woolley and Hackman [4] using dyad teams measured high and low spatial and object cognitive style capability and found that teams performed better when tasks assignments were matched to their individual capabilities. In additional research, Woolley et al. [5] identified a collective intelligence of teams, showing that team performance is not affected by individual participant intelligence levels, but is moderated by a cognitive based "social sensitivity" or empathy for others point of view, as well as distribution of conversational turn-taking.

2 Cognitive Diversity and Team Cognitive Diversity

Samuel Messick, a leader in the field of cognitive style research and its impact on educational testing, defined cognitive styles by its meta-effect on behavior. Messick states that cognitive styles "appear to serve as high level heuristics that organize lower-level strategies, operations, and propensities – often including abilities – in such complex sequential processes as problem solving and learning" [6].

Maria Kozhevnikov, a Harvard neurologist, defines cognitive style as "relatively stable individual differences in preferred ways of organizing and processing information that cut across the personality and cognitive characteristics of an individual" [7]. Cognitive styles involve both the preferred skills a person brings to problem solving and the preferred behaviors they use to problem solve.

For the purposes of this study, we define cognitive style as the manner in which a person perceives information and uses that information to solve problems.

The blending of cognitive styles can become quite complex in the context of a group or team, where individual team members may bring very different cognitive skills and personality traits to the group task. Differing cognitive styles can also contribute to the development of "representational gaps" where team members see problems completely differently based on how they process information and how they prefer to solve problems [8].

There are many psychometric tools for measuring individual cognitive style and data from these instruments can be assembled to form a team view, as shown in Fig. 1. For any given cognitive style measure, it can be normed based on the specific instrument ranges (from 0.00 to 1.00) and compared to other cognitive style measures. In a team context, these cognitive style measures can then be compared across team members and combined for a team view and comparison across teams.

The first measure is a team mean score for any cognitive style measure, which is simply the sum of individual measures divided by the number of team members.

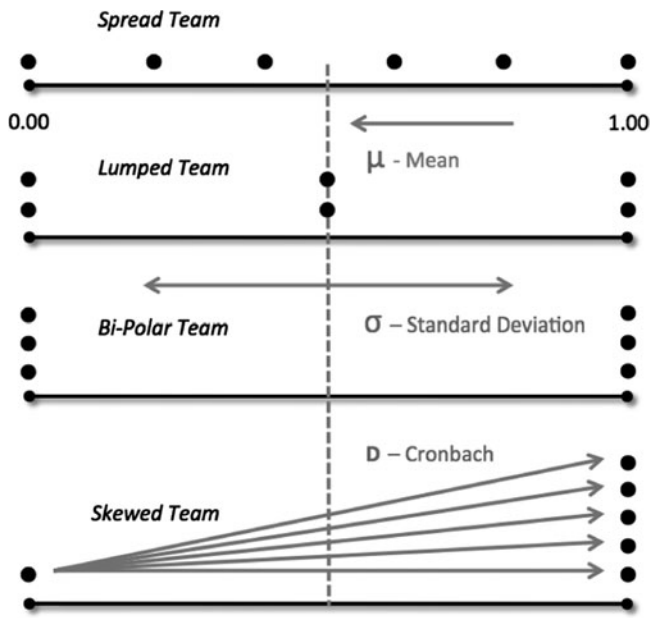


Fig. 1 Team cognitive diversity models

TeamType	Spread	Lumped	Bi-Polar	Skewed
Mean	.50	.50	.50	.83
Standard Deviation	.37	.45	.55	.41
Cronbach D	8.4	12.0	18.0	10.0

Fig. 2 Mean, standard deviation and Cronbach's D for sample teams

A second measure is standard deviation, which is a measure of the variance of scores within the team environment. The working assumption is that the higher the variance in cognitive style scores, the more cognitive style diversity exists on the team.

A third measure is Cronbach's D (*D*), which is the orthogonal absolute distance from any one score to all other scores, summed to the team level [9]. There is a positive, linear relationship between standard deviation scores and *D* scores; however, this *D* measure provides both a team measure and an individual measure, which helps to identify "outliers" within a team score set.

The differences in this scoring system can be shown through different kinds of teams. In the "spread" team, individuals are spread equally across the range of possible score, while the "lumped" team has members at opposite ends of the scale but also representation in the mid-range. These mid-range members may offer a bridge between two team members at opposite extremes. The "bi-polar" team has membership equally split between the polar opposite scores, offering the best opportunity for conflict. And finally the "skewed" team shows the effect of an extreme outlier on team measurement.

The mean, standard deviation and *D* for these four types of teams are shown in Fig. 2. The mean of scores shows the relative balance of teams with little information on intra-team diversity dynamics. In this example, the "skewed" team would have the average highest team scores on a particular measure, perhaps indicating an abundance of this particular preference, but does not reveal the presence of an outlier team member. The standard deviation of scores indicate that the "bi-polar" team has the most individual score variance, and is a good proxy for cognitive style diversity. This variance is considered a measure of "coping behavior" required by all individuals to adjust to differences in cognitive style [10, 11].

Cronbach's *D* can reveal the presence of an outlier within team scores. The *D* score of an individual outlier, compared to individual *D* scores of other team members will be significantly higher. With this measure, we can examine the effect of an outlier on team dynamics and satisfaction.

3 Research Hypotheses

Team research on diversity groups has often defined diversity as "visible diversity" and "non-visible diversity." Visible diversity includes outward signs of difference that are obvious and easy to categorize. Non-visible diversity includes education, cognitive skills and abilities, values, attitudes, and personality differences [1].

Research on visible diversity (such as age, ethnicity and gender) has shown that there is no effect, or a slightly negative effect, on team performance [12].

The ME310 student population for this design team research has little visible diversity. Students are of a similar age (early 20's), gender split is 67%/33% male-to-female and 82% of respondents are of European or North American descent. However, non-visible diversity may vary significantly as students come from at least 12 different countries, several different educational majors and may bring a range of cultural attitudes and values. These non-visible differences should lead differing cognitive styles and problem solving preferences. Therefore, we expect that cognitive style diversity will be significantly different higher among teams than any measure of visible diversity:

Hypothesis 1 (H1) – The ME310 student population will show more cognitive diversity than either age, gender or racial diversity.

Sociologist Peter Blau argues that, based purely on statistics, heterogeneous groups will result in greater contact between diverse individuals [13]. Wilde states "people who individually have only a few problem solving strategies can pool these on a good team to make it overcome any obstacle it encounters" [14]. Simply based on more approaches to problem solving within a diverse team, we propose that teams with greater cognitive style diversity will have better problem solving ability and, therefore, have better project performance.

Hypothesis 2 (H2) – The greater cognitive diversity on ME310 project teams will lead to better project performance.

There is an inconsistent link in the research between diversity and team conflict. Functional diversity is associated with task conflict, while racial and tenure diversity drive emotional conflict [15]. On the other hand, task conflict (differing approaches to problem solving) has been shown to improve a team's performance on cognitive tasks. Therefore, we propose that he reverse of this may also be true, that higher cognitive diversity will predict more team conflict and lead to lower team satisfaction.

Hypothesis 3 (H3) – The more cognitive diversity on ME310 project teams will lead to lower intra-team satisfaction.

The above three hypotheses form the basis for our theory that cognitive style diversity is an important contributor to design team performance. Below we describe the field study conducted to test our hypotheses.

4 Methodology

4.1 Background

This study examines 15 independent project teams within the context of a year-long, project-based global design innovation course. The course spanned a full academic year and was hosted simultaneously at nine international universities. Most of the project teams included students from two different universities; though the students have a handful of opportunities to meet face-to-face during the project, the majority of their work is conducted in a state of distributed collaboration. A few of the teams were "local-only" and did not have any international counterpart. The teams ranged in size from three to nine individuals (predominantly first-year Masters-level engineers, but also some students of industrial design, business and marketing, and information technology). Students were graded and evaluated as a team.

A corporate "client" who provides a brief project prompt at the beginning of the course sponsors each team. The prompts vary in scope, subject matter and difficulty and provide a rough outline for the project work. The prompt is likely to be revisited, refined and redefined through subsequent meetings between the team members and a corporate liaison. The students had approximately 8 months to develop a solution and deliver a functional proof-of-concept prototype. The design process, though undoubtedly implemented differently on each team, combines exploration, need-finding, iterative prototyping and user testing. The teams had no appointed leader or project manager, providing the potential for emergent leadership within the teams. Their work was guided by feedback from teaching team members as well as industry coaches and corporate liaisons.

There were notable differences in how the program was set up at each of the different universities. While one of the universities hosted ten projects (32 students total), six others hosted two or fewer (with a total of eight students or less). Teaching teams (including professors and teaching assistants) varied in size from zero to six members, and conducted varying amounts of weekly lecture/feedback sessions. However, all students had access to lecture videos from other universities and the course content was largely standardized. Though the major milestones and deadlines were largely shared across universities, grading policies were different in some cases. Not all universities assigned independent industry coaches to their teams. Each team was provided a research and development budget of approximately equal size (Fig. 3).

4.2 Independent Psychometric Measurement

Psychometric and ethnographic survey instruments were administered to the students during the early phases of the course. The psychometric instruments included the Kirton Adaption-Innovation Index (KAI), the Big-Five Personality Index, the Herrmann Brain Dominance Instrument (HBDI) and the Wilde-Type Teamology method. These four instruments were selected from a much larger pool of potential psychometric instruments; our selection was based on relevance of the instruments to our construct of cognitive diversity, personal experience with the instruments, extent of scientific support in the literature and variability across instruments for a multi-faceted approach (Fig. 4) [16].

Location	Students	Project Teams	Teaching Team Size	Weekly Sessions	Coaches Assigned
California	32	10	6	2	Yes
Scandinavia	23	6	6	2	Yes
France	13	3	3	0	Yes
Mexico	8	2	7	2	No
Northern Germany	6	2	3	1	Yes*
Colombia	6	2	6	Unknown	Unknown
Japan	4	1	0	0	No
Southern Germany	4	1	3	1	No
Switzerland	4	1	2	Unknown	Unknown

*Shared coach

Fig. 3 Student breakdown by location

Instrument Name	Output Variables	Survey Items	Survey Instrument
Kirton	1	32	KAI
NEO FFI	5	10	B5-TIPI
Herrmann	4	120	HBDI
Wilde-Type	4	20	W-TTI

Fig. 4 Psychometric instruments

Kirton Adaption-Innovation Inventory (KAI) – The KAI is a 32-item self self-reported instrument that returns a single preference score on a bi-polar scale ranging from "adaptor" to "innovator" [17]. Adaptors are drawn to "do things better" and exhibit behaviors like seeking "solutions to problems in tried and understood ways" and "resolving problems rather than finding them," with work habits that value "precision, reliability, efficiency, methodicalness, prudence, discipline, conformity." Kirton's Innovators are drawn to "doing things differently" and exhibit behaviors like "queries problems' concomitant assumptions; manipulates problems" and "discover(s) problems and discover(s) avenues of a solution" with work habits that "treats accepted means with little regard."

Over time, Kirton has adopted an orthogonal, second axis which is called "problem solving level," which is reflective of the experience with either problem solving style. For example "High Level Adaptors" who understand the benefit of adaptive behavior so well that they know "exactly when its most valuable to break a rule" [10]. Similarly, "High Level Innovators" who value change may best understand when it's not "wise to break certain rules."

NEO-Five Factor Personality Inventory (NEO-TIPI) – The NEO-FFI (also known as the "Big Five Factors") was developed by Costa et al. to measure personality differences. It is a five-factor model measuring neuroticism (N), extraversion (E), openness (O), agreeableness (A) and conscientiousness (C) [18]. The TIPI variation is a ten-item version of the NEO-FFI developed as an alternative where "where very short measures are needed, personality is not the primary topic of interest, or researchers can tolerate the somewhat diminished psychometric properties associated with very brief measures" [19].

Herrmann Brain Dominance Indicator (HBDI) – The HBDI provides, on the basis of 120 items, a four-factor classification of mental preferences or cognitive styles [20]. The HBDI-A factor reflects a preference for solving analytical and factual problems using logical and reason, while the HBDI-B factor shows a preference for temporal and sequential reasoning, sequencing content and the application of rules [21]. The HBDI-C factor reflects a problem solving preference for interaction with others, sensing and reacting to input from others, while the HBDI-D factor shows a preference for imaginative or conceptual problem solving, synthesizing input and viewing problems in a holistic manner.

Herrmann states that about 60% of the population have two dominate problem-solving preferences and an additional 30% have three dominate preferences. Scores in HBDI-A and HBDI-B, and HBDI-C and HBDI-D have positive correlation while HBDI-A and HBDI-C and HBDI-B and HBDI-D have negative correlation [22].

Wilde-Type Teamology Indicator (W-TTI) – The W-TTI was developed at Stanford University by Dr. Douglass J. Wilde. W-TTI uses a 20-item survey instrument that follows the format of the Myers-Briggs (MBTI) questionnaire. Using Jung's original construct theory, the W-TTI segments the results into eight cognitive style (or "mode") preferences. The eight cognitive styles are paired, resulting in four complementary modes of positive-negative polarity. According to Wilde, a positive score on the W-EN factor shows a tendency to rearranges "known concepts into novel systems," while a positive score on the W-ES factor indicated a preference for discovery of "new ideas and phenomena by direct experience." A positive score on the third factor, W-ET reveals a preference for "efficiently managing resources, decisive, imposes structure" while the fourth factor of W-EF shows a preference for "expressive, tactful builder of group morale." These modes are complemented by four introverted modes exhibiting different preferences.

Wilde's theory on team formation holds that individuals who demonstrate relative preferences for these cognitive styles comprise "affinity groups" within a given population. The strongest teams will have representation in each affinity group. Whereas some individuals exhibit many strong preferences, others can exhibit no particular preferences; both are examples of flexibility in thinking style. Wilde also believes that teams work best when all members know their preferences, share their preferences with others and adopt team roles that support their problem solving preferences [23].

4.3 Demographic Data

In total, nine points of standard ethnographic data (age, gender, etc.) were captured in addition to 14 psychometric variables. These 25 data points for each individual, as well as fixed team parameters (team size, project difficulty, etc.) comprised our independent variable data set.

4.4 Project Difficulty Rating

Due to the variability of scope, content and complexity in the initial project prompts, it was necessary to establish a difficulty rating that could be used to adjust the final project performance scores. Adjusting by overall difficulty allowed us to make more valid performance comparisons across the different projects. Project difficulty was rated by assessing the technical complexity, breadth of scope, extent of ambiguity and overall difficulty of each project.

The assessment was conducted by a sampling of 32 design professionals with an average of 12 years' experience in the field. The inter-rater agreement was ($\alpha = .81$). Two additional projects were inserted into the rating, one deemed easy and one hard by the researchers, to check the validity of the ranking. The ratings had an internal consistency ($\alpha = .839$), so we used to the mean of the four measures as the numerical difficultly rating for each project. The overall project difficulty ratings were proportionately adjusted so that the easiest project had a rating of 1.00, while the most difficult project had a rating of 1.41, reflecting a 41% difference in difficulty ratings between the easiest and hardest projects.

4.5 Project Performance Rating

Project performance was assessed according to four performance variables and three process variables. The performance variables were: proposed solution effectiveness at addressing the stated need; usability of the solution were it to be implemented as a real product; technical feasibility of the solution to be implemented as a real product; and originality of the proposed solution. The process variables examined whether the team redefined the problem statement as given by the prompt, whether their final prototype successfully demonstrated the proposed solution, and whether the stated need was compelling.

The researchers reviewed all project materials and were trained in rating and rating agreement. After rating, the internal consistency among the performance variables was .639. As a result we opted to use the sum of all four-performance variables as the raw project performance score. Process variables were tracked but not immediately included in the analysis. Prior to adjusting for difficulty, the range of scores obtained was 2.0–8.0 (of a possible 0.0–8.0 integer range). Project performance ratings were multiplied by the project difficulty scores to produce the final adjusted performance rating. These ratings ranged from a low of 2.07 to a high of 10.85.

4.6 Team Satisfaction Rating

At the conclusion of the projects, students were asked to complete a ten-item satisfaction survey. This survey explored conflicts and conflict resolution within the team, shared goals, liking and closeness of team members, emergent leadership, teamwork skills and general satisfaction with the team's performance. All questions were administered on a Likert-style scale except for the Aron Scale of Closeness item and two open-form questions about leadership.

To determine whether there had been emergent leadership within the team, students were asked to identify which team member (if any) had taken the leadership role and also to make some qualitative remarks on their leadership style. The condition for emergent leadership was the case where two thirds or more of all team members identified the same individual as leader.

4.7 Team Discussion Sessions

Discussion sessions were conducted with six of the project teams at two different universities at the approximate midpoint of their project work. These interviews explored problem solving, working and thinking-style diversity, conflict, conflict resolution and communication within the teams. The list of questions was fixed for all sessions, though open-ended responses were encouraged and additional topics would be pursued if they arose. Audio recordings were made of all sessions and transcriptions were prepared afterward. The aim of these sessions was to capture firsthand accounts of team dynamics experiences; these impressions could later be used to understand how cognitive effects manifest themselves in actual teamwork.

5 Results

5.1 Subject Diversity

The subject population for this research was comprised of masters-level engineering design students, which as a self-selecting group may have means that are skewed from general population measure norms. This is particularly true for visible signs of diversity, like age and gender. For example, the mean age of the sample population was 24.1 years and the data were not normally distributed. There was a positive skew as the vast majority of participants had ages in the early twenties, with none younger than 22 and only three subjects older than 28 years.

On less visible demographic variables there was a mixed level of diversity. For example, most of the sample had limited work experience as 88% of subjects had less than 3 years work (non-academic) experience and 47% of the subjects had

	Research Sample		Instrument Standard			Research Sample		Instrument Standard	
	$\bar{x}$	s	μ	σ		$\bar{x}$	s	μ	σ
KAI	**.549**	**.098**	.491	.088	**HBDI-A**	**.541**	**.168**	.454	.142
					HBDI-B	**.411**	**.111**	.437	.127
B5-N	.666	.221	.638	.188	**HBDI-C**	.405	.143	.418	.122
B5-E	**.639**	**.213**	.567	.187	**HBDI-D**	**.471**	**.145**	.393	.113
B5-O	**.792**	**.146**	.730	.145					
B5-A	**.586**	**.196**	.705	.150	**W-EN**	.488	.128	N/A	N/A
B5-C	.703	.188	.733	.179	**W-ES**	.515	.105	N/A	N/A
					W-ET	.617	.140	N/A	N/A
					W-EF	.493	.113	N/A	N/A

Fig. 5 Mean and standard deviation of psychometric instruments. All instruments have been normalized to the instrument ranges for comparison purposes. Bold indicates research sample significant difference from instrument standard ($p < .05$)

no work experience. Similarly, there was less diversity in academic experience, as 59% of the subjects reported "engineering" as their primary field of study, 23% of the subjects reported "design", 16% "business" and only 2% "other."

Culturally, the sample was more diverse. As mentioned earlier, students were drawn from nine universities in eight countries and reported over 12 countries of origin. One psychometric instrument was offered in 17 different languages and the sample population chose to use eight of these languages, indicating significant language differences. For those instruments which were offered only in English, it is worth noting that skill in English was a prerequisite for the course at all locations; we believe all respondents were proficient enough in English to respond accurately.

The psychometric instruments showed a broader range of cognitive diversity in the sample population, as shown in Fig. 5. The research sample had a higher mean score on "emotional stability" (B5-N), extroversion (B5-E), "agreeableness" (B5-A), and "innovativeness" (HBDI-D) and a lower mean score on "planning" (HBDI-B). Importantly, the standard deviation of scores for the subject population (on 10 of the 14 instruments where instrument standards are available) was greater on eight instruments indicating a broader range of scores than what typically exists in the general population.

5.2 *Factor Analysis*

A factor analysis was conducted on the 14 psychometric instruments to identify underlying trends in the subject population. Four factors emerged which in total describe 58% of the variance in the sample (as shown in Fig. 6). The first factor, which has the working title "openness to innovation" is compromised of HBDI-D,

Factor	Working Title	Primary Correlates	% Variance	Cumulative Variance
1	**Openness to Innovation**	HBDI-D B5-O KAI	24.92%	24.92%
2	**Rationality**	HBDI-A B5-N	13.05%	37.97%
3	**Planning & Organization**	B5-C W-ET HBDI-B	11.13%	49.10%
4	**Teaming Skills**	B5-A W-EF	9.29%	58.39%
5	**Factor X**	B5-E	8.67%	67.06%
6	**Factor Y**	W-EN	8.00%	75.06%

Fig. 6 Factor analysis of psychometric instruments. Component 5 and 6 were not included in the subsequent analysis because they are single instrument components with an Eigenvalue <1.0

BF-O and KAI and explains 24.92% of the variance in the sample. We found this promising, as it seemed a logical factor to track for its impact on design outcomes.

The second factor was given the working title of "rationality" because it is comprised of HBDI-A and B5-N (which is a positive measure of emotional stability). The "rationality" factor explains 13.05% of the variance in the sample. As this characteristic is common in the engineering discipline that dominates our subject population, it seemed a promising variable to track for the impact of standard engineering thinking on design outcomes.

The third and fourth factors were given the working titles of "planning and organization" (B5-C, W-ET, HBDI-B) and "teaming skills" (B5-A, W-EF) and explain 11.12% and 9.29% of sample variance, respectively. Again, these variables seemed promising, given that the projects at hand require extensive planning, organization and attention to team dynamics; altogether, these four factors seemed to cut a reasonable swatch across cognitive variables affecting design outcome.

5.3 Team Psychometric Diversity

Thirteen of the fifteen teams were distributed between students at two different universities in different countries. The remaining two teams had no international partners. In most instances, students were placed on teams by teaching team members. In some cases, students were allowed to self-select teams. This self-organization occurred before projects were assigned, so students tended to gravitate toward friends or acquaintances to form teams, rather then using any psychometric or problem-solving preference to form groups.

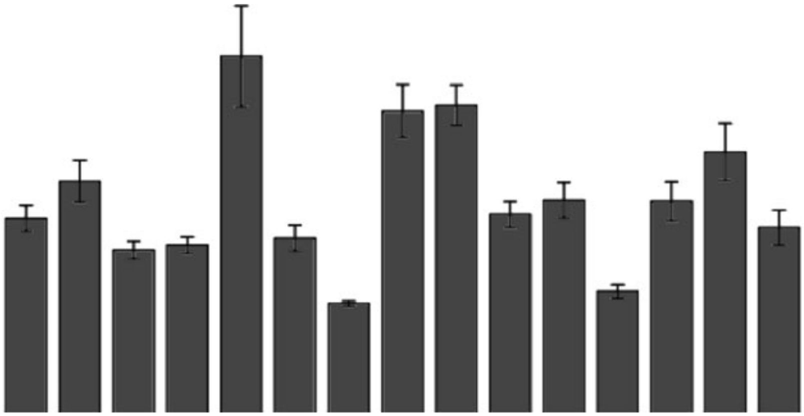


Fig. 7 Team diversity as measured by Cronbach's D, summed across all psychometric variables. N = 15, Error bars = 1 SEM

The resulting team make-up showed a surprising range of diversity for this study. For example, team size ranged from a low of three to a high of nine, with most teams [9] having seven members. The percentage of males on a team ranged from a low of 44% to a high of 100%. California/US had the single-largest body of students (32 of 97); consequently, the majority of the projects (10 of 15) had a Californian component.

Team cognitive style diversity was measured using Cronbach's D, for each psychometric measure and summing across all measures. The results are shown in Fig. 7.

This analysis revealed a surprising diversity in cognitive style and problem solving preferences across the 15 teams. In particular, the four highly diverse teams where significantly different that the four least diverse teams, singling out "outlier" groups for future analysis.

Based on the wider spread of scores in the psychometric instruments and the resulting differences in team scores on cognitive styles we concluded that there is support for Hypothesis 1 (H1): the ME310 student population does show more cognitive diversity than age, gender or cultural diversity.

5.4 Team Performance and Team Size

Team size tended to positively correlate with team performance (.42, p = .12). In addition, the percentage of male members on a team tended to negatively correlate with team performance (.45, p = .09), although many of the smaller teams were predominately male, so this effect may be redundant. As we will show later, males and female team members tended to have different psychometric profiles, with female team members possessing more of the characteristics that contributed positively to team performance.

The sheer quantity of work on these projects may have contributed to the fact that larger teams tended to perform better. Though they may have been frustrated with the greater amount of effort spent on group management, individuals on larger teams were able to effectively share the workload between more people. Unfortunately, due to the character of our sample we cannot draw any conclusions about what the "ideal" team size might be, if such a thing exists in this context.

5.5 *Team Performance and Cognitive Style Measures*

Overall, there seemed to be little correlation between team performance and the overall level of a cognitive style – with one notable exception. For example, team performance showed little correlation with team level cognitive style measures like KAI (−.02, p = .95), B5 extroversion (.14, p = .63), HBDI-B planning (−.08, p = .78) or HBDI-D innovation (−.09, p = .76), as shown in Fig. 8.

The notable exception was the W-TTI cognitive style measure of W-EF or "extraverted feeling." The higher a team level on this cognitive style characteristic the better team performance (.74, p = .00). In the factor analysis of the subject population, this indicator paired with B5-Agreeableness to form the "teaming skills" factor, which in total explained only 9.29% of the subject sample variance. In this analysis, B5-A did not correlate with team performance at the same level as W-EF. It is important to note the results of HBDI-C (.42, p = .11), a similar measure to W-EF, showed moderate positive correlation with team performance.

W-EF and HBDI-C were not overrepresented characteristics in our subject sample. W-EF was a W-TTI preference for 25% of the sample population, while average of HBDI-C scores (.405) were equal to the HBDI instrument norms (.418). Furthermore, there is evidence that a diversity of scores in these modes has no effect on team performance as discussed later; it is better to have a higher overall score than a spread.

Instrument	*R*	*p<*	*Instrument*	*R*	*p<*
KAI	-.02	*.95*	**HBDI-A**	-.31	*.25*
			HBDI-B	-.08	*.78*
B5-N	-.37	*.18*	**HBDI-C**	.43	*.11*
B5-E	.14	*.63*	**HBDI-D**	-.09	*.76*
B5-O	-.39	*.15*			
B5-A	.13	*.64*	**W-EN**	-.03	*.99*
B5-C	-.11	*.69*	**W-ES**	.24	*.38*
			W-ET	-.04	*.90*
			W-EF	.74	*.00*

Fig. 8 Correlations of team performance and psychometric instruments

In terms of construct validity, W-EF and HBDI-C have characteristics in common and may provide insight into why teams with higher levels of these cognitive styles performed better. Wilde's description of the W-EF (extraverted feeling) cognitive style is almost a personality indicator, someone who is an "expressive, tactful builder of group morale." Herrmann describes HBDI-C as "when the mood of an individual or group changes, [people with strong HBDI-C] are immediately aware of the change and [are] ready to respond to it, usually in a soothing or conciliatory way." p82 [20].

Extraverted feeling is typically associated with "social sensitivity" and a bias for pro-social behavior. Recent research on groups has yielded results that are qualitatively similar, suggesting that social sensitivity is a greater measure of team effectiveness than collective intelligence [21]. These behaviors can lead to a more considerate and positive atmosphere of communication within a team, as well as more coherence to group social norms and swifter conflict resolution. From this perspective, it is clear how W-EF and HBDI-C thinking could positively impact team performance.

5.6 Team Performance and Subject Factor Analysis

While most of the individual cognitive style measures did not show a strong relationship to team performance, the factors defining the subject population revealed another interesting insight into team performance. The first factor (Openness to Innovation; $r = -.37, p = .18$) and third factor (Planning and Organization; $r = -.31, p = .26$) showed no correlation to team performance.

However, the second factor (Rationality, $r = -.49, p = .07$) and the fourth factor (Teaming Skills; $r = .47, p = .08$) were moderately correlated with team performance. The Teaming Skills factor showed a moderate positive correlation with team performance, which is consistent with our earlier finding around W-EF and HBDI-C measures. The Teaming Skills factor consists of the measures B5-Agreeableness and W-EF. This is further reinforcement that empathetic, people-oriented problem solving preferences positively affected team performance.

This Rationality factor was comprised of the cognitive style variables HBDI-A (factual/analysis) and B5-N (emotional stability) and reflects a preference for "analyzing, dissecting, figuring out, solving problems logically, and getting the facts." p79 [20]. Said another way, the more stable, the more rational cognitive style of a team, the worse the team performed.

This may be because calm, analytical thinking is ill-suited to the design task at hand. On its face, it suggests that rationality is in some way opposed to design thinking. Elements of the design process that encourage "wild ideas" or innovative thinking may specifically (though implicitly) avoid the rational approach to problem solving.

There is some evidence to suggest that a high diversity of Rationality scores on a team is a good thing, though this result is not statistically significant. This could mean that a small number of predominantly rational thinkers are good thing, but it should not be the dominant thinking style on the team. However, it would likely require much larger sample sizes to see this effect in action.

5.7 *Team Performance and Cognitive Style Diversity*

The core hypothesis of this research is that cognitive style diversity on a team would lead to better performance over the course of a longer-term, multi-faceted design project. We measured cognitive diversity using Cronbach's D (*D*) as a measure of an individual's difference on any one cognitive style from all other members of the team, and then summed this difference for the team. More cognitively diverse teams would have a higher *D* score.

Correlation analysis on team *D* scores showed little overall relationship to team performance, as shown in Fig. 9. For example, cognitive style team diversity on instruments KAI (−.04, p = .90), B5-Agreableness (.04, p = .90), HBDI-C (.27, p = .33) or W-TTI EN (−.05, p = .85) had little or no correlation with overall team performance.

However, two interesting correlations stand out. First, the strong correlation between team levels of W-EF/HBDI-C and team performance is not apparent when W-EF (.03, p = .90) and HBDI-C (.27, p = .33) are measured by team variance rather than mean. This indicates that these characteristics may be valuable to team performance in the absolute and that variation or diversity in this cognitive style is not helpful to team performance.

Second, there is a modest positive correlation between variance of HBDI-A and team performance (.50, p = .06) suggests that diversity on this particularly cognitive skill may improve team performance. This might mean that a team with a range of skills ranging from analytical to intuitive might lead to more robust problem solving that a team comprised exclusively of analytical thinkers.

Instrument	*R*	*p<*	*Instrument*	*R*	*p<*
KAIv	-.04	*.90*	**HBDI-Av**	.50	*..06*
			HBDI-Bv	.35	*.20*
B5-Nv	.15	*.60*	**HBDI-Cv**	.27	*.33*
B5-Ev	.13	*.64*	**HBDI-Dv**	.33	*.23*
B5-Ov	.08	*.77*			
B5-Av	.04	*.90*	**W-ENv**	-.05	*.85*
B5-Cv	.23	*.41*	**W-ESv**	-.21	*.46*
			W-ETv	.34	*.22*
			W-EFv	.03	*.91*

Fig. 9 Correlations of team performance and the Cronbach D (variation) measure of psychometric instruments

There is more work to be done with this data set as a way to understand the impact of cognitive style diversity and team performance. We hope to reconstruct the team level data based on multi-variant individual difference, controlling for variables like project complexity, location and team size to develop a more detailed evaluation of the results. However, based on the results as presented we must conditionally reject our hypothesis (H2) that greater cognitive diversity on design project teams will lead to better project performance.

5.8 *Team Performance and Team Satisfaction*

Team performance tended to correlate with positive team behaviors, as shown in Fig. 10. Satisfaction questions were drawn from previous work on individual team member "psychological safety" within a working team environment [24].

Team performance tended to improve when teams reported a higher level of goal sharing (.50, p = .08), higher levels of teamwork (.47, p = .11) and better team skills at resolving conflict (.45, p = .12). This is consistent with Hackman's team performance research that shows a common direction, enabling structure and supportive context are important elements of team effectiveness [25]. However there was no apparent team performance benefit to agreeing on the actual project work or ultimately satisfaction with the team itself.

Interestingly, while team size had a positive impact on team performance, it seemed to have little impact on team satisfaction. There was no strong positive or negative correlation between team size and virtually any of the team satisfaction measures.

This may indicate that team size has a dualistic effect; it can help with performance because there are more available resources but factors other than team size moderate team satisfaction. Larger teams (in the six to nine member range) tend to become much harder to organize, requiring more effort in communication and self-management. It can become more difficult to gain consensus, create a shared

	Team Performance		***Team Size***	
Team Member Feedback (N = 13)	**R**	***p<***	**R**	***p<***
Share similar goals	.50	*.08*	.13	*.67*
Good teamwork skills	.47	*.11*	.14	*.65*
Resolving team conflict	.45	*.12*	.12	*.70*
Agreement about the work	.08	*.79*	.05	*.87*
Discuss conflict together	.33	*.27*	.24	*.44*
Like the other members of your team	.38	*.20*	-.11	*.71*
Satisfied working on this team	.10	*.73*	.10	*.73*
Aron "Scale of Closeness"	.09	*.79*	-.08	*.82*

Fig. 10 Correlations of team performance and team size with satisfaction

understanding and align goals in larger teams. However, if team members possess the necessary behavioral skills to manage these challenges, then the inherent difficulties of team size seems to have little impact on team satisfaction.

We also looked at *D*, our measure of cognitive diversity, and it's relation to team satisfaction. Overall, there is no correlation between team-level *D* and any measure of team satisfaction. Team Satisfaction highly correlated across measures ($\alpha = .83$) and average Team Satisfaction showed no correlation (.17, $p = .55$) with team cognitive diversity as measured by total *D*.

Based on these results, we reject our hypothesis (H3) that more cognitive diversity on design project teams will lead to lower intra-team satisfaction. It appears that cognitive diversity, per se, has no impact on team satisfaction, probably because team satisfaction is negotiated over time and team members adjust (or not) to differences in cognitive style.

5.9 Team Performance and Leadership

Four out of the 15 project teams in our study satisfied the condition for emergent leadership (that team members had at least two-thirds agreement on who the leader was). Through an open form question about leadership style, we were able to determine that of the four teams with emergent leaders, two had a qualitatively positive response from teammates and two had a negative response.

Positive versus negative leadership did not seem to make an impact on performance, but on average the four teams with emergent leaders exhibited lower performance than those without. Though this result was not statistically significant, we believe it indicates that a single-leader team is not the optimal model for these types of projects. The majority of individuals on all teams did single out one team member as the leader, but they generally did not agree on whom that person was.

This is evidence of a shared sense of leadership in the team, where responsibilities and decision-making are perhaps more evenly balanced among members. This attitude may also foster a sense of shared ownership of the project that encourages engagement from all members and facilitates goal alignment. Our data suggest that this state of shared team leadership may be better suited for design work than the single-leader model of team organization.

6 Discussion: Limitation and Future Study

Certainly, the effect of cognitive diversity on team performance is complex and difficult to observe. Our investigation to date has found no strong evidence that cognitive diversity has a clear-cut impact (positive or negative), at least not according to our current model. There are several steps that we could take to further explore the meaning of cognitive diversity on teams, both theoretical and methodological. On the theoretical side, we may need to re-examine our construct of cognitive diversity and develop something more directly relevant to teamwork (such as focusing on communication). Thankfully, our data set is rich and thoroughly complete and should continue to offer further opportunities for analysis and exploration into team cognitive diversity.

Methodologically, we have several sources of error that could be corrected for in future experiments. Though our sample of cognitive style data was moderate in size ($n = 97$), all team-level analysis was conducted only at the $n = 15$ level and therefore makes it difficult to draw statistically sound conclusions. We would need to expand the study to include more teams to circumvent this limitation.

Additionally, as our data is based on observations in the field and all parameters were outside of our control, there is much noise and undesirable variation in the data set. One example of this is that the projects were not all equal; the 15 projects had different clients, different subject matter and varying degrees of difficulty. Though our difficulty assessment attempts to adjust for some of this variability, the performance data would be much more reliable had we had all teams take on an identical design challenge. Developing a standardized, repeatable design task that could be conducted and observed in a laboratory setting (lasting minutes or hours, not months) would allow us to have much more reliable data on team effectiveness and performance.

It proved to the distributed nature of the course and students' busy schedules, it was quite difficult to coordinate data collection at all nine universities. This caused our cognitive style data collection to be spread out over a protracted period of several months, when ideally we could have assessed the whole population simultaneously before real project work began. Other timing considerations in data collection include the administration of the satisfaction survey on the final day of project work during the students' exposition. This is a time of high-stress, little sleep and strong emotions for the students; it is likely that all of this will have an unpredictable influence on self-reported satisfaction in the short-term. Conducting satisfaction surveys both mid-stream and at the end of the projects could have helped us to stabilize this information.

How individual cognitive styles actually manifest themselves in teamwork is an open question for future study. For example, if a certain cognitive style preference is associated with analytical thinking, can we expect those individuals with this set of preferences to exhibit analytical reasoning or behavior in a team setting? What does this behavior look like, and how is it measurably different from the behaviors of individuals with other sets of preferences? We hope to evaluate these psychometric instruments in a laboratory setting to determine the specific impact of certain preferences and styles on teamwork and performance.

In fact, many of the exercises and dogmas within the design process and educational framework of the course already appear to directly encourage W-EF type behaviors. An emphasis on constructive feedback (rather than critique), on behalf of the students and the teaching team, tend to color the team conversation with a more "socially sensitive" attitude. This is specifically evidenced in exercises such as "I like, I wish" and "Yes, and…" Furthermore, students are encouraged to openly discuss disagreements and develop skills in conflict resolution.

In fact, much of the (explicit and implicit) contribution of teaching team members and coaches is not content-specific technical assistance, but rather team dynamics-focused mediation. A weekly social mixer was part of the course program at several of the universities involved, aimed specifically at encouraging social interactions across teams. The importance of this pro-social element is even reflected in one of the professors' remarks that he views his position as "chief cheerleader" for the students. In short, it appears that W-EF-type behaviors have already been recognized as making important and meaningful contributions to these teams' performance. Our cognitive assessment of the teams appears to support this notion.

Acknowledgements We would like to acknowledge the generous support of the Hasso Plattner Design Thinking Research Program for making this research possible. Also, we would like to earnestly thank the students who spent a good amount of their valuable time to complete our surveys, as well as the international teaching team for granting us access to their courses and students and facilitating data collection. We would also like to thank Rich Shavelson for his invaluable perspective and guidance, Pam Hinds for her extensive insight into groups and teams, and Jon Shemwell for assisting with statistical analysis.

References

1. Mannix E, Neale MA. What Differences Make a Difference? Psychological Science in the Public Interest. 2005;6(2):31.
2. Van Knippenberg D, Schippers MC. Work group diversity. Psychology. 2007;58.
3. Baer M, Oldham G, Jacobsohn G, Hollingshead A. The Personality Composition of Teams and Creativity: The Moderating Role of Team Creative Confidence. The Journal of Creative Behavior. 2008 Dec 1;42(4):255-282.
4. Woolley AW, Gerbasi ME, Chabris CF, Kosslyn SM, Hackman JR. Bringing in the experts: How team composition and work strategy jointly shape analytic effectiveness. Small Group Research. 2008;39(3):352–371.
5. Woolley AW, Chabris CF, Pentland A, Hashmi N, Malone TW. Evidence for a Collective Intelligence Factor in the Performance of Human Groups. Science. 2010 Oct 1;:6.
6. Messick S. The nature of cognitive styles: Problems and promise in educational practice. Educational Psychologist. 1984;19(2):59–74.
7. Kozhevnikov M. Cognitive styles in the context of modern psychology: Toward an integrated framework of cognitive style. Psychological Bulletin. 2007;133(3):464.
8. Cronin MA, Weingart LR. Representational gaps, information processing, and conflict in functionally diverse teams. Academy of Management Review. 2007;32(3):761-773.
9. Cronbach LJ, Gleser GC. Assessing similarity between profiles. Psychological Bulletin. 1953;50(6):456–473.
10. Kirton MJ. Adaption-innovation: In the context of diversity and change. Psychology Press; 2003.
11. Burke RJ, Weir T. Coping with the stress of managerial occupations. In: Current Concerns in Organisational Stress. London: Wiley; 1980.
12. Kochan T, Bezrukova K, Ely R, Jackson S, Joshi A, Jehn K, et al. The effects of diversity on business performance: Report of the diversity research network. Human resource management. 2003;42(1):3–21.
13. Blau PM. Inequality and heterogeneity: A primitive theory of social structure. Free Press New York; 1977.

14. Wilde DJ. Teamology: The Construction and Organization of Effective Teams. Springer Verlag; 2008.
15. Pelled LH, Eisenhardt KM, Xin KR. Exploring the black box: An analysis of work group diversity, conflict, and performance. Administrative science quarterly. 1999;44(1):1–3.
16. Coffield F. Learning styles and pedagogy in post-16 learning a systematic and critical review. London :: Learning and Skills Research Centre,; 2004.
17. Kirton MJ. Adaptors and innovators: A description and measure. Journal of Applied Psychology. 1976 Oct;61(5):622-629.
18. McCrae RR, John OP. An introduction to the five-factor model and its applications. Journal of personality. 1992;60(2):175–215.
19. Gosling SD, Rentfrow PJ, Swann WB. A very brief measure of the Big-Five personality domains. Journal of Research in Personality. 2003;37(6):504–528.
20. Herrmann N. The Creative Brain. Revised. Lake Lure, NC: Ned Herrmann Group, The; 1989.
21. Herrmann N. The whole brain business book. McGraw-Hill Professional; 1996.
22. Bunderson CV. The validity of the Herrmann Brain Dominance Instrument. Creative brain. Lake Lure, NC: Brain Books. 1989;
23. Wilde D. Personalities Into Teams. Mechanical Engineering: The Magazine of ASME [Internet]. 2010 Feb;Available from: http://memagazine.asme.org/Articles/2010/February/Personalities_Into_Teams.cfm
24. Edmondson A. Psychological safety and learning behavior in work teams. Administrative science quarterly. 1999;44(2):350–383.
25. Hackman JR. Leading teams: Setting the stage for great performances. Harvard Business Press; 2002.

Monitoring Design Thinking Through In-Situ Interventions

Micah Lande, Neeraj Sonalkar, Malte Jung, Christopher Han, and Shilajeet Banerjee*

Abstract Building on existing knowledge of design and design thinking we apply several other fields of knowledge such as emotion coding, improvisation, ethnography, social psychology, and decision analysis into key metrics we call Design Thinking Metrics (DTM). We applied these metrics to analyze and assess videos of software design teams. We then conducted a workshop series with a professional software design team to use DTM as a *perceptual* tool to test a number of *action-repertoires* and building theory that could be used to improve Design Thinking practice. The result is multi-disciplinary perceptual monitoring of design thinking activity in professional software practice.

M. Lande
Department of Engineering, Arizona State University, 7231 E. Sonoran Arroyo Mall, Mesa, AZ 85212, USA
e-mail: Micah.Lande@asu.edu

N. Sonalkar • M. Jung
Center for Design Research, Stanford University, 424 Panama Mall, Building 560, Stanford, CA 94305, USA
e-mail: sonalkar@stanford.edu; mjung@stanford.edu

C. Han
Department of Management Science and Engineering, Stanford University, Huang Engineering Center, 475 Via Ortega, Stanford, CA 94305–4121, USA
e-mail: Christopher.han@stanford.edu

S. Banerjee (✉)
Department of Mechanical Engineering, Stanford University, Building 550, Stanford, CA 94305–4021, USA
e-mail: banny@stanford.edu

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_12,

1 Introduction

Design thinking (Dym et al. 2005) is the way designers approach messy situations (Rittel and Webber 1973) in order to create radical innovations. With the messiness of the situation and designers' intention to realize novel ideas, design thinking is a unique practice without rigid procedures. How do we understand such a complex dynamic activity? Furthermore how do we engage with practitioners in improving this complex dynamic activity?

Researchers trying to understand design thinking are much like the blind men trying to make sense of an elephant in a popular parable. We are blind on two fronts: first is the limit of our disciplinary lens used to study design thinking, and the second is an insistence on *observing* the design activity, which eliminates the researcher from actively engaging with the phenomenon to be studied. In this chapter, we present an attempt to overcome these two limitations through the combined used of a concurrent multi-disciplinary approach, and in-situ interventions into the practice of design thinking in software design teams.

We chose software design as a domain for our inquiry. While traditional product development teams have experienced success at adopting design-thinking methods, software development teams have encountered difficulty. Investigating the reasons for this phenomenon and developing effective interventions could potentially propel radical innovation in software. This could have far-reaching impact on society given how software, in particular enterprise software, has transformed the way people work and has contributed immensely to economic progress.

2 Research Questions

The research focus for this study evolved out of authors' desire to take their academic perspectives from the laboratory into the field. The questions that initiated the study were generative design questions (Eris 2002) intended to build new observations and hypotheses.

The research team focused efforts from their multiple perspectives. Spurred by the nature of doctoral work of the researchers that emphasized direct empirical observation and a software design focus by the nature of this research program, the research team converged on the following guiding questions:

Research Question 1: What can we learn about Design Thinking by having a team of researchers with multiple perspectives observe practitioners in action and conduct real-time interventions in professional design teams?

Research Question 2: What form does Design Thinking take in professional software design practice?

To develop an empirical understanding of design thinking, the researchers combined four different perspectives on design. Design Conceptualization, Co-creation, Decision Analysis and Affect are the set of Design Thinking Metrics (DTM).

3 Research Method

As researchers, we tackled two cases of experimental observation and analysis. *Case 1 Laboratory Analysis of Software Design Practice* consisted of video interaction analysis of three professional software design teams. This case followed a methodological approach rooted in video ethnography (Jordan and Henderson 1995) and in sequential analysis of video data (Bakeman and Gottman 1997). *Case 2 Field Probe of Software Design Practice* consisted of four workshops that were facilitated by the research team for a software design team in a Fortune 100 software company. This case followed a methodological approach rooted in action research (Lewin 1946) and participatory action research (Whyte 1989).

Both cases were informed by the doctoral work of the four researchers with regards to design learning (Lande and Leifer 2010), co-creation (Sonalkar 2010), decision analysis (Han 2010) and emotions (Jung and Mabogunje 2008). This created a unified underlying frame for the two cases and led to a synthesis of findings across the laboratory and the field of practice.

4 Case 1. Laboratory Analysis of Software Design Practice

4.1 Context of Study

The research team participated in a National Science Foundation sponsored workshop on “Studying Professional Software Design” in February 2010 at UC-Irvine (Petre et al. 2010). Three pairs of software designers were recorded by the workshop organizers. A set of DVDs with these videos and transcriptions was distributed in Fall 2009 to research groups around the world in a manner similar to the Design Thinking Research Symposia (Cross et al. 1992) (McDonnell and Lloyd 2009). The research team participated and presented at the Studying Professional Software Design workshop.

4.2 Study Design

Three software teams – all professional software designers at large software companies – participated in the study. The three participating design teams were from Adobe, Amberpoint and Intuit. These names will be used to refer to the two-person design teams.

The pairs worked at a white board to address a design prompt for a "Traffic Signal Simulator." Participants were given an identical three-page design brief with the problem description, requirements, desired outcomes and timeline. The teams were given 2 h to develop the user interaction and basic code structure for the imagined software application. Of the three video segments, Intuit lasts for approximately 1 h and the other two go on for almost 2 h.

4.3 Data Analysis

The research team first power browsed (Sonalkar et al. 2007) the first half of the video sessions to get a feel for the presentation and subject matter covered. We piloted our individual qualitative coding schemes and set up a dedicated space in the Design Observatory (Carrizossa et al. 2002), see Fig. 1.

The research team watched the complete set of three video sessions on consecutive days to capture observations and assessments within the real-time watching of the design activities. The concept of real-time analysis of design activity is part of the design instrumentation framework that motivates this research. The intention was to identify in real-time, the process metrics that can be predictive of design outcome in order to be able to provide appropriate feedback to the design team as to positively influence design activity.

The researchers attempted a version of real-time coding which meant that once we started playing the video, we did not stop or replay it. We categorized behavior as it played out on the videotape at a normal playback speed. The real-time coding activity was supplemented by a deeper dive into certain sections of the video that we identified as interesting. Taken together these analyses resulted in the identification of certain patterns of behavior that we describe in the next section.

The result of this analysis was a near real-time identification of events in team interactions using our four key metrics called Design Thinking Metrics. The metrics cover the following critical aspects of team interaction: divergence (generating ideas) and convergence (evaluating alternatives); managing disagreement (Jung

Fig. 1 Set-up in the design observatory

Table 1 Coding schemes

Design conceptualization

Design Thinking Activities:

Idea: new words or descriptors

Conceptual prototype: conceptual modifications, definitions

Experience prototype: scenario of use

Engineering Thinking Activities:

Functional prototype: implementation, math estimates

Co-creation

Product concept: verbal representations of possible product arrangements that occur in the present or future

Process concept: verbal representations of possible process arrangements that occur in the present or future

Conditional utterance of a concept: concepts uttered with language cues that denote a conditional possibility; *could, can, I guess, I imagine, I suppose, we probably, kind of.*

Forceful utterance of a concept: concepts uttered with language cues that denote a certainty; *need to, have to, have got to, should, must,* or use of present tense implying an already existing reality

Decision analysis

Instances of discussing the design basis

Frame, Preferences, Alternatives, State of information

Affect

Adapted from SPAFF Simplified Specific Affect for emotions (Coan and Cottman 2007)

SPAFF simplified	SPAFF original
High Negative	*contempt, belligerence, criticism, anger, defensiveness, disgust, domineering, threats, stonewalling*
Low Negative	*fear/tension*
Neutral	*neutral*
Low Positive	*validation*
High Positive	*affection, enthusiasm, humor, interest*

and Mabogunje 2008) that invariably arise in teams; design conceptualization assessing values and ways of thinking about design activity (Lande and Leifer 2010) and shared design values that persist in the group.

The above Table 1 indicates the coding scheme from each of the four perspectives.

4.4 Team Performance

Another aspect of data analysis was the evaluation of team performance. In order to contextualize our insight in terms of team performance, it was necessary to get an evaluative ranking of the three teams along meaningful criteria of team effectiveness. In terms of outcome of the software design activity, the three teams had representations on the whiteboard at the end of their session; though it was hard to identify a distinct deliverable that could be objectively judged. Nonetheless, we developed criteria relevant to the design task that could be used to rank the three teams. These are:

1. **Human-centeredness** as evidenced by the attention given to user interaction in their activity
2. **Modeling ability** as evidenced by the coverage of aspects of the traffic situation being discussed and the simplicity of abstraction.
3. **Documentation** as evidenced by the quality and quantity of whiteboard representation conveying the key points discussed during designing.

The first two criteria are directly derived from the desired outcomes mentioned in the design brief. In terms of the documentation, we attempted to isolate the Unified Modeling Language diagrams and rank them according to cue-based criteria. This did not enable us to develop a distinct team ranking. Hence we followed a more subjective route. As a team, we discussed our subjective impressions as to how the teams fared on these three criteria. None of us are domain experts in terms of software engineering, however it was interesting to note that we each arrived at similar ranking for the teams. Table 2 below lists the team rankings according to the three criteria.

4.4.1 Findings

Key findings are reported from each of the four perspectives that make up Design Thinking Metrics.

4.5 Design Conceptualization

When ratios of Design Thinking Activities to Engineering Thinking Activities, Table 3, are calculated the picture then becomes much clearer. The Adobe team is almost half that of the other two; they bounce back and forth much more often. The

Table 2 Team performance rankings by criteria

Company	Human-centered	Modeling	Documentation
Adobe	Medium	High	Medium
Amberpoint	High	Medium	High
Intuit	Low	Low	Low

Table 3 Ideation densities for teams per design thinking and engineering thinking activities; switches between activities

Company	Design thinking activities/min	Eng. thinking activities/min	DT:ET ratio	Switches gross/net
Adobe	2.8	0.5	5.3	58/36
Amberpoint	3.0	0.3	11.6	22/15
Intuit	3.7	0.4	9.0	28/16

actual number of switches between the Design Thinking and Engineering Thinking Activities is listed in Table 3. Again it shows the Adobe team moving more regularly and consistently among the Design and Engineering Thinking spaces, possibly addressing both the problem space and solution space more evenly and in tandem. We would then predict higher performance for the Adobe team based on this analysis.

4.6 Co-creation

Considering a ratio of conditional to forceful concepts, Adobe has the highest ratio of 1.38, followed by Amberpoint with a ratio of 0.86 and Intuit with a ratio of 0.47. Adobe has greater number of conditional concepts than forceful concepts while other two teams have the opposite characteristic with Intuit having half as many conditional concepts as forceful concepts.

So what can we infer from this analysis? One hypothesis is that conditionally expressed concepts are more conducive to co-creation as they enable the other participant to build-on and contribute to the concept. Forcefully expressed concepts on the other hand enforce a certain concept and do not leave room for improvisation. Prior work by Wilson (Wilson 2007) seems to point in the same direction. Wilson studied the learning and sense-making behaviors of teams as they engaged in extreme adventure racing. Wilson observed that in conditions of uncertainty, high performing teams made tentative claims that allowed their team members to negotiate and participate in the sense-making, while low performing teams made assertive claims that suppressed such participation (Fig. 2).

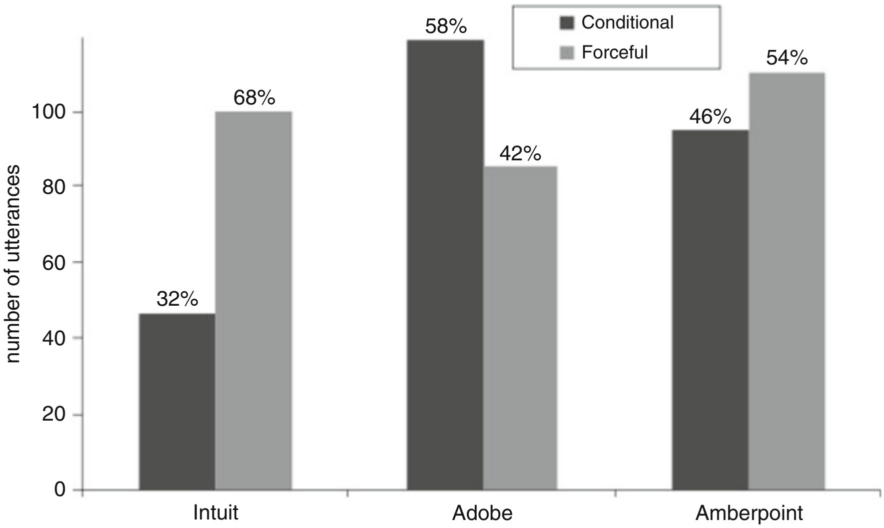


Fig. 2 Graph of conditional and forceful utterances in each team. The Y-axis denotes the number of utterances in each category

Based on this hypothesis, one could suggest that Adobe is a high performing team while Intuit is a low performing team.

4.7 Affect

The results of the simplified SPAFF (Coan and Cottman 2007) coding are summarized in Fig. 3. Each pie chart shows what percentage of the overall time a team displayed one of the five behavior types. Overall high negative behavior patterns such as contempt or belligerence and high positive behavior patterns such as excitement and humor were rarely displayed. The only team that showed some high negative behavior was the Amberpoint team, of which one team member showed domineering behavior during several moments. The high positive behavior patters consisted mostly of display of interest. Other high positive behavior patters such as humor and excitement were rarely observable across all three teams. The differences between the three teams from a specific affect perspective are mainly based on differences in the expression of tension (low negative) and validation (low positive). Ranking the three teams by the amount of positive affect expressed Adobe leads with 58%, followed by Intuit (45%) and Amberpoint (43%). Ranking the teams by the amount of negative affect expressed, Intuit leads with 25%, followed by Amberpoint (10%) and Adobe (7%).

The results of the Motivating Engagement Behavior coding are summarized in Fig. 4. Each line displays the cumulative sum of positive/motivating (+1 to +7) and negative/demotivating (−1 to −7) behaviors over time for each team. A positive slope means that there are more positive than negative behaviors per time-unit, which indicates an engaging interaction style. A negative slope means that there are more negative than positive behaviors per time-unit, which indicates a disengaging interaction style. A positive slope for a certain segment indicates that positive or

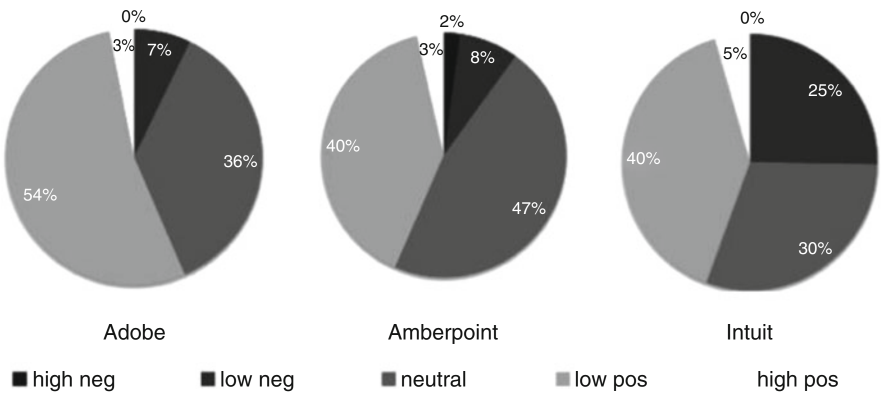


Fig. 3 Specific affect coding

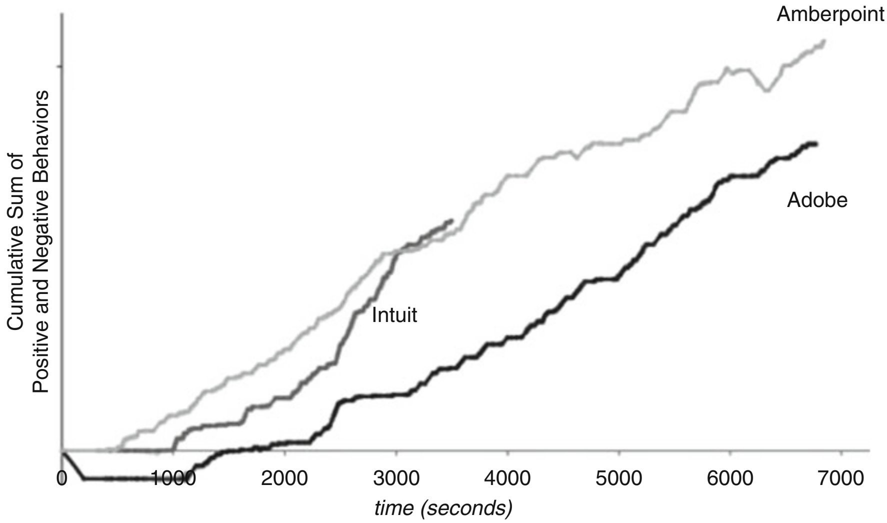


Fig. 4 Summarized results of motivating engagement behavior coding for each of the three teams

motivating behavior dominates during that segment. A negative slope for a segment indicates that negative or de-motivating behavior dominates during that segment. All three teams show slightly positive slopes, which means that in each team there is more positive behavior displayed over time than negative behavior. All three teams are similar in terms of this measure.

4.8 Rankings

Overall, we found a convergence in the four perspectives that we used to analyze the data. The design conceptualization perspective found that Adobe transitioned more often and more evenly over time between Engineering Thinking and Design Thinking. The co-creation perspective also found that Adobe had a greater ratio of conditional to forcefully expressed concepts. This leads us to suggest that conditionality in expression could influence transitions between Engineering Thinking and Design Thinking or vice versa.

Similarly there was convergence in the affective perspective and co-creation perspective as well. Intuit had the lowest conditional to forceful concept expression ratio. Intuit also had the lowest positive to negative affect ratio. It is conceivable that a negative affective climate has an influence on how a team perceives uncertainty. Also a forcefully expressed concept could communicate dominance and lead to a negative affect in a team.

The design basis could also be influenced by affect and conditional language. Understanding and developing a design basis occurs through a conversation between team members. If negative affect is expressed in a conversation, it could have a detrimental effect on the ability of a team to probe the design basis. Similarly the use of forceful language prevents exploration of alternatives in a conversation thus negative influencing the design basis.

Comparing the four perspectives with the team effectiveness ranking, we found similarities in the way each perspective ranked the teams and the way the teams were ranked through the team effectiveness criteria. Based on the similarities observed, we suggest the following hypotheses:

H1: The rate of transitions between Engineering Thinking and Design Thinking activities is directly correlated to team effectiveness.
H2: The failure to address each aspect of design basis is inversely correlated to team effectiveness.
H3: The ratio of conditionally to forcefully expressed concepts is directly correlated to team effectiveness.
H4: The ratio of positive to negative affect expressed in an interaction is directly correlated to team effectiveness.

5 Case 2. Field Probe of Software Design Practice

The first case study enabled us to probe design activity in real time and test if we could perceive elements of design thinking in action. We used four different dimensions – design conceptualization, co-creation, decision analysis, and affect – to guide our perception of design activity. The experience resulted in a sharpening of our own perceptual fields and noticing capabilities, as well as, in a set of testable hypothesis. The next question before us was – could we take these perceptual capabilities, and translate them into meaningful improvement of design activity in a professional setting? Could we create a model that would allow practitioners to do the same? The second case study presents our attempts to answer these questions.

5.1 Context of Study

This study was conducted in collaboration with a large enterprise software company. We collaborated with a software design team that was tasked with developed a novel concept and generating UI designs for it. Since the team was involved in early conceptual design, their activities gave us a meaningful context to engage in with our DTM framework and develop a model of perceptions influence design action.

5.2 Theoretical Approach

The theoretical approach we took builds on Schön's elaboration of knowing-in-action and reflection-in-action (Schön 1983). Schön proposed that the professionals don't just apply technical knowledge to situations of practice which are generally messy and ambiguous, but rather their knowing is situated in and linked to their doing of professional activities – their knowing-in-action and reflecting-in-action. Schön further elaborates on the 'seeing-as', framing, moving and reflecting that happen in "action-present" and result in an interaction – a give and take, or a conversation with the evolving situation of practice. Ingold (Ingold 2001) proposes a similar emphasis on "practices of skills" rather than "products of intelligence". Ingold considers the coordination of perception and action at the core of a skill. He argues that we need to "shift our analytic focus from problem-solving, conceived as a purely cognitive operation distinct from the practical implementation of the solutions reached, to the dynamics of practitioner's engagement, in perception and action, with their environments."

Deriving from Schön's and Ingold's work, we identified three key elements of such conversation with the situation that forms the core of a professional's practice (Jung et al. 2010):

1. Perception, and perceptual-field
2. Action, and action-repertoire
3. Theory – implicit and explicit

Perception, and perceptual field – Perception is the activity of sensing and naming a certain phenomenon. Perceptual field is a collective of such perceptions that have been acquired to be meaningful to a certain context of practice. It can be defined as sensing organized around a purposeful activity.

Action, and action-repertoire – Action is the activity of moving in a purposeful way. An action-repertoire is collection of such actions that have been acquired in the context of practice.

Theory – implicit and explicit – An implicit theory refers to the heuristics, beliefs and expectations associated with particular perceptions and actions in the context of practice. An explicit theory on the other hand is a codified understanding of phenomenon that has been learned in the technical context of practice.

The following diagram, Fig. 5, refer to the relationship between perception-action occurring around an implicit theory in practice. Explicit theory can be derived from this practicing.

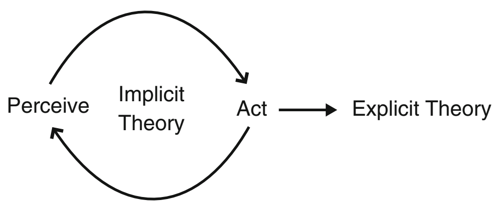


Fig. 5 Perception action theory loop with implicit and explicit theory

5.3 Study Design

The study was designed as a series of four workshop interventions with the team of software designers. These interventions were a collaboration between the team of researchers and the team of software designers. The software designers brought in content relevant to their design task. The researchers brought in a set of activities to the structure and probe engagement with the design content. These activities centered on the content brought in by the design team, gave us an opportunity to probe the perception-action and implicit theories that both the designers and the researchers held about design thinking. We as researchers followed the model of perceiving – assessing the need for intervention – acting out an intervention – and then assessing the effect. This at one level, enabled the designers to become more aware of their own implicit theories and modify their perceptual fields and action-repertoires, and another level enabled the researchers to test out the model of perception-action in a professional setting. The following diagram, Fig. 6, visualizes the engagement with the design team.

5.4 Data Analysis

Data was collected during the sessions in the form of photographs and notes, and after the session in audio recordings and notes of the research team's post-debriefing.

The following table gives an overview of the four workshops (Table 4).

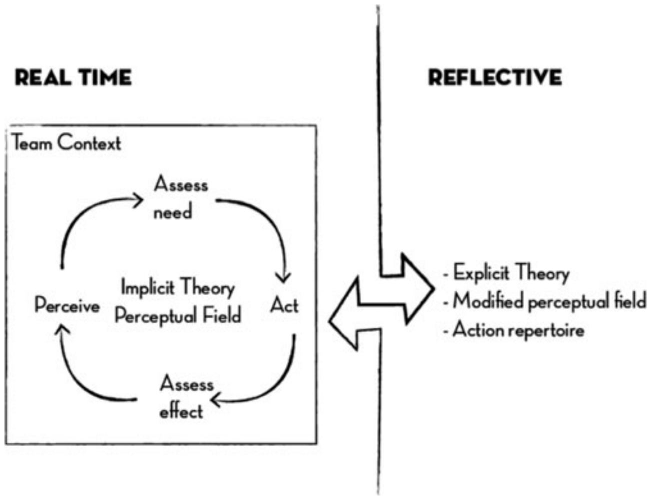
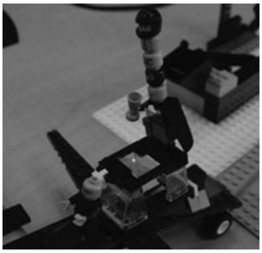
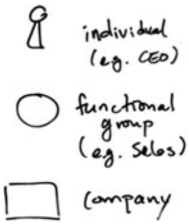
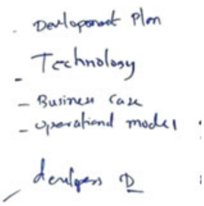


Fig. 6 Modeling engagement with the design team

Table 4 Workshop summaries

	The first workshop was a visioning workshop that involved the use of Legos as tangible media for illustrating design concepts and scenarios for future
	The second workshop was focused on stakeholder analysis for the design project. The participants sketched out their understanding of the project space
	The third workshop was a continuation of the second. The participants developed the point of view in terms of stakeholders of the project
	The fourth workshop involved further planning of their design activity

The researchers facilitating a particular session met immediately afterwards and discussed their perception-action interventions and the theory behind their interventions. These debriefing researchers discussions were audio recorded after the session.

Data analysis consisted of consolidating and analyzing the perception-actions pairs along with their implicit theory. The following table is sample of the analysis conducted (Table 5).

Table 5 Perception, theory and action effect examples

Example 1	
Perception	Tense behavior from SB communicated orally as a wish or non-verbally – fidgeting, back and forth focus on a particular thing, shifting around
Theory	I wanted participants to be at ease when doing the Lego activity. It was my responsibility as the facilitator to put them at ease and if I noticed unease, I wanted to act in a way so as to remove it
Action	I suggested that SB get his phone left in the car and which may have causing him distress
Effect	SB uttered his thanks and went off to get his phone
Example 2	
Perception	Awareness of time it took to complete different activities
Theory	My goal was to be able to complete the activities we had planned for the 2 h
Action	Either a self-regulation of my own expectation or an external intervention to speed things up
Effect	I become less anxious with an internal intervention and allowed the activity to go on in a different direction. With an external intervention the activity was kept on an expected path
Example 3	
Perception	Verbal expression to hold on to the model
Theory	The team would own the model and it would be more likely to enter their practice is it has persistence in space
Action	Giving the team permission to keep the Lego blocks for a week
Effect	The team was verbally enthusiastic about the model and about sharing it

The perception-theory-action-effect pairs occurred at different levels of in-the-moment analysis. Most of them were focused on the design team and their behaviors. However some of them as noted in the table above were also focused on perception of researcher's own emotional and cognitive state.

5.5 Outcome

The workshops were acknowledged to be successful by the design team in enabling them to engage with their design project in a different way. The Lego visioning workshop was especially well received with the team inviting the researchers to do a similar activity in a session with their clients.

The key outcome from the research perspective was more methodological in nature. We implemented a perception-action approach to intervening in a design situation and at the same time building a set of theories tied to the perceptual field and action-repertoire meaningful in the context of practice. The results were encouraging. We were able to adapt a coding scheme – the Design Thinking Metrics, draw out the perceptual units of design thinking behavior and intervene successfully by developing an action-repertoire in a context of professional practice.

6 Conclusions

This research study opens up a discussion for developing a new epistemology of design research. One that is based not on a methodology derived from natural sciences or social science, but rather from a methodology of design practice that emphasizes linking perception-action of a human with the theories derived and implemented in a given situation.

Acknowledgements The research team would like to thank the organizers of Studying Professional Software Designers workshop as well as the participants in both case studies examined.

References

Bakeman, R, & Gottman, J (1997) Observing interaction: An introduction to sequential analysis. Cambridge Univ Press.
Carrizossa K, Eris O, Mabogunje A, Milne A and Leifer L (2002) Building the design observatory: a core instrument for design research. Design 2002 Conference.
Coan J and Cottman J (2007) The Specific Affect Coding System (SPAFF). In: Handbook of emotion elicitation and assessment.
Cross, N, Dorst, K and Roozenberg, N, eds. (1992) Research in design thinking. Delft University Press, Delft, The Netherlands.
Dym C, Sheppard S, Agogino A, Leifer L, Frey D, and Eris O (2005) Engineering Design Thinking, Teaching, and Learning. Journal of Engineering Education.
Eris O (2002) Perceiving, Comprehending, and Measuring Design Activity through the Questions Asked while Designing. Dissertation, Stanford University.
Han C (2010) Design basis and other useful distinctions for creativity and cognition. ACM Creativity & Cognition.
Ingold, T (2001) Beyond art and technology: the anthropology of skill. Anthropological perspectives on technology, 17–31.
Jordan, B and Henderson, A (1995) Interaction analysis: foundations and practice. The Journal of the Learning Sciences, 4(1), 39–103.
Jung M and Mabogunje A (2008) Emotion-dynamics in multi-disciplinary design teams. Symposium on cognitive, social and emotional processes in creative teams. International Conference of Psychology.
Jung, M, Sonalkar, N, Mabogunje, A, Banerjee, B, Lande, M, Han, C and Leifer, L (2010) Designing perception-action theories. DTRS8 Design Thinking Research Symposium.
Lande M & Leifer L (2010) Incubating engineers and hatching design thinkers: mechanical engineering students learning design with ambidextrous ways of thinking. American Society for Engineering Education Conference.
Lewin, K (1946) Action research and minority problems. Journal of Social Issues, 2(4), 34–46.
McDonnell J and Lloyd P, eds. (2009) About design: analysing design meetings. Taylor & Francis, London.
Petre, M, van der Hoek, A and Baker, A (2010) Editorial. Design Studies. Volume 31, Issue 6, Pages 533–662.
Rittel H and Webber M (1973) Dilemmas in a General Theory of Planning. Policy Sciences. Vol 4, pp. 55–69.
Schön, D (1983) The reflective practitioner: how professionals think in action: Basic Books.

Sonalkar N, Mabogunje A, Leifer L, Eris O and Jung M (2007) Powerbrowsing: a method to accelerate designers' familiarity with video information in digital libraries. Proceedings of the International Conference on Engineering Design.
Sonalkar N (2010) Co-creation for engineering design. ACM Creativity & Cognition.
Whyte, W (1989) Advancing scientific knowledge through participatory action research.
Wilson D (2007) Team learning in action: an analysis of the sensemaking behaviors in adventure racing teams as they perform in fatiguing and uncertain contexts. Dissertation, Harvard University.

Part IV
Design Thinking in Information Technology

On the Perception, Adoption and Implementation of Design Thinking in the IT Industry

Tilmann Lindberg, Eva Köppen, Ingo Rauth, and Christoph Meinel*

Abstract In recent times, addressing the social aspects of IT products has become an important competitive factor on IT markets. IT development is forced to focus on more user-centeredness and the non-technical aspects of design problems. Against this background, design thinking has been discussed and applied as a new design paradigm for IT development. Basing on expert interviews and case study research, we examine in our research project what it means to put design thinking into operation in an IT context. We explain why design thinking is complementary to traditional IT design paradigms and what issues are involved in the subjects of perceiving, implementing and adopting design thinking in IT development.

1 Introduction: Design Thinking as a Complementary Design Paradigm for the IT Industry?

IT development processes call for highly trained professionals who are qualified to deal with complex technical issues, as programming languages or software and hardware architecture. Competencies in software engineering are not only important for taking part in programming, but also in designing the software.

As every decision about the design of an IT-system unavoidably manifests at the level of architecture or code, expert knowledge and thinking already play an important role in early design decisions. Consequently, the educational background of hardware and software engineers has a strong influence on mind-set building and

T. Lindberg • E. Köppen • I. Rauth • C. Meinel (✉)
Hasso Plattner Institute, University of Potsdam, P.O. Box 900460, 14440 Potsdam, Germany
e-mail: tilmann.lindberg@hpi.uni-potsdam.de; eva.koeppen@hpi.uni-potsdam.de; meinel@hpi.uni-potsdam.de

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_13,

problem solving. As a result, IT development has the tendency to take place within an 'exclusive' experts' world (Lindberg and Meinel 2010).

In past and present times, these circumstances have led to the fact that technically and analytically trained IT engineers take on the designer's role as well, although they have not been professionally trained in that field. The word *software design* is indeed strongly associated with technical issues. This 'technical bias' in IT design leads to the tendency that the technical complexity of a design problem receives more attention than its social complexity. IT development has been struggling with the situation that products, functionalities and interfaces shape up as incomprehensible, inappropriate or simply unoriginal for the user, while other features or products considered as essential or meaningful from a user's point of view are not being addressed.

Overcoming this situation has become a key issue for the IT industry. As times went by in which the IT market grew mainly driven by technology push dynamics, the challenges IT development has faced exceed the established focus of an engineering experts' world and ask for the integration of further perspectives on problem framing and solution finding. Within the IT world, this problem has been tackled so far in two different ways. On the one hand, new design disciplines such as interaction or user experience design came up taking on specifically the role of the "user's advocate" within development teams (Buxton 2007; Mandel 1997; Vredenburg et al. 2002). On the other hand, new software engineering approaches, in particular those summarized under the umbrella term "agile development", put strong emphasis on an incremental and iterative development processes that is adaptive to user feedback throughout (Beck and Andres 2004; Pichler 2008).

The present debate on applying design thinking to the IT industry also addresses this problem, however with a primary focus on broadening generally cognitive problem solving patterns in IT development. Design thinking is associated with a problem solving style that supports the consideration of socially ambiguous aspects of a design problem. In contrast to orthodox engineering design paradigms, the corresponding problem solving patterns build upon heuristics and situational reasoning rather than on analytical thinking as focused in the IT engineering curricula. Design thinking differs from analytical thinking in diverse aspects:

- Design thinking relies on the development of concepts by using preliminary and even intuitive knowledge about a design problem, while proofs of concepts are adduced by the negotiation between different and probably conflicting stakeholder perspectives – i.e. the users, clients, manufacturers, law-makers (Dorst 2006; Lawson 2006; Owen 2006; Krippendorf 2006). Analytical thinking, in contrast, asks for definite knowledge to frame design problems into well-structured units before the actual problem solving process starts.
- In design thinking, problems are perceived as 'wicked' (Rittel 1972), saying that there is no definite formulation of a design problem at all. In this perspective, structuring a problem is rather seen as a process of taming instead of defining a problem. The relation between problem and solution therefore is not like deriving the latter from the former, as purely analytical approaches would

Table 1 Analytical thinking vs. design thinking

	Analytical thinking	Design thinking
	Ill-structured	Wicked
Problem perception	Well-structured	Tamed
Relation problem/ solution	Solution as a derivative consequence of a well-structured problem	Co-evolution of problem and solution
Key knowledge	Expert knowledge	Stakeholder knowledge
Key process	Defining and deriving	Iterations of observing and synthesizing, ideating and prototyping
Design paradigm	Rationalist problem solving	"Reflective conversation with the situation"

suggest, but like framing problems and solutions *interdependently* in frequent iterations. This is what Cross calls the "co-evolution of problem and solution" (Cross 2007). Analytical thinking, however, pursues the analytical investigation of a problem setting by decomposing all its components or its determining factors and uses this knowledge to compose a design concept as a logical consequence.

- The key knowledge in design thinking is not the expertise of specialists but the knowledge of stakeholders that is supposed to be learned anew for every design process. The process behind design thinking thus builds rather on learning about problem and solution than on applying already learned knowledge, and therefore supports all activities of grasping multiple knowledge and multiple perspectives for inspiration as well as the creative transformation into new concepts (Beckmann and Barry 2007; Brown 2008; Lindberg et al. 2010; Owen 2006).

Seen together, analytical thinking and design thinking suggest different paradigms of designing (see also Table 1). Analytical thinking advocates applying scientific-rationalist problem solving to design problems and is particularly prevalent in areas with a strong focus on technical rationality – such as in IT engineering. In design thinking, however, solving design problems is regarded as a "reflective conversation with the situation" (Schön 1983) or as "reflection-in-action" (Dorst and Dijkhuis 1995) – a discursive and creative activity of developing design solutions in frequent communication with the stakeholders of a design problem. Therefore, design thinking focuses on those 'fuzzy' aspects of a design problem, which are in purely engineering-led approaches left aside, and is thus suggested as a useful supplement to problem perception and solving in 'traditional' IT development approaches.

In consequence, problem solving approaches in design as well as in IT engineering complement each other in principle. Whereas design thinking allows dealing with the ambiguity of design problems as wicked problems, the thinking of IT engineers instead supports the effective technical realization. The inherent difficulties to communicate between experts and non-experts during the IT development process make the complementary use of both approaches complicated and lead to a dominance of analytic-systematic approaches to problem and solution finding. Against this background, design thinking can take the role of a meta-disciplinary rational, which allows a team across the disciplines to develop a mutual and general understanding of problem and solution, as it broadens disciplinary reasoning and helps, for example engineers, to forget about the patterns for a moment that they have internalized in their academic training – until a problem has been defined precisely enough so that professional rationales and expert knowledge may suitably be applied.

2 Research and Methodology: Analyzing Language Games on Design Thinking in the IT Industry

Beyond purely conceptual thought on applying design thinking to IT development, important questions remain open. How can design thinking be conceptualized and distinguished within software engineering on a practical level? How is it understood and adopted to daily work routines? How can it be imparted and organizationally implemented? Finding answers to those questions is still a challenging endeavor, as there is a lack of both conceptual models and hands-on experience. In our research project "Design Thinking in the Development Processes of the IT Industry" we set up initial research to explore those questions.

We conducted expert interviews mainly with IT developers that have been trained in design thinking workshops as well as with trainers and observers of those workshops. The basic insights gained from these interviews showed the topic's complexity quite clearly. Albeit the majority of interviewees regarded design thinking by some means as enriching, we discern partly different views on what design thinking is and how it can be adopted and implemented. We hypothesize that this variety of perspectives affects not only implementation and adoption, but shows also a paradoxical trait of design thinking itself, namely that it is neither perceived as an insubstantial 'buzz word', nor as a delimited concept. Between both extremes, its meaning seems to be strongly subjected to vivid 'language games'.

'Language games' is a concept developed by the philosopher Ludwig Wittgenstein (1984) that explains how one word can carry an infinite series of meanings – depending on the context or situation in which a word is used. Wittgenstein puts language in analogy to games, because each game represents a certain set of backgrounds, goals and rules. Some are constitutive; others are rather implicit and can be modified. Wittgenstein applies this as metaphor to language, stating that rather the language games than the words decide whether communications work: when people play with the same words but according to different rules, there would be a great juxtaposition of language games hindering each other to succeed. Thus we regard it as important to identify and distinguish 'language games' on design thinking in order to create a basic understanding about how to apply design thinking to the IT industry. To do so, we employ a systematic approach to qualitative text analysis based on 'grounded theory'. In what follows, we will give a short overview about our research setting and our method of investigation.

We conducted 30 expert interviews (Bogner et al. 2005) with three groups of people working in the IT industry in Germany and the US: first, design thinking experts that educate IT engineers in design thinking; second, IT engineers that have been trained in design thinking; and third, experts from specialized design disciplines like user experience design that observe these efforts.

All interviewees were involved with design thinking in the form of a particular didactic workshop model either as trainer, participant or observer. Those workshops followed an approach popularized by the design agency IDEO as well as the Stanford d.School model (Plattner et al. 2009), in which small, multidisciplinary teams (generally without a professional design background) tackle a seemingly simple design challenge (for instance: how to improve ticket machines for public transport) and are supposed to understand how far one's own imagination of a viable solution changes after learning about the stakeholder perspectives (in particular the users'). To do so, those workshops suggest a prescriptive process model guiding the team through a design workflow in which first the team learns about the problem, then synthesizes the gained information to a framework of knowledge, using this framework as inspiration for ideation, and develops, prototypes and refines those ideas iteratively by means of frequent user feedbacks.

We developed interview guidelines with slight variation between the first and the other groups. Those guidelines contain three groups of questions: first, questions about the interviewee and his department's role in the company; second, questions about his view on design thinking; and third, questions about his opinion on how to implement and adopt design thinking to IT companies. Each interview lasted approximately 1 h, was recorded and later on literally transcribed.

We used grounded theory as our methodological framework for data analysis (Strauss 1998). Grounded theory is an approach developed in social science for empirically substantiated theory generation and is particularly useful when it is required to frame a fuzzy empirical setting. Condensing empirical data in frequent iterations and comparisons in order to develop coding schemes is the main driver of theory generation. We pursued an 'axial coding' approach, as we presupposed 'language games' as core category for the data analysis process. We used the software MAXQDA to support the data analysis process (Kuckartz 2007). We synthesized our coding schemes to three hypotheses, which are depicted in the next chapter.

3 Results: On the Perception, Adoption and Implementation of Design Thinking in the IT Industry

We looked at three issues in our analysis: first, how is design thinking understood as such; second, how is it understood in respect of IT development; and third, how is it discussed within the scope of the implementation to organizational structures and the adoption to personal working routines. Dealing with these questions, we developed three hypotheses, which will be expounded below.

Hypothesis 1. The understanding of design thinking is more aligned when it comes to describing its general goals and principles; differences however increase when it comes to describing design thinking on a more applied level.

This hypothesis is related to the question of how design thinking is understood as such. We found out that there are no contradictory differences of opinion on what design thinking generally is, albeit the ways to express this vary. One interviewee stating that design thinking is *"willingness to ask (...) 'am I really solving the right problem'; and then to try out what the right ways are to solving this problem"*, stresses another aspect than an interviewee stating design thinking is "*that the usability of the product and acceptance of the end user determines the design of a product"*.[1] Another quote combines both messages: "*design thinking is a way to get out of your narrow view of what your problem is and (...) look broader and take everything from your environment in a view that kind of helps; (...) solve the problem that you need to solve, so most of the time it's going out and talking to users and talking to customers and anybody who is (...) associated with that problem.*" Generally spoken, the interviewees emphasize either one or both of the following aspects, namely (a) finding the viable solution to the fairly understood problem, and (b) both the viable solution and the fairly understood problem are delimited by the user's point of view. Both aspects are deeply complementary, so that we do not see any confusion of language games when it comes to a general explanation of design thinking.

However, when it comes to applied explanations of design thinking, we discerned two divergent views. On the one hand, design thinking is explained as a methodology with a strong focus on a prescriptive process model, supportive tools and an underlying team structure. This can be exemplified with one interviewee distinguishing three levels of information about design thinking: first, "*specific tools and techniques, which are things like how to run a brainstorming workshop or how to do user interviews or the very specific tangible activities and tools that you do*"; second, *"the group dynamic piece" of (...) teams working on problems (...) and how do you get them to (...) come up with new ideas*"; and third, "*the overarching categories*" (he uses 'categories' instead of 'phases' as he wants to avoid the image of sequential process). On the other hand, design thinking is seen rather as a mind-set from which people draw their actions without relying on instructions from a formalized method. One quote shows this transition quite clearly: *"On the one hand, (...) (design thinking) is a method that I associate mostly with the whole process and its phases; and on the other hand it is a sort of mindset. (...) And I think you don't have to go through the whole process when you have this attitude (...). You just should have the intention in mind and try to live it."* We see in both views fundamentally different qualities. The first view regards design thinking as a bundle of methods that can be realized by means of organizational arrangement; the second regards it as a way of thinking that has to be

[1] German quotes are translated to English by the authors.

internalized by means of education. Thus we assume that this causes a juxtaposition of language games that can make it difficult to agree on the concrete purpose of design thinking in a company: Is it a meta-disciplinary attitude that people should learn, or is it an organizational technique that people should stick to?

Hypothesis 2. Design thinking is understood as a learning approach contributing to IT development rather than a development approach in itself.

Our second hypothesis is connected with the question how far the understanding of design thinking is related to IT development. This is a central aspect as it entails, whether design thinking *competes* conceptually with existing IT development techniques, or if it is regarded as contribution to those techniques. We found quite a clear picture. None of our interviewees sees design thinking as a clear-cut alternative to existing software development approaches, independent of the approval of agile approaches as SCRUM or sequential approaches such as the waterfall model.[2] Instead, the general focus is on the learning aspect of design thinking regardless of what development approach is in favor. The following quotes exemplify this: *"(Design Thinking) is imagining, understanding a problem space, and eventually the search for solutions, whereas one lets things drift at times, without any restrictions imposed upon oneself, but open for all possible kinds of ideas, then however making very quick steps to find out what is viable and what is not."* This interviewee, a software developer, emphasizes the value of design thinking in fast-track (and thus inexpensive) learning about problem space and potential solution paths outside the prearranged restrictions (that IT development altogether would entail). Another interviewee points out the difference between design thinking as a learning approach and developing itself: *"Design Thinking does not guarantee an outcome. That is completely in conflict with the idea of working with uncertainty. So, understand that forming fast, lean, simple, even with prototypes that are reflective of the end state, you are not moving forward to but make you smarter about how the end state should be. That in itself is a tool. It is not an alpha release (...). It's just a thinking tool to understand the problem."* We see that design thinking is regarded as a contribution to a certain notion on software development in general, namely 'how to build up a novel and viable design' – whereas the notion 'how to build up a functioning IT system in time and budget', which every IT development process has to tackle as well, does not play any role. This however suggests that the first notion has not been effectively addressed in IT engineering as otherwise design thinking would have been perceived rather redundant than contributing. As one lead IT developer stated about design and development in general: *"There is no specific statement on design in the building process, so that the build process doesn't say anything about design. But it is up to the individuals who implement the build process and then apply their*

[2]Albeit the majority of our interviewees showed preferences for agile development approaches, sequential approaches are favored when it comes to large-scale IT development projects due to its better planning reliability.

design thinking based on their understanding." This statement exemplifies the inherent 'technical bias' in IT development (see chapter "Tele-Board: Follow the Traces of Your Design Process History"). It indicates that the process of building software is more constituted than the process of designing software, so that the question how to build a viable design is likely to be subordinated to how to build a functioning system. The knowledge gap that this imbalance creates seems to be the reason why design thinking attracts developers.

Hypothesis 3. There are two groups of language games when it comes to implementing or adopting design thinking to IT development: the firsts treats design thinking and IT development as two separate worlds, and the second as an integrated one.

Our third hypothesis relates to both questions: how people speak about the organizational implementation and how they speak about the personal adoption of design thinking to IT development. We found equally two underlying language game patterns, namely that design thinking is treated (a) as an external, self-contained matter linked to but not into integrated to IT development, and (b) as an influence to change IT development itself. We call these patterns the 'two-worlds games' and the 'one-world games' respectively.

When it comes to implementaion, the 'two-worlds games' manifests in the idea that design thinking is realized in a project prior to the actual development process. Our interviewees describe this either as a service by an external 'task force', and/or as a form of workshop in which also some developers contribute substantially so that they can act as "design advocates" in the development process later on. However, the crucial transition between both worlds is generally a prototype as the outcome of the design thinking process that ought to serve as a starting point for the development process – and thus gets *"thrown over the fence"*, as two interviewees say. Against that background, discussions on adopting design thinking lead to controversies on how design thinking prototypes can be picked up by the developers later on. One interviewee sees different conceptions of prototypes as a hurdle between both worlds. He stresses that design thinking prototypes eventually embody completed concepts *"to which you can get down afterwards asking: how can we translate this to a real product?"*, whereas prototypes in IT development instead initiate a process of conceptualizing: "*You build software prototypes because you have otherwise nothing to look at when you discuss what you actually need – which would be extremely difficult. (...) It is easier to define requirements in the software world as a delta to something existing*." This discussion shows that there is a danger of misconception at the transition between both worlds. Developers are more used to treating prototypes as a form of tangible assistance for the development process and not as a non-technical blueprint for the final product. Connecting both worlds is regarded as critical as the following statement exemplifies: *"(...) design thinking people should learn that what they deliver is not enough for that what developers need. On the other hand, also developers should learn that sticking post-its casually and creatively, filling them with writings and permanently rearranging them is also serious and valuable work that provides results that the developers need afterwards."*

We found many perspectives on how to merge both worlds to a ‘one-world game’. The most general of these views treats design thinking as an *imperative*, or rather as a kind of ‘wake-up call’ for developers to alter the way they work. Implementation happens in this sense through people who take this up and change their mindsets and problem solving routines. As one of our interviewees states: *“It is cultural change. The people just have to learn to change their views.”* This is rather a symbolic approach of implementing, as it is about demonstrating the benefits of design thinking-led problem solving and asking people to internalize and to apply it. Yet, as one interviewee stated about the workshop experience: *“Many were enthusiastic about it. Many said: ‘I want to adopt it somehow, I just do not know how,’ or: ‘how can I tell my boss?”*

As this quote exemplifies, we were able to identify a strong tendency that when design thinking is communicated as an appeal to developers, they appreciate the general idea but doubt being able to apply it within the tight frames of a development organization. Moreover, we found out that there is severe risk perception involved. Many had problems aligning the openness and the ‘explorative detours’ in design thinking with common performance measurement systems for IT development projects that rely on project plans, milestones and punctual shipment: *“When you are under time pressure and have to finish your tasks, then you refer to what you are assessed by and what you have to fulfill. Those things where everybody would say, ‘yes, that would make sense’ are skipped anyway.”* Against the background of those organizational practices, design thinking is perceived as an uncertain method, which may be helpful to come up with innovations, but also entails a high planning (and justification) risk. The willingness to apply design thinking in daily work therefore ends when superiors ask for results without explicitly backing the use of a design thinking approach.

We found two views that try to overcome this dilemma. Some suggest to implement design thinking in the form of an obligatory phase of development processes, others intend to translate design thinking to an adaptive toolbox that can be applied by developers depended on what kind of problem they are faced with (instead of in which phases they are). The first would treat the learning effect of design thinking itself as an objective that has to be achieved; the second would make the use of design thinking more flexible so that requests of using design thinking methodology could be formulated very specifically. However, within the frame of our study, both ways of implementing design thinking were not realized so that we could not gain further insights about them.

4 Conclusion

In summary, this discussion confirms the observation that the range of perceptions of design thinking diverge as more applied people think and speak about design thinking. Table 2 summarizes the variety of views on how to implement and adopt design thinking in the IT industry.

Table 2 Language games on implementing and adopting design thinking

	Implementing	Adopting
Two-worlds games	DT as a foregoing project; DT as a service	'Picking up what is thrown over the fence'
One-world games	DT as a 'wake-up call'; DT as a process phase; DT as an adaptive toolbox	'DT is appealing, but needs backing by the organization'; 'Choosing DT tools when it helps'

Drawing on the image of 'languages games', we showed that there is no single way of how meaning is created about design thinking in IT development, but rather an evolving variety of ways. This was substantiated in the hypotheses 1 and 2 pertaining to the range of understandings of design thinking in general and in IT development in particular, as well as in hypothesis 3 as for matters of implementing and adopting. We regard an initial incongruity between design thinking and IT development as a basic cause for this juxtaposition of language games. Design thinking is not a concept that seamlessly infixes as a further development approach to the IT world. Instead, it is a self-contained methodological field that can serve as an example to tackle shortcomings of established IT development approaches, i.e. the technical bias, by suggesting further attitudes towards knowledge and categories of knowledge for IT development. As a result, it remains fuzzy what exactly the overlaps between design thinking and IT development are like. Applying design thinking to IT development thus presupposes strong *translation efforts* that set off – as shown in our study – the emergence of divergent and partly incongruent language games.

We regard the resulting juxtaposition of language games as both helpful and destructive. It is helpful when it stimulates reflection and awareness of the constraints and limitations of established IT design approaches. It can be destructive when it comes to implementing and adopting design thinking, as there is a danger that parallel meanings weaken the communicability of the concept, and dissolve it in the end within a 'semantic nirvana'. As implementation and adoption require clear-cut concepts, it is not surprising that our interviewees tried to wipe out the fuzzy overlaps between both worlds either by separating them into two distinct worlds, or by merging them into one integrated world. With both ways, our interviewees sought to bring clarity to the rules determining the language games on design thinking in IT development. This shows quite plainly that promising attempts to implement or adopt design thinking presuppose clear images of how design thinking can be thought within IT development. On the basis of our study, we can distinguish different models connecting design thinking and IT development processes:

– In the **split project model**, design thinking is handled as a separate process performed by a specialized "design thinking team" before the IT-development process starts (→ design thinking as a service). Its main purpose is to map out potential directions in terms of user needs and to inform the IT development process with an initial "package" that is handed over to the subsequent development process.

- In the **overlapping teams model**, an initial design process is likewise used to inform the subsequent development process. But instead of "throwing the package over the fence", one or more project members of the development team participate in the design thinking process to be able to act as an communication agent to explain and maintain the gained design knowledge throughout the development process.
- In the **unified project model**, design thinking is a central technique for the front-end of the development process itself. The overall process is changing from a design thinking to an IT development process when the conceptions of problem und solution are specified enough to translate them to development tasks. This implies that there is a strong overlap of personnel and management responsibilities between both the design thinking and the development phases.
- In the **toolbox model**, design thinking is not regarded as a distinct project or process phase, but as a bundle of methods developers can draw on to solve certain design problems they could not solve by means of common IT development methods. In this case, design thinking is narrowed down to a well-defined box of tools for adaptive support.

However, also this range of models carries some inherent contradictions due to the fact that the respective implementation strategies imply different conceptualizations of design thinking itself. A design thinking toolbox, for instance, focuses rather on handy and selective techniques, while a split project focuses also deliberately on a coherent design thinking process carried out by skilled personnel. This observation is supported by our first hypothesis, stating that design thinking is more apparent as a general concept than as an applied one. Implementing design thinking seems to be thus strongly connected to a conceptualizing process – and a wide range of potential design thinking implementation models seems to be an inevitable consequence. To further explore and to develop the range of models in greater detail will be an important task for both future research and management practice.

References

Beck, K.; Andres, C. (2004) Extreme Programming Explained – Embrace Change, Addison Wesley-Longman, Amsterdam.

Beckmann, S. L. and Barry, M. (2007) Innovation as a Learning Processes - Embedded Design Thinking, *Californian Management Review*, 50(1), 25–56.

Bogner, A.; Littig, B. and Menz, W. (eds.) (2005) *Das Experteninterview – Theorie, Methode, Anwendung*, VS-Verlag für Sozialwissenschaften, Wiesbaden.

Brown, T. (2008) Design Thinking, *Harvard Business Review*, June, 84-92.

Buxton, B. (2007) *Sketching User Experiences: Getting the Design Right and the Right Design*, Morgan Kaufmann, San Francisco, CA.

Cross, N. (2007) Designerly Ways of Knowing, Birkhäuser, Basel etc.

Dorst, K. (2006) Design Problems and Design Paradoxes, *Design Studies*, 22(3), 4–17.

Dorst, K.; Dijkhuis, J. (1995) Comparing paradigms for describing design activity, Design Studies 16, 261–274.

Krippendorf, K. (2006) *The Semantic Turn – a New Foundation for Design*. Taylor & Francis, Boca Raton etc.

Kuckartz, U. (2007) *Einführung in die computergestützte Analyse qualitativer Daten*, VS-Verlag für Sozialwissenschaften, Wiesbaden.

Lawson, B. (2006) *How Designers Think – the Design Process Demystified*, Architectural Press, Oxford.

Lindberg, T; Meinel, C. (2010) Design Thinking in the IT Industry?, Electronic Colloquium on Design Thinking Research, Report No. 1.

Lindberg, T.; Noweski, C. and Meinel, C. (2010) Evolving Discourses on Design Thinking – How Design Congnition Inspires Metadisciplinary Creative Collaboration', *Technoethic Arts*, 8(1), 31–37.

Mandel, T. (1997) *Elements of User Interface Design*, John Wiley & Sons, Hoboken etc.

Owen, C. (2006) Design Thinking – Notes On Its Nature And Use, Design Research Quarterly, 1:2, December, 16-27.

Plattner, H.; Meinel, C. and Weinberg, U. (2009) *Design Thinking*, mv-verlag, Munich 2009.

Pichler, R. (2008) Scrum – Agiles Projektmanagement erfolgreich einsetzen, dpunkt.verlag, Heidelberg.

Rittel, H.W. J. (1972) On the Planning Crisis - Systems Analysis of the First and Second Generation. Bedrifsokonomen No. 8, 390–396.

Schön, D.A. (1983) The Reflective Practitioner – How Professionals Think in Action. Perseus Books, New York.

Strauss, A.L. (1998) *Grundlagen qualitativer Sozialforschung*, Wilhelm Fink Verlag, Munich.

Vredenburg, K.; Isensee, S. and Righi, C. (2002) *User-Centered Design – An Integrated Approach*, Prentice Hall, Upper Saddle River.

Wittgenstein L. (1984) *Tractatus logico-philosophicus*, Suhrkamp, Frankfurt a.M.

Determining the Effect of Tangible Business Process Modeling

Alexander Luebbe and Mathias Weske*

Abstract We have created a haptic toolkit that people can use to map and discuss their working procedures. We call it tangible business process modeling (t.BPM). Process modeling is an approach to capture work items, their order constraints, the data processed and people responsible in a graphical model. Typically, experts create these models using software tools. Domain experts are questioned but passive when the model is created. Our approach uses a set of plastic tiles and whiteboard markers for modeling. Thereby, we can engage novice users into shaping their processes at the table.

In the first year, we iterated towards the solution. While we are convinced that our approach yields advances, scientific investigation was yet missing. In this year, we have conducted a controlled experiment that compares t.BPM to structured interviews. We found that people have more fun, learn more, do more reviews and corrections with t.BPM. Finally, people take more time to think and talk about their processes. In this chapter, we outline our approach and research agenda. We present the experiment setup and results. Finally, we explain our next steps towards method development.

1 Introduction

Business process modeling is the act of visualizing the knowledge about work procedures in an organization. The model is a shared representation used in discussions about the process. Business process modeling is a common activity within the scope of business process management which is a holistic approach to

A. Luebbe • M. Weske (✉)
Hasso Plattner Institute, University of Potsdam, P.O. Box 900460,14440 Potsdam, Germany
e-mail: alexander.luebbe@hpi.uni-potsdam.de; mathias.weske@hpi.uni-potsdam.de

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_14,

structure, measure and coordinate work in organizations. The idea is to investigate, change and monitor procedures that drive the daily business operations. To discover opportunities for improvement, a proper understanding of the current situation with knowledge from interdisciplinary teams is often necessary. All members of the team share the process model as a common point for discussion.

Process models are very popular to facilitate the communication between business and IT departments. Since software is omnipresent in today's organizational environments, changes in the process often affect software systems that need to be changed as well. Supporting business processes with software systems yields great potential to save time, enhance reliability and deliver a standardized result. But to change a process often means to change the software as well. The communication between business and IT departments is therefore crucial to the success of the overall business. It is important that business departments can structure and show their daily working procedures to the IT departments. Likewise it is important for IT departments to communicate opportunities and constraints to the business. Together they can realize the potential by streamlining business and IT activities. Business process modeling can provide a common language for these departments to facilitate their communication.

Current State of the Art

At present, process modeling is a special skill. Typically, business analysts get trained in process modeling and sell this skill as a consulting business. They

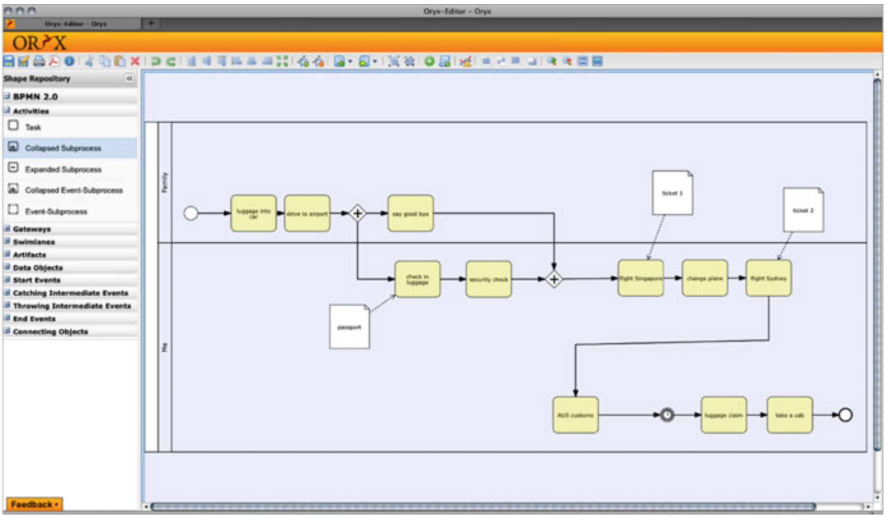


Fig. 1 BPMN diagram modeled in Oryx – a process modeling software

interview the stakeholders of the process and create a model that reflects their understanding of the organizational procedures. The model is created using a specialized software tool. Various expert tools have evolved for that purpose; see Fig. 1 as an example. In any case, the modeling tools remains in the hands of the consultant for efficient use. To gather feedback from domain experts the process is printed and passed back. But often enough the domain experts do not sufficiently understand the models or decide that their knowledge is not well captured. Additional effort is needed to explain the model and correct mistakes. Three to five iteration cycles are a common theme to reach consensus about the final model, in other words, to create a shared view.

This situation was the starting point for our research 2 years ago. We started by prototyping ideas for stronger user involvement and for model building together with the domain experts (Edelman et al. 2009; Grosskopf et al. 2009; Plattner et al. 2010). Our result was an approach that we call tangible business process modeling (t.BPM). It consists of a toolkit and a method for its application.

The toolkit has a set of four basic shapes that reflect the iconography of the Business Process Modeling Notation (OMG 2009). The concepts represented are work items (tasks), events, routing notes (gateways) and the information used in the process (data objects). The thick acrylic tiles are laid out on a table and are transcribed with whiteboard markers. The responsibilities for work items and the actual flow of the process steps are marked on the table with the same whiteboard markers. As you can see in Fig. 2, t.BPM is a low tech tool for process modeling that has the same expressiveness as high fidelity software tools. Indeed the process depicted in Fig. 2 is the same as in Fig. 1.

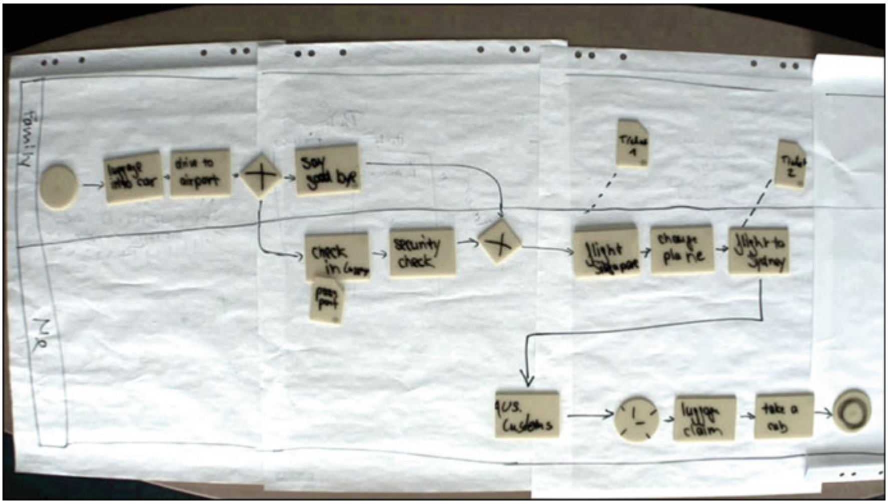


Fig. 2 Same process as in Fig. 1 – modeled with t.BPM

While software tools yield advances such as seamless replication and simulation of models, the key selling point for t.BPM is the actual application together with domain experts in workshops. The participants gather around a table; they discuss and map out their knowledge using the t.BPM toolkit. The process modeling expert is no longer the person to elicit information and create the model. Instead, the method expert becomes a facilitator that helps the domain experts to create their process models themselves. At first, the minimum amount of concepts is introduced to the participants. Namely, the concepts of tasks (work items) and the start and the end event as the scope of the process. If more concepts are needed they are introduced by the facilitator during the modeling session. The goal is to create an initial overview fast and iteratively refine the model. When more detailed information is required, e.g. about the information flowing in the process, the required concept, e.g. the data object, is introduced and the model is iterated. At each stage, the participants must understand the model, because they have to create and refine it. The facilitator ensures that concepts are not misused and helps the participants to frame their knowledge into a process model. It can be helpful to play a game in the beginning. In that case, the participants model an artificial process that everybody is equally aware of, such as withdrawing money from an ATM. The game helps to get into process thinking without getting too much into arguments about the content of the process. While toolkit and application method are equally important, we focus on toolkit for this year's research.

2 Research Framework

In last year's report we presented a series of prototypes that led to the development of the t.BPM toolkit. We also presented our research methodology as depicted in Fig. 3.

By applying a learning cycle, we were able to develop a working solution. We know it is working because we get positive feedback from the workshop participants and the consultants alike. Yet, a proper scientific investigation was missing. In this paper, we present this scientific investigation by means of a

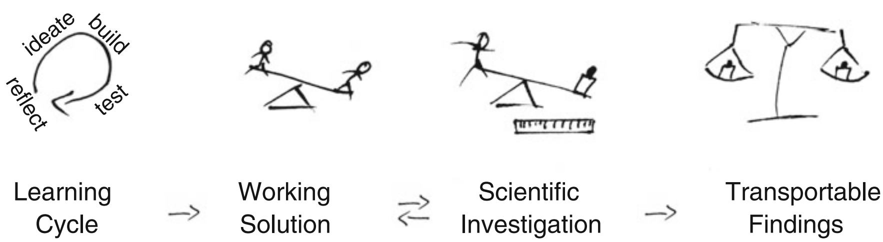


Fig. 3 Sketch of research methodology employed for the t.BPM project

controlled laboratory experiment. By measuring the effect of tangible business process modeling on individual persons, we aim to determine the more transportable findings. We elaborate on this throughout the rest of this chapter.

3 What Makes t.BPM Modeling "Effective"?

We think that t.BPM modeling is more effective than traditional ways of process elicitation. We set out to examine the difference between t.BPM and structured interviews with single individuals. We choose structured interviews because they are seen as the most effective requirements elicitation technique (Davis et al. 2006). We reduced the scope to single individuals to have more control about the variables in the setup and to focus on the effect that t.BPM causes within the individual subject. By "effective" we mean that it produces a "desired or intended result" (Stevenson 2010). In requirements engineering, more information is seen as more effective elicitation. But more information does not imply better information. Moreover, it was already shown in other investigations that visual representations do not necessarily create more information (Davis et al. 2006). We think effective process elicitation has more dimensions, such as user engagement, iterated (higher quality) results and better feedback on process models. We decomposed these areas further into 14 hypotheses based on the following considerations:

3.1 More User Engagement

User engagement, here the degree of participation, is widely recognized as a key factor for success of collaborative efforts (Krallmann et al. 2007; Sedera et al. 2004; Stirna et al. 2007). In HCI research, tangible interfaces (Ishii and Ullmer 1997) are seen as an important factor to impact task engagement. In those cases, engagement is typically measured as the time spent on a task (Xie et al. 2008). We therefore also opt to measure time and hypothesize that people will *spend more time talking* about the process but also *spend more time to think* about what they do. Schaufeli developed different instruments to measure work engagement which he sees as the opposite of a burnout (Schaufeli and Martinez 2002). For him, work engagement has two dimensions, activation and identification (Schaufeli and Salanova 2002). One may argue that activation is already measured with the time spent on the task. But we additionally hypothesize that people have *more fun* and have *more motivation* to accomplish the task, which are other aspects of activation. The aspect of identification inspires us to hypothesize that people modeling with t.BPM are *more committed to the solution* that they shaped. That also means, they would have a *clearer goal* understanding of what they are doing, which we hypothesize.

3.2 Better Information from Elicitation

The cognitive load theory (Sweller and Chandler 1991) postulates that our brain has limited capacity, called work memory. The fundamental insight was first reported by Miller in 1956 who found that people can hold on to "seven, plus or minus two" information pieces at a time without context (Miller 1956). The amount of information to be kept in the work memory can be reduced by externalizing knowledge (Zhang 1997) as it is done with t.BPM or other mapping techniques. Reduced load of working memory enables people to get into details more extensively. Thus, we hypothesize that people share more detailed process knowledge such as *more problems* with and *more phases* in the process when using t.BPM. But as we said in the beginning, better information does not simply mean more information. And it was shown by other researchers that visual representations do not necessarily mean more information is being elicited. We therefore additionally want a measure that indicates the quality of the workshop results. We decided that pragmatic quality of the initial workshop results can be measured by the amount of iterations needed to agree on this process model afterwards. As described above, consultants elicit a process in the workshop, model it afterwards, and then send it out for people to review it, approve it or propose corrections. In t.BPM the result of the initial elicitation workshop is already a process model. Since information is immediately mapped and framed, we hypothesize that people will do *more reviews* of the process model and apply *more corrections* to their initially elicited story when using t.BPM due to the mapping effect. We assume that this leads to better information quality.

3.3 Better Feedback on Process Models

We strongly believe that better feedback is grounded in a deeper understanding. It is suggested that students who actively engage with the material are more likely to

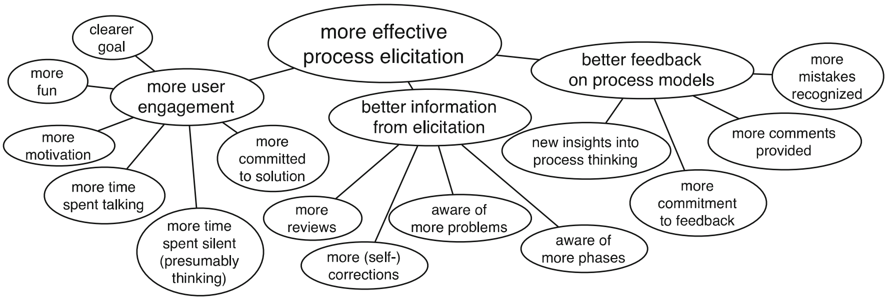


Fig. 4 Hypotheses at a glance – main goal defined with three subgoals – translated into 14 hypotheses

recall information afterwards (Bruner 1961). Recall is the first stage of understanding before retention and generation (Mayer 1989). Consequently, we assume a learning effect for the use of t.BPM and hypothesize that people have *more new insights into process thinking* due to their hands-on experience. Better understanding should also enable people to read and interpret models better. Understandability tests for process models are at their early stage (Melcher 2009; Laue and Gadatsch 2010). Thus, we hypothesize about the positive effect to be expected in the field, e.g. that people with t.BPM experience will *find more mistakes* and *provide more comments* to process models when asked for feedback. Furthermore, we do think that better understanding will lead to *more commitment to feedback* and therefore hypothesize this is an indicator for the understanding that people build.

All hypotheses can be found at a glance in Fig. 4. They are operationalized using either video analysis, questionnaires or process model feedback tests. This is further explained in the experiment planning.

4 Experiment Planning

To test our hypotheses, we designed the following experimental setup, see also Fig. 5. Subjects get first conditioned to a certain level of BPM understanding. Afterwards, subjects are randomly assigned to do either interviews or model with t.BPM. The topic is randomly chosen between buying expensive equipment and running a call for tender. Two experimenters operate the experiment. One guides the subjects in the role of an interviewer, the other experimenter observes the situation and ensures a stable treatment throughout the experiment. They randomly swap roles.

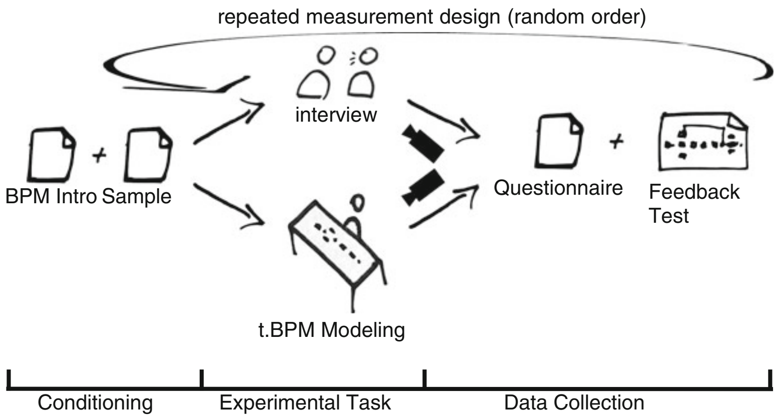


Fig. 5 Experiment design

During the experimental task data is collected using video recording. Afterwards, a questionnaire is to be filled in and a sheet with a process model is handed to subjects. They are asked to provide feedback to process models that depict "finding a new flat" or "getting a new job", chosen randomly. In every step of the experiment, the time is tracked but time constrains are not imposed on subjects. After the first run, subjects rerun the experimental task using the other method and the other process to report on. They do the questionnaire the second time and get the other process model to provide feedback to.

In other words, the sampling strategy is a randomized balanced single factor design with repeated measurements (Wohlin et al. 2000) also known as a within-subjects design (Greenwald 1976). All subjects get both treatments assigned in different order. All subjects do interviews and process modeling. And all subjects get both processes to report on and both feedback tests, again randomly assigned. Subjects are rewarded for their participation with a chocolate bar and a cinema voucher.

5 Experimental Material

As follows, we briefly outline and explain the printed material that we used in the experiment.

- **BPM Introduction**
  A two page document explaining the terms Business Process Management, Business Process Modeling and process models.
- **Sample Model**
  A one page document that depicts the process of "Making Pasta". It also contains a legend of the BPMN elements used and four pragmatic hints on process modeling. In particular, it suggests the balanced use of gateways, an 80% rule for relevance to set granularity, verb-object style activity labels as suggested by Mendling et al. (2009) and a notational convention for conditions at gateways.
- **Task Sheet**
  One paragraph explaining the experimental task. Subjects are asked to model or report on one of the following processes: "buying a new flat screen for the entrance to the company building" or "running a call for tenders to build a new warehouse". The introduction explicitly sets the context, the start and the endpoint of the process.
- **Interview Guide** (for Experimenter)
  Experimenters guide through the modeling/interview by asking the same six questions in the same order in the experimental task. It started with "Please identify all relevant steps", went on with "Which documents play a role?" and concluded with "Which problems are you expecting in this process" and "Is there anything else you want to tell us about the process?" Experimenters read out the exact questions from the interview guide. The interview guide also contains standardized answers to questions from participants, such as "Make an assumption and proceed from there".

- **Questionnaire**
  Eighteen items to be rated on a 5-point Likert scale. Three items operationalize one hypothesis. In each case, two items are formulated towards the hypothesis, one is negatively formulated. For negatively formulated items we turn the values around (6-value) to retrieve the actual variable to work with.
- **Feedback Test**
  It contains, a process model, a sample annotation and the request to "provide feedback" to the model. Two versions of this test exist. One on "Moving to a new flat" and another one on "Getting a new job". The process models contained problems that we intentionally built into them. However we consider all feedback given to the model as valuable.

6 Participant Selection

The sample population, used in research studies, should be representatives of the population to which the researchers wish to generalize (Cooper and Schindler 2008). Thus, we want potential users of t.BPM to participate in the study. In prototyping sessions we identified clerks as the most suitable target group. They run processes on an operational level and might be questioned in business process elicitation projects as stakeholders of the processes.

We contacted a trade school in Potsdam (Germany) and got access to run the experiment on-site. Among other professions, the trade school educates office and industrial clerks. Industrial clerks do planning, execution and controlling of business activities. Office clerks do supporting activities in a department, e.g. as office managers. On the job, both professions might overlap depending on the size of the company. Both groups might be questioned in process-oriented projects by external consultants. Thus, they represent the target population that we like to address with t.BPM.

7 Experiment Execution

The experiment design was executed in December 2009. The experimenter team was located in a lecture room at the trade school in Potsdam for 1 week. Within this week 20 slots were offered to the students by short teasers given in the classes. Students could choose to swap one lecture unit for experiment participation. Seventeen students did take part during the week. Each experiment run started with a short informal warm-up chat and then followed the design as outlined above. One experimenter ran the experiment, the other one operated the cameras and observed the situation to ensure a stable treatment. Figure 6 depicts sample photos from the two experimental tasks as taped by the video cameras.

We expected to test industrial clerks only. We wanted the most homogenous group possible and we were told, that industrial clerks were in a non-crucial phase

Fig. 6 Experiment execution as taped by the video cameras

of their studies. However, during the week it was not possible to recruit enough industrial clerks. Thus, we opened up the experiment design to both groups, office and industrial clerks. We ended up testing seven office clerks and ten industrial clerks. As explained before, both groups represent potential users of the tool.

8 Data Analysis and Findings

8.1 Descriptive Statistics

From 17 students, we collected one questionnaire and one feedback test per run. From two runs, this results in 34 questionnaires and 34 feedback tests. With 18 items per questionnaire 612 statements were collected in total for evaluation. Furthermore, we conducted video analysis based on 6.74 h of video material.

Videos taken during t.BPM sessions took 20 min (19.52) on average ranging from 10 (10.25) to almost 40 min (38.98). On the other hand, interviews took about 5 min (5.42) on average ranging from 3.5 (3.53) to 10 min (9.68) at most. The differences in duration of talking and silence are distributed correspondingly.

8.2 Statistical Analysis

To analyze our data we use a repeated measurements analysis of variants. The analysis of variance (ANOVA) is a family of statistical tests to compare groups in different conditions and explain the variation in a set of dependent variables with the variation from one independent variable, in our case the method. To do that, the data set is partitioned and sums of squares of deviations from the mean value per group are compared. We use a special case, the repeated-measures ANOVA, which allows us to distinguish the variation between individuals and within each individual. Moreover, we can determine the variation within each individual that can be explained by the method. The result is a set of values that indicates whether a hypothesis holds. Meaning it is true with a probability of error that is less than 5%, called the significance level. Additionally, we conducted a dependent t-test. It is a statistical method to compare groups in different situations and determine whether the effect is significant. Again it means that with a probability of error that is less than 5%, the difference is to be expected in the real population (scaled from our sample size). An extensive reporting on the statistical analysis of this experiment can be found in (Luebbe and Weske 2010). We take a shortcut to the findings here.

8.3 Findings

In particular, we found by video analysis that subjects do significantly more corrections in t.BPM sessions (mean values t.BPM = 3.00, interview = 0.3). We counted a correction if the context of an already explained process part is explicitly changed. In t.BPM sessions this involved re-labeling or repositioning that impacts the process model's meaning. In interviews, explicit revisions of previously stated information were considered corrections. A similar effect was described by Schneider (2007) who observed instant feedback through immediate information mapping.

Furthermore, participants spent significantly more time silent in t.BPM sessions (mean values t.BPM = 5.54 min, interviews = 0.95 min). Although we can only judge on the observed behavior, we interpret the silent time as time taken to think about the process. We conclude that t.BPM affords people to think more extensively in elicitation sessions in contrast to interviews in which talking is the purpose of the session.

Finally, participants reported significantly more insights into process thinking when using t.BPM (mean values t.BPM = 3.75, interviews = 3.43 on a five point likert scale). The difference, however, is not too significant, and when comparing the perceived measures with actual performance of the participants there is no correlation. In other words, people believe that they have learned. But it does not mean they really perform better, for example with respect to the process model feedback test.

Only three out of the 14 hypotheses did hold (see above) based on rigor scientific standards. In the remaining part of this section, we sketch out the other hypotheses before we move on to discuss what this all means to us.

A set of three more hypotheses could probably be accepted with a slightly larger sample set. Namely, participants reported to have had significantly more fun (mean values t.BPM = 4.16 interview = 3.90), we have observed them to spent more time talking (mean values t.BPM = 4.65 min interview = 3.49 min), and do more reviews (mean values t.BPM = 0.81 interview = 0.19). Yet we did not formally accept these hypotheses because the confidence intervals were not sound. The confidence interval describes a range around the mean values. The real mean value is within that range with 95% probability. For the three hypotheses stated above, we cannot say with scientific rigor that they hold. But we will come back to them in the discussion.

Most other hypotheses simply did not hold. People did not find more mistakes or give more comments to the process model in the feedback test because of t.BPM. Moreover, there was no learning effect whatsoever. For example, people did not find other, or different mistakes in the first or second test. Nor did they after modeling with t.BPM. We conclude that people did not acquire better process understanding, although they say they did. But we have to admit that better understanding was measured here simply by the amount of feedback provided. Properly tested process model understandability tests are ongoing research (Laue and Gadatsch 2010).

For motivation, we assume a ceiling effect. There is a slight difference in the motivation reported that is in favor for t.BPM (mean value t.BPM = 4.45, interview = 4.37) but this is not significant. On a five point Likert scale (values [1..5]) it shows for both groups a very high level. In other words, even if t.BPM was more motivating for participants, we could not measure a significant difference at the edge of our scale. We conclude that chocolate, a cinema voucher and time off from class have been superseded by any effect possibly caused by t.BPM.

Subjects also did not find more phases or problems in the process that they reported. As indicated earlier, information mapping does not necessarily lead to more information. Nevertheless it was interesting to see that an average of three phases and two problems was a satisfying answer for almost every participant.

Ultimately, people did not report more clarity of the goal, more commitment to the solution or more commitment to feedback by using t.BPM as a method. Investigating the data further showed, that clarity significantly grows in the second run, no matter what method was used. With clarity comes the commitment to the solution. We conclude that this is a learning effect due to our repeated-measurement design. In other words, iteration matters more than method, at least for clarity and commitment.

9 Experiment Discussion

Only three of our hypotheses can be accepted with scientific rigor (more silence, more corrections, more insights) Three more might hold if the sample size was bigger (more talking, more reviews, more fun). We conclude that,

- t.BPM creates a different working mode. I.e. people talk more and think more about their process.
- t.BPM fosters instant feedback. I.e. people review their process more often and also apply more corrections during the elicitation session.

- t.BPM is fun to learn with. I.e. people report to have more fun and more new insights into process modeling.

We also learned from those hypotheses that did not hold and the combination of data collected. For example, we could not find that people are aware of more problems and phases through the use of a mapping tool. We could also not find that people have more clarity or commitment with t.BPM. Instead, we found that repetition helps people to build a better understanding of the expected task outcome.

Although people reported more fun and insights about their process, we found no indication that people build up formal process knowledge (Grosskopf and Weske 2010). We base this on the feedback test that showed no additional process modeling knowledge due to the tool used. That indicates to us the importance of professional guidance in a process mapping session.

9.1 Transportable Findings

We think the findings about t.BPM can be generalized from the sample group to our target population. All participants are affiliated with companies and represent exactly the group we want to address with the t.BPM tool. Moreover, we think we can generalize the effect of tangible prototyping when compared with pure talking.

We have observed that people spent significantly more time thinking and talking if t.BPM, an external visualization, is present. The same treatment also led to significantly more corrections. We think that the affordance of an external visualization in addition to the discussed knowledge is nothing specific to t.BPM. Schneider (2007) reported similar effects when using instant visual representation of information.

The aspect of tangibility enables novices to easily work with the representation and express their knowledge. We think this leads to deeper involvement and a stronger learning effect (here insights) through hands-on experience. However, a different test would be needed to determine exactly which aspects are provoked by visualization and which ones by tangibility. In any case, the findings should hold for other knowledge representations as well, not only processes.

9.2 Lessons Learned

This experiment was the first one of this size for us. We got advice from researchers in social sciences and psychology. However, some insights come only from doing. So, if we had to start over again, we would reduce the scope to those six interesting hypotheses. Now that we learned of six possible effects, we can reinforce or show them.

We also learned that the perceived measures do not relate to the actual task performance of participants. In particular, we found that there is no correlation between the errors found in the feedback test and the reported insights into process modeling. That means, perceived measures should always be matched with objective measures such as video analysis or tests. For us, the video-analysis was the most powerful evaluation tool. However, rigorous video analysis is also the most time-consuming evaluation task.

Finally, the compact on-site experiment was a good idea. Instead of spreading it out over various weeks with changing conditions, we collected the data in a compact week with a stable setup. Moreover, the two experimenters who reviewed each other's work did ensure a stable setup.

10 The Next Step – t.BPM in Group Modeling Sessions

Determining the effects of t.BPM on individuals was an important first step to understanding why t.BPM works and how it supports process modeling. We think that t.BPM can leverage its full potential in groups where stakeholders come together to create a shared process understanding using t.BPM as a tool. While the effects on individuals will hold in those situations, additional aspects become part of the considerations.

Given that the participants are novices in process modeling, we need to develop an efficient introduction for them to get started with t.BPM modeling quickly. A mini-game might be suitable here. We have made first good experiences and will follow up on this to develop it further. Most importantly, the groups need guidance by a professional in process modeling. These people are experienced in process elicitation and framing. They can guide the domain experts when they map their processes. We want to guide those facilitators in their job. The goal is to provide them best-practice hints that they can use when applying t.BPM in their modeling project. To work out these aspects, we think an artificial setting, such as we created for individuals, is no longer suitable. Instead, we need to go out and assess our ideas in real projects. Only then, can we ensure that the result is usable in practice.

10.1 Action Research

The aspiration to evaluate our ideas in real projects requires new scientific tools for development and evaluation. For example, different projects might vary in setup, member skills, management support, goals of the modeling effort and more variables. The modeling effort becomes an inherently complex social problem. We therefore need a research method to guide us in such an environment.

In 1946, Kurt Lewin proposed such a method called "Action Research". In his words it was made for scientists building "hard hitting teams of practitioners" (Lewin 1946). He argues that theoretical science fails to address complex social problems. Thus, instead of isolating and theorizing a problem and solution, these types of problems should be solved within the domain. Solutions should be

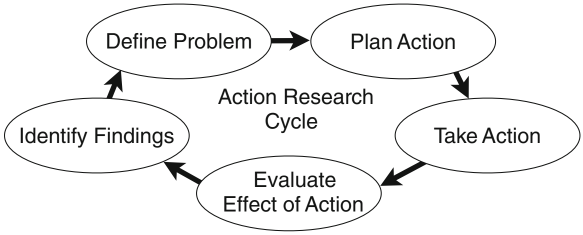


Fig. 7 Action research cycle (Adopted from Susman and Evered 1978)

implemented directly within the social context and evaluated for their effects. Different streams of Action Research have evolved from there typically distinguished in diagnostic, empirical and participatory Action Research (Susman and Evered 1978), which relate to the degree of involvement that the researcher has in the situations that is being investigated.

Action research happens in a learning cycle, see Fig. 7. It starts by defining a problem to be solved. In the next stage solution ideas are developed, alternatives are considered and the action is selected. The selected action is then implemented, called “Take Action” in Fig. 7. Data about the action is typically collected through observations, interviews, or questionnaires (Susman and Evered 1978). The data is used to study the consequences of an action in the subsequent evaluation phase. The generalized learnings derived from the evaluation are the identified (transportable) findings. That implies abstracting from the actually observed situation and identifying more broadly applicable knowledge. The findings may also yield new insights about options for improvement or side effects of actions taken. These findings are used for the next iteration of the research cycle. Again a problem is defined and all phases are conducted. Action research projects may differ in the number of phases carried out by the researcher, the client or the both together.

For our work, the client is a process consultant applying t.BPM. From previous observations, we can recommend initial guidelines for the t.BPM application, the first action planning. The consultant implements the action itself. We observe and collect data, which is then evaluated with respect to the guidelines proposed, and the remaining problems to be solved. These are the findings that set the path for the action research cycles to come.

11 Conclusion

This paper reports on the second year of the t.BPM developments. While the first year was on prototyping and tool development, this second year was about the effect that the t.BPM toolkit causes within subjects. We conducted a controlled experiment with student clerks in a trade school in Potsdam. We found that people using t.BPM take more time to think about their process, they apply more changes and they report more insights as opposed to people doing interviews. We think that these findings hold for all tangible modeling approaches when compared to talking.

While we have now better understood which aspects make tangible media attractive to individuals, we seek to apply this in reality with groups of modelers. This chapter is embedded into a larger research endeavor. In the next year, we will conduct use action research as a scientific method to study tangible business process modeling and create method guidance for consultants who want to use our tool.

Acknowledgements We are grateful to the students that supported this work. First and foremost Karin Telschow. She helped setting up, running and evaluating this experiment. Likewise, Markus Güntert helped to setup and run the experiment. Finally, we thank Carlotta Mayolo for her support in the video analysis phase.

References

Bruner, J.S., 1961. The act of discovery. *Harvard Educational Review*, 31(1), p.21–32.

Cooper, D. & Schindler, P., 2008. *Business Research Methods*, McGraw-Hill Higher Education.

Davis, A. et al., 2006. Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review. In *14th IEEE International Conference Requirements Engineering*. p.179–188.

Edelman, J., Grosskopf, A. & Weske, M., 2009. Tangible Business Process Modeling: A New Approach. In *Proceedings of the 17th International Conference on Engineering Design, ICED'09*.

Greenwald, A.G., 1976. Within-subjects designs: To use or not to use. *Psychological Bulletin*, 83(2), p.314–320.

Grosskopf, A., Edelman, J. & Weske, M., 2009. Tangible Business Process Modeling – Methodology and Experiment Design. In B. Mutschler, R. Wieringa, & J. C. Recker, hrsg. *1st International Workshop on Empirical Research in Business Process Management (ER-BPM'09)*. Ulm, Germany: Springer Verlag, p.53–64.

Grosskopf, A. & Weske, M., 2010. On Business Process Model Reviews. In *workshop proceedings of ER-POIS: Empirical Research on Process Oriented Information Systems affiliated to CAISE10*. p.31–42.

Ishii, H. & Ullmer, B., 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. p.234–241.

Krallmann, H., Schönherr, M. & Trier, M., 2007. *Systemanalyse im Unternehmen*, Oldenbourg Verlag.

Laue, R. & Gadatsch, A., 2010. Measuring the Understandability of Business Process Models – Are we asking the right questions? In *Proceedings of the 6th International Workshop on Business Process Design (BPD 2010)*.

Lewin, K., 1946. Action research and minority problems. *Journal of social issues*, 2(4), p.34–46.

Luebbe, A. & Weske, M., 2010. *The effect of tangible media on individuals in business process modeling – a controlled experiment*. Technical Report 43. Hasso-Plattner-Institute for IT Systems Engineering. Available at: http://bpt.hpi.uni-potsdam.de/Public/AlexanderGrosskopf

Mayer, R.E., 1989. Models for understanding. *Review of educational research*, 59(1), p.43.

Melcher, J. u. a., 2009. On Measuring the Understandability of Process Models (Experimental Results). In *Proceedings of the 1st International Workshop on Empirical Research in Business Process Management (ER-BPM)*.

Mendling, J., Reijers, H. & van der Aalst, W.M., 2009. Seven Process Modeling Guidelines. *Information and Software Technology (IST), 53*. p.127–136.

Miller, G.A., 1956. The magical number seven, plus or minus two. *Psychological review*, 63, p.81–97.

OMG, 2009. *Business Process Modeling Notation (BPMN) 1.2*, OMG.

Plattner, H., Meinel, C. & Leifer, L.J. hrsg., 2010. *Design Thinking – Understand, Improve, Apply*, Springer.

Schaufeli, W.B., Martinez, I.M. u. a., 2002. Burnout and engagement in university students: A cross-national study. *Journal of Cross-Cultural Psychology*, 33(5), p.464.

Schaufeli, W.B., Salanova, M. u. a., 2002. The measurement of engagement and burnout: A two sample confirmatory factor analytic approach. *Journal of Happiness Studies*, 3(1), p.71–92.

Schneider, K., 2007. Generating Fast Feedback in Requirements Elicitation. *Lecture Notes in Computer Science*, 4542, p.160.

Sedera, W. u. a., 2004. A success model for business process modeling: findings from a multiple case study. In *Proceedings of the 8th Pacific-Asia Conference on Information Systems, Shanghai, China*.

Stevenson, A., 2010. *Oxford Dictionary of English*, Oxford University Press.

Stirna, J., Persson, A. & Sandkuhl, K., 2007. Participative Enterprise Modeling: Experiences and Recommendations. *Lecture Notes in Computer Science*, 4495, p.546.

Susman, G.I. & Evered, R.D., 1978. An assessment of the scientific merits of action research. *Administrative science quarterly*, 23(4), p.582–603.

Sweller, J. & Chandler, P., 1991. Evidence for cognitive load theory. *Cognition and Instruction 8(4)*. p.351–362.

Wohlin, C., Runeson, P. & Höst, M., 2000. *Experimentation in software engineering: an introduction*, Springer Netherlands.

Xie, L., Antle, A.N. & Motamedi, N., 2008. Are tangibles more fun?: comparing children's enjoyment and engagement using physical, graphical and tangible user interfaces. In *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*. New York, NY, USA: ACM, p.191–198.

Zhang, J., 1997. The nature of external representations in problem solving. *Cognitive science*, 21(2), p.179–217.

Applying Design Knowledge to Programming

Bastian Steinert and Robert Hirschfeld*

Abstract Arguably programming involves design: computational logic – the program – is constantly reorganized to keep complexity manageable and provide for current and future coding activities to be feasible. However, design practices have gained less attention in the field of programming, even though decades of research on design have led to a large body of knowledge about theories, methods, and best practices. This chapter reports on the first results of our research efforts to transfer and apply design knowledge to programming activities. We improved tool support for software developers in two respects, both of which are based on key concepts in design practices: continuous feedback and ease of exploration.

1 Introduction

Agile software development and Design Thinking build on similar values and principles. Agile processes such as Extreme Programming or Scrum are based on short iterations. This approach has many advantages. It results in regular delivery of value to the customer and it enforces developers to constantly face feasibility questions, resulting in feedback on different aspects. Agile processes assume co-evolution of problem understanding and the implementation of a proper solution.

Techniques and values of Design Thinking can be a useful supplement to Agile principles [23]. Both Design Thinking and Agile processes value feedback and encourage team members to interact closely with each other and prospective users. They also emphasize the importance of directness and doing – being continuously involved and in dialog with the product to be created.

B. Steinert • R. Hirschfeld (✉)
Software Architecture Group, Hasso-Plattner-Institute, University of Potsdam, 14482 Potsdam, Germany
e-mail: bastian.steinert@hpi.uni-potsdam.de; robert.hirschfeld@hpi.uni-potsdam.de

* Principal Investigator

H. Plattner et al. (eds.), *Design Thinking Research*,
Understanding Innovation, DOI 10.1007/978-3-642-21643-5_15,

Efforts to bring Design Thinking to the development of software systems should not be limited to the domain of user interfaces and end-user interaction, but needs to be carried far beyond that (Fig. 1).

Developers are constantly involved in design activities while working on a software system. This includes, for example, the selection and representation of domain concepts and the organization of programs in logical units and code entities. Main goals of these design activities are conceptual integrity and ease of understanding. These characteristics are important as software systems are improved and enhanced over time. Requirements change if new functionality needs to be supported and existing functionality must be modified and updated. Every such change builds on the system's current design. New features and modified requirements can be realized more easily, if the system's design features simplicity and ease of understanding. Thus, keeping the software system as simple as possible is an important design goal. Following this line of thought, programming can be regarded as a design discipline that has programmers as affected users of the design outcome.

While programming arguably involves design, knowledge about design has gained less attention in the field of programming. Driven by pure curiosity and also economical interest, the nature of design has been studied for decades [10]. Design-related aspects has been investigated from various perspectives ranging from social sciences over artificial intelligence to brain research, considering design as a collaborative endeavor, as a problem-solving activity, as a conversation with materials, or as hard work towards creative leaps, amongst other. Efforts are put to scientise design [10] to allow for better reflection on design activities and to develop theories and methods that may provide guidance if needed. All this investigations led to a huge body of knowledge about design, the application of which should not be limited to interface design and end-user interaction.

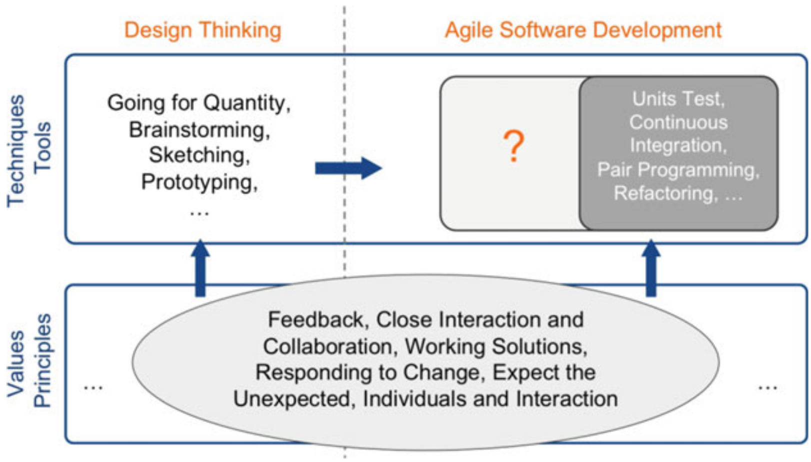


Fig. 1 Learning from design thinking

We investigate the transfer and application of design knowledge to programming activities and the design of software systems. To take advantage of experience from the design domain, software developers need to be provided with both methods and tools that allow them to work and interact with their materials and artifacts as designers do with theirs. We expect that the transfer of such methods and the provision of accompanying tools allow developers to work more efficiently on design tasks.

In this chapter, we present our first results of this research effort. We applied two key concepts of design practice to improve on development support for programming activities, which are described in the next section and the section after next respectively. First, we present continuous selective testing, our approach to provide for continuous feedback on current coding activities and thus allows for instantly assessing their effect. Second, our interactive approach to run-time analysis provides for immediate access to visualizations of run-time information, which arguably support understanding abstractions.

2 Continuous Feedback on Programming Activities

A manual and explicit activity, the frequent selection and execution of tests require considerable discipline. Our approach automatically derives a subset of tests based on actual modifications to the code base at hand, then continuously executes them transparently in the background, and so supports developers in instantly assessing the effect of their coding activities with respect to the overall set of unit tests to be passed. We apply techniques of selective regression testing, mainly relying on dynamic analysis. By taking advantage of the internal program representation available in IDEs, we do not need to rely on expensive comparisons of different program versions to detect modified code entities.

2.1 Motivation

Test-driven development [5] (TDD) is a cornerstone of agile software development methodologies such as Extreme Programming [19] (XP). This technique suggests writing test cases before the code they are intended to cover. Written first, tests serve multiple purposes. First, they represent a specification for the system to be developed. Next, they document the system and help other developers in comprehending the system. Finally, they ensure that every single change violating one of the required features described in the executable form of a test is reported.

While testing is an important part of regular development activities, Integrated Development Environments (IDEs) have little support for selecting and (re-) executing tests relevant with respect to modifications applied to the system under development [16].

There are a few approaches that support (re-) running the test suite automatically every time a file is saved in the IDE [16, 27]. However, *test selection* as such is traditionally not performed: it is always the complete test suite that is run, including irrelevant tests, leading to an execution overhead that is larger than it actually needs to be.

For that reason, developers often manually select a few tests that seem appropriate, run them explicitly, and wait for feedback. The manual, regular, and explicit selection and execution of tests requires considerable discipline. Moreover, success is guaranteed only if no relevant test cases are omitted in the selection. A solution that automatically selects test cases to be executed in the background based on the applied changes to source code is preferable.

Approaches to test case selection are established: *Selective regression testing* [24] has long been a subject of research. Selective regression testing is concerned with reducing the set of tests that need to be executed to detect failures caused by recent modifications to the code base. However, researchers have not yet investigated the potential of integrating this technique into an IDE and having selected tests execute continuously in the background.

We suggest selecting and executing tests automatically whenever the code status demands this. More precisely, it would be desirable to have support for TDD that, whenever source code is changed, *automatically executes exactly those tests that are affected by the actual modification*, giving developers instant feed-back on whether the applied change breaks something or not.

In this section, we describe *continuous selective testing* (CST) and present an implementation thereof in Squeak Smalltalk[1] [18]. Using an implementation of the suggested approach, developers will be supported as follows:

- Sets of relevant tests are selected based on dynamic analysis during the regular execution of tests,
- Relevant tests are executed continuously in the background after every modification to the code base,
- Developers are instantly informed about places in code that, resulting from an applied change, are no longer covered by tests,
- The introduction of new defects is made apparent immediately, which in turn lets developers focus on problems right away.

With that, our approach significantly improves on the way IDE tools provide immediate feedback in a development process adopting TDD.

The main contributions of this work are as follows:

- We present continuous selective testing as an approach relieving developers from the burden to select and run tests explicitly,

[1] www.squeak.org

- We describe how test case selection in general can benefit from the internal program representation already available in IDEs and how differencing of two versions of a program can be avoided,
- We describe our approach to test case selection based on dynamic analysis, being not limited to statically-typed languages.

The remainder of this section is organized as follows. First we summarize TDD and state of the art in tool support. Afterwards we motivate the need for improvement and describe our proposed CST approach.[2]

2.2 Background

First, we will briefly introduce the terms and concepts of TDD. We then discuss current practices of developing tests and application code in accordance with TDD and point out the need for better tool support. Afterwards, we introduce the concepts of regression test selection and discuss current approaches.

2.2.1 The Three Phases of Test-Driven Development

Test-driven development distinguishes three phases of development [5]:

- **Red** Tests are written that specify new requirements on the system in an executable manner. When these new tests are run for the first time, failures or errors occur, as the system does not yet support the new requirements. An important guideline is to avoid writing application code if there is no test case that fails.
- **Green** Developers enhance the code base to make the failed test "green", i.e., run successfully. Adding only functionality that is essential to the test in question is recommended. A successful test signals that the developer is done implementing the new requirement. Note that it might happen that the system already fulfills a newly defined requirement, without adding new code.
- **Refactor** The developer refactors towards the simplest design they can imagine. By definition of refactoring [11], new functionality must not be added during this phase. The tests can ensure that all required and specified features work after a refactoring. Running tests after each and every little change helps to avoid breaking features and provides instant feedback.

We can observe that tests and the regular execution of tests play an important role when developers employ the principles of TDD.

[2]The evaluation of CST is described in the original paper [29].

2.2.2 Tool Support for Test-Driven Development

Best practices in working with tests suggest making only small changes and run tests immediately afterwards to get feedback. This suggestion is based, among others, on the following observations:

- Implementing new application functionality is a very complex activity. As every single step is inherently fault-prone, regular feedback is essential for detecting faults.
- Modifying source code without breaking existing functionality is also difficult. Adapting source code to new requirements or refactoring source code to a simpler design requires very detailed understanding, which is hard to acquire since source code abstracts from concrete execution paths. Having tests covering all parts of the respective code entities and running these tests regularly helps to detect faults early.
- The more steps are passed without getting feedback, the more difficult locating the source of a fault becomes. When a couple of source code entities are changed without running tests, and one or more tests fail later on, isolating the modification that has caused the failure is not straightforward. Typically, developers are unaware of the complete set of modifications done before running the tests. Moreover, multiple failures might have different causes, and combinations of modifications might lead to completely unexpected behavior. To locate the defects, developers can revert modifications step by step or debug the current version. Both ways are tedious and time-consuming.

Running tests often and regularly helps developers to detect faults early, reduces the time required to localize defects, and gives confidence for the next adaptions and refactorings. However, running tests as often and regularly as suggested requires much discipline.

The necessary discipline is sometimes hard to summon, for novices as well as experts. It is all too easy to ignore TDD theory, though well-understood and accepted, and continue modifying code without running tests. It is not necessarily only external factors, such as project schedules, that influence such decisions, but also internal ones like the strong will to finish a task. These aspects contradict the required discipline.

Another issue involving the theory of testing and test-first development is the implicitness of the relationship between test cases and application code they cover. When code is refactored or new features are implemented, existing code has to be modified. However, while developers are aware of recently implemented tests, they cannot know the set of all tests relying on a particular method. Hence, developers do not know the set of tests to be executed after a modification of a particular method. Consequently, all tests should be run after each modification, which is, however, increasingly time-consuming as projects grow. As a result of this, developers run only some tests regularly and the suite of tests is rarely executed, e.g. during integration builds.

Both aspects discussed above, the implicitness of the relationship between test cases and application code as well as the discipline required to run tests after each modification, question the usefulness of tests and test-first development. Our work provides tool support for TDD that alleviates these limitations and strengthens the benefits of testing.

2.3 Continuous Test Queuing, Selecting, and (Re-)Executing

In this subsection, we describe our approach called continuous selective testing (CST). It enables the continuous execution of selected tests directly after code modifications. Such automation relieves developers from the burden of executing tests manually. Selecting a subset of all tests and omitting those that cannot reveal faults reduces execution time and helps to provide feedback instantly. We have implemented the suggested approach in Squeak Smalltalk.

In the following, we will first introduce the concepts of regression test selection and then present the use of the IDE's program representation to detect and handle modifications to the code base. After that, we describe the queuing of tests and the selection and (re-)execution of tests according to the modification at hand. Finally, we present our extensions to the IDE providing instant feedback on test results.

2.3.1 Regression Test Selection

Regression testing refers to the practice of validating modified software; in particular, asserting that applied changes do not affect the software adversely [15]. The simplest approach to regression testing is to reuse the test suite used to exercise the previous version of the software. Fully running a large test suite can be unnecessarily costly, e.g., if only a few parts of the system were changed.

A technique to reduce the number of tests is *regression test selection*. It selects tests that have to be re-run to reveal a fault resulting from a particular change. Selecting an optimal set of tests is, however, generally inefficient [24]. Still, the set of tests traversing modifications can be computed efficiently. This set of *modification-traversing tests* can be considered a superset of the *fault-revealing tests* when the *Proper Regression Testing Assumption* [24] holds (*P* refers to a program and *P'* refers to the modified version of this program):

> When *P'* is tested with *t*, we hold all factors that might influence the output of *P'*, except for the code in *P'*, constant with respect to their states when we tested *P* with *t*.

A regression test selection technique is furthermore considered *safe* if it ensures not to omit tests revealing faults [15]. Several safe techniques have been proposed for purely procedural (e.g., [2, 8, 25]) as well as for object-oriented programming languages (e.g., [15, 26]). Object-oriented programming is special, as inheritance, polymorphism, and thus late-binding have to be considered.

The most efficient test selection technique that is also safe is based on detecting modified code entities, such as functions or storage locations [24]. This technique was first implemented in TestTube [8] for software written in C. The technique is based on dynamic analysis [3]; test coverage information is recorded during each test run. For a new version of a program, the set of modified code entities can be detected. Based on coverage information, the technique selects and re-executes all tests that exercised the modified code entities in the previous version of the software. For object-oriented languages, the modified entity selection technique requires additional considerations due to language features such as inheritance and polymorphism enabling late binding.

Our approach, CST, is based on this technique of detecting modified code entities. CST records coverage information and selects tests on a method level. This procedure may select tests that do not traverse the modifications, because a test might only traverse unmodified parts of a method, for example. However, tracing on a more fine-grained level is much more expensive and does not pay off unless methods contain many control blocks [6].

2.3.2 Propagating Modifications to the Code Base

Most approaches to test selection are based on comparing the new with an earlier program version to detect change entities. Our approach takes advantage of an IDE's internal program representation. Figure 2, on the left, depicts the setup of traditional approaches. IDE and test tools are not integrated and do not work together, rather each of them works separately on external program representations. In this setup, however, a test selection technique requires a comparison of program versions to detect modifications between two versions of a software program. There exist differing concepts and tool for both source code [1, 15] and byte code [17].

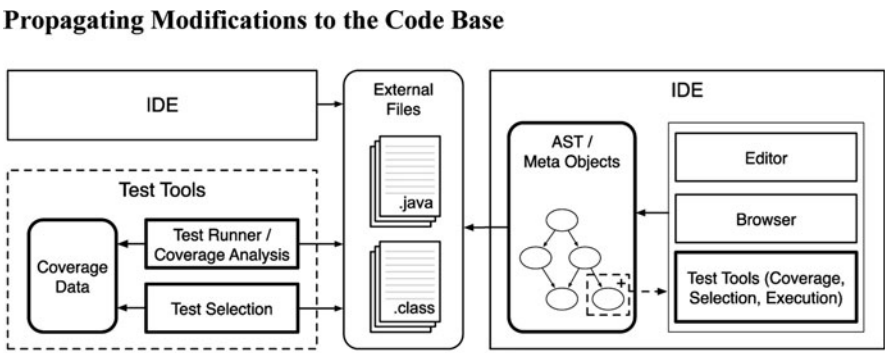


Fig. 2 The *left-hand side* shows a traditional setup where test selection tools and IDE work independently of each other. The *right-hand side* depicts CST integrating test selection into the IDE and taking advantage of the internal program representation

We suggest a better integration of the tools for testing and test selection into the IDE as depicted on the right of Fig. 2. Every modification applied to the code base can produce an event notifying the IDE about the respective change. Using this notification mechanism, the test tools can process each modification to the code base. The tools are now able, for example, to automatically select and re-execute a set of test cases as necessary for the modification applied.

The set of events used to propagate code modifications to IDE tools has to be designed for the particular programming language and IDE, respecting the features of the language and the architecture of the IDE. In Squeak Smalltalk, for example, there are basically two operations to create or modify code objects. Sending a subclass-message *subclass: instanceVariableNames: classVariableNames: pool-Dictionaries: category:* to a class *c* creates a new or modifies an already existing subclass of class *c*. Sending the *compile:* message to a class object allows to compile a source code text of a method and puts it in the method dictionary of the corresponding class. Based on the effects of these two operations, the following change events can be defined for the Smalltalk [13] programming language, which is a rather simple language and does, for example, not provide any visibility modifiers; *class added*, *class removed*, *superclass changed*, *instance variable added*, *instance variable removed*, *method added*, *method modified*, and *method removed*. Note that class-specific ("static") state or behavior do not require special treatment as classes are also normal objects whose state and behavior are defined by meta-classes.

2.3.3 Queuing and Executing Tests for TDD

CST builds upon a well-defined set of different kinds of modification to the system. The event mechanism described above, with the possible modification events it includes, allows for the continuous selection and execution of tests according to the current state of development.

Our approach distinguishes code entity modifications by their referal to test case code or non-test code. By convention, those methods of a class extending *TestCase* that are prefixed with *test* are treated as *test case methods*. Source code entities of test classes that are non-test methods, that is, attributes, *setUp*, *tearDown*, and other utility methods, are treated equally to application code.

When the creation of a test case method or modifications to one are reported, the developer is assumed to be in the red phase of the TDD cycle. The test runner will immediately execute the corresponding test case and provide instant feedback on the result. If the test fails, it will be queued. Failed test cases will be re-executed whenever a modification not related to a test method is reported. Now the developer is expected to be in the green or refactor phase, so the change has the potential to fix a test. All tests that still fail stay in the queue. A change of an entity can fix one or more tests cases, but the change can also introduce a fault that breaks other test cases. All test cases that might be affected by the reported change need to be re-executed. A technique to select the corresponding test cases is presented in the

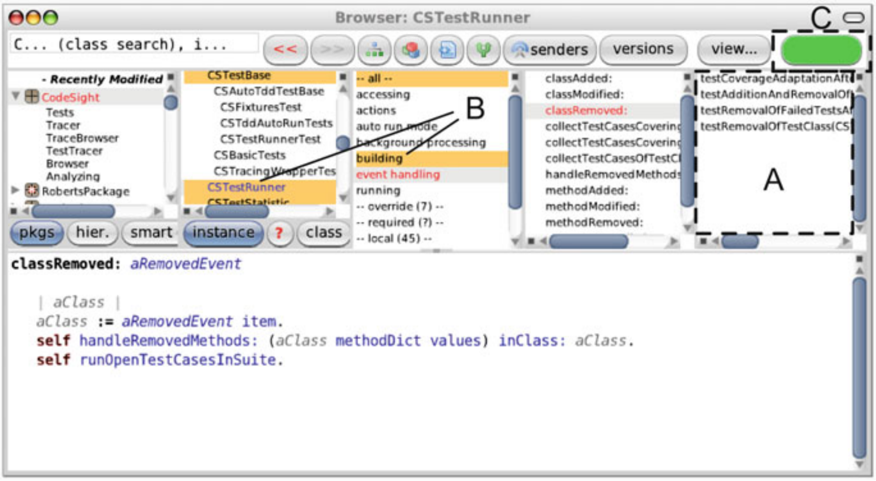


Fig. 3 An extended code browser in Squeak; having an additional panel on the *right* (**a**) that shows test cases covering the selected method named *classRemoved:*. Uncovered classes and methods are *highlighted* (**b**). A new widget (**c**) informs the developer on the current status of the test runner; whether it is currently running tests, and about the number of tests that have failed

next subsection. The tests in the queue, which failed before, are run first, providing earlier feedback on whether the current modification makes the failed test(s) pass.

To provide feedback on the test runs, we extended the tools for browsing and editing code. Whenever a modification is reported and the test runner executes tests, a newly introduced GUI widget will inform the developer about the test runner's activities and the current status of the test result (Fig. 3). The widget turns red as soon as one test has failed. Tests are executed in a background process allowing the developer to navigate to the next code entity of interest and start editing it.

2.3.4 Re-executing Selected Tests for OO Software

The set of tests to be re-executed for an applied change should be minimized. CST relies on collecting test coverage information, and using this information to select tests that might be affected by a modification.

Using this coverage information of previous test runs, the CST tools can determine the set of tests that is to be re-executed for any reported change. Selecting affected test cases is a two-step procedure:

1. If a non-test method is modified, the test runner collects and re-executes all test cases that covered this method previously. Therefore, the test runner can simply navigate the coverage relationship between the corresponding method objects.
2. CST also deals with modifications such as adding a method or changing the superclass that might affect late-bound method invocations. When, for example,

an application method *m'* is added to a class *c'*, and *m'* overrides a method *m* in a superclass *c*, CST will execute tests that have covered *m'*. More precisely, it will select those tests that previously exercised *m* for instances of *c*.

As mentioned above, the set of meaningful events, which report modified code entities, may vary between languages providing different sets of features. The algorithms to be applied to determine a safe set of tests may vary as well. If the language supports multiple inheritance, for example, the algorithms have to consider the possibility of multiple superclasses and the respective linearization order applied to method dispatch.

As pointed out in [15], a safe test selection technique for object-oriented software must also consider exception handling. CST allows considering exceptions similarly to other code entities. A basic method constructing an exception object needs to be instrumented; for instance, default constructors in Java, or *basicNew* in Smalltalk. Using the receiver's dynamic type recorded for each method call, we can determine whether an exception was created and thrown during the execution of a test case. If the exception class hierarchy is changed, all test cases that might be affected can be identified easily.

2.3.5 Establishing a Coverage Relationship

Test coverage information used for test selection is collected during regular test execution. We decided to collect this information only for packages and classes of interest. This typically excludes basic development classes such as the collection or system libraries. The selection of relevant packages and exclusion of others avoids unnecessary overhead [14]. To record method coverage information, we use method wrappers [7]. Actual method code is wrapped in tracing code that records the call of the wrapped method in the context of the currently running test case, and forwards the sent message to the wrapped method afterwards.

Test coverage information is integrated into the IDE's program representation. In CST, we establish and maintain a coverage relationship between test case methods and methods covered during test execution, as depicted in Fig. 4. Here,

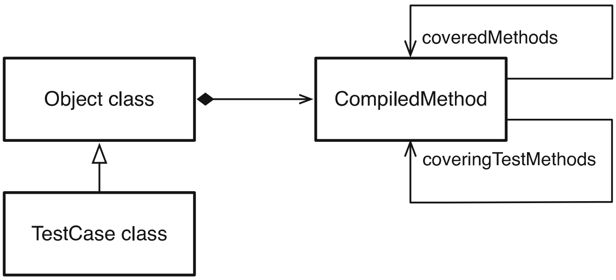


Fig. 4 The coverage relationship between test methods, included in *TestCase* classes, and application methods covered by them

we generally refer to objects representing methods in the IDE; Squeak Smalltalk provides so-called *CompiledMethod* objects to reflect upon and work with methods in the system.

Employing the test-first principle and using CST, tests run frequently and the coverage relationship has to be maintained for test runs. To avoid unnecessary start-up costs, tracing logic is installed incrementally after each compilation step. When the developer selects packages and classes of interest, wrapper logic is initially installed. If source code entities matching the selection criteria are added, they are wrapped directly after creation. This incremental approach avoids the need to instrument source code for each test run.

Using CST, developers can also be provided with instant feedback regarding test coverage. Classes and methods that are not covered any more are highlighted in the code browser (Fig. 3). The feedback supports developers in ensuring high method coverage. We further extended the code browser with an additional fifth panel (Fig. 3) that shows all test cases covering the currently selected method. This extension makes the coverage relationship visible and the applied test selection technique transparent for developers.

3 Providing Examples to Support Learning the Abstract

Visualizations of actual run-time data support program comprehension, like examples support the explanation of abstract concepts and principles. Unfortunately, the required run-time analysis is often associated with an inconvenient overhead that renders current tools impractical for frequent use.

We describe our interactive approach to collect and present run-time data. An initial shallow analysis provides for immediate access to visualizations of run-time information. As users explore this information, it is incrementally refined on-demand. We present an implementation that realizes our proposed approach and enables developers to instantly explore run-time behavior of selected code entities. Our empirical evaluation shows that run-time data for an initial overview can be collected in less than 300 ms for 95% of cases.

3.1 Motivation

Developers of object-oriented software systems spend a significant amount of time on program comprehension [4, 9, 20]. They require an in-depth understanding of the code base that they work on; ranging from the intended use of an interface to the collaboration of objects, and the effect of a method's activation during this collaboration. Gaining an understanding of a program by reading source code alone is difficult as it is inherently abstract.

The visualization of run-time information supports program comprehension as it reports on the effects of source code and thus helps in understanding it. At run-time, the abstract becomes concrete: variables refer to concrete objects and messages get bound to concrete methods. For example, profilers and debuggers support run-time exploration to answer questions such as: “What is the value of a particular method argument?” or “How does the value of a variable change?”

Unfortunately, the overhead imposed by current tools renders them impractical for frequent use. We argue that this is mainly due to two issues: (a) Setting up an analysis tool usually requires a significant configuration effort, as well as a context switch, (b) performing the required in-depth analysis is time-consuming. Both issues inhibit immediacy and thus discourage developers from using these tools frequently.

We argue that the overhead imposed by current approaches to dynamic analysis is uncalled-for and that immediate accessibility of run-time information is beneficial to program developers. Continuous and effortless access to run-time views on source code supports developers in acquiring and evaluating their understanding. Run-time views are based on actual data. Thus, they arguably encourage the evaluation of assumptions and eliminate space for speculation.

We employ a new approach to dynamic analysis enabling a feeling of immediacy missing from current tools. The central contributions of this work are:

- A novel approach to dynamic analysis based on a shallow analysis and detached in-depth on-demand refinements,
- A realization of this approach by providing an integrated tool for accessing run-time information during program development,
- Empirical results to evaluate our claims with respect to feasibility.

We will first highlight the benefits of dynamic views for program comprehension and discuss desired tool characteristics. Afterwards, we present our interactive approach to dynamic analysis that collects data exactly when needed.[3]

3.2 Background

Due to its abstract nature, source code provides a limited perspective on software systems. Conversely, dynamic views support program comprehension as they aid developers in understanding how a system works. In this section, we illustrate this by means of a running example. We continue by discussing requirements that visualization tools should meet to encourage their frequent adoption in practice.

[3]The empirical evaluation of this approach is described in the original paper [22].

3.2.1 Exploring a Program's Run-Time

Visualized run-time information helps developers to better understand program behavior. In our running example, a developer faces the task of understanding a simple clock application, which provides an analog and digital view. Figure 5 shows the structure of the application that is based on the Observer design pattern [12]. The `ClockTimer` subject represents a ticking clock, whose instances either of the two concrete observers can display. Each `tick` invocation notifies the observers about the change of state.

The developer in our example is unaware of these internals, but can use visualized run-time information to learn about them, and to eventually discover the Observer usage. This process could look as follows.

The visualized information in Fig. 6 primarily consists of a call tree that reflects a particular run of the application. A call tree provides comprehensive information

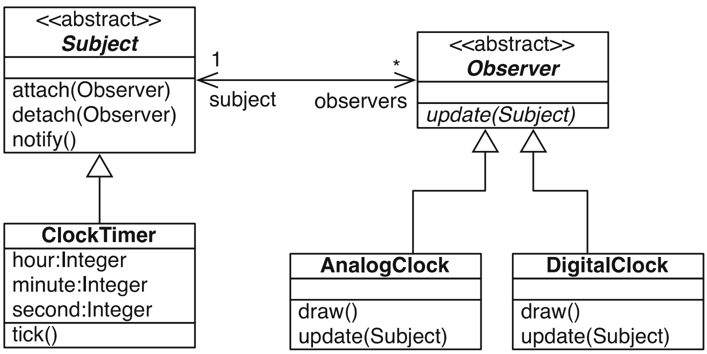


Fig. 5 Observer pattern running example

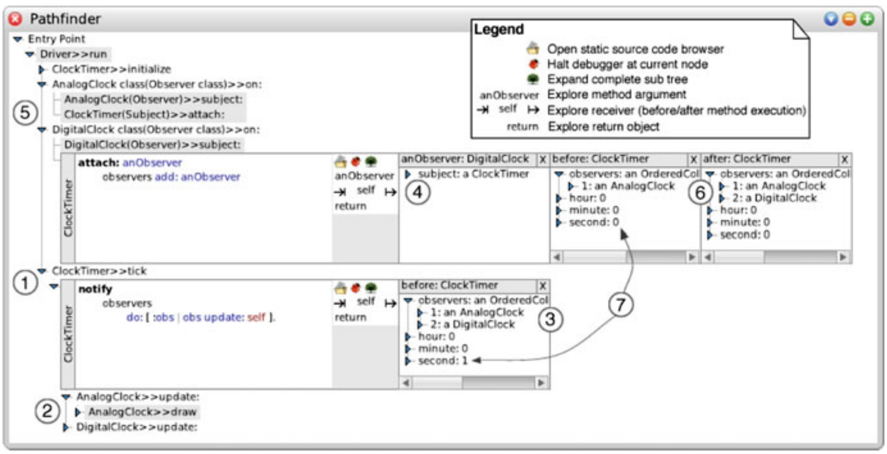


Fig. 6 Pathfinder is our interactive dynamic analysis tool for the Squeak IDE

of the entire program execution rather than a single execution path. Some of the tree nodes have been expanded to reveal details: for instance, it is evident that `tick` invokes `notify` (at index 1).

The figure shows (at index 2) that `notify` sends the `update:` message to two different clocks. From this information, the developer can conclude that there exist two observers, and ascertain this by inspecting the run-time state information attached to the execution of `notify`. The object explorer view at index 3 confirms that the `observers` list contains two clock objects. Moreover, index 4 highlights that a `ClockTimer` participates as the subject in the Observer pattern.

The provided run-time view helps to answer follow-up questions. For instance at index 5, the developer speculates that `attach:` is responsible for registering observers. In an expanded `attach:` invocation, at index 6, the combined *before* and *after* views of a method node execution show how a `ClockTimer` registers a `DigitalClock` observer. As another example, index 7 marks two views that show how the state of the subject changes after a `tick` execution. If interested, the developer could now further examine the implementation of that method to continue exploring.

In a nutshell, the developer is able to identify the conceptual structure of the Observer pattern as part of the application. In addition to comprehending structural aspects, the developer also gains deep insight about the interactions of structural entities at run-time.

Visualized run-time information sensibly augments the information available from static views on applications, e.g. their source code. For instance, the authors of the *Gang of Four* book on design patterns [12] aid comprehension of their examples in readers by presenting sequence diagrams alongside class diagrams to visualize collaborations among objects.

Visualizations of run-time data make the mental model readily available and obviate its manual elaboration. There exist valuable approaches to building mental models of software systems from static representations. IDEs support developers in navigating a code base, for example by tracing message sends, in order to understand how a system works. However, visualizations such as call trees put application source code and structure into meaningful behavioral contexts, and object explorers provide actual examples of objects rather than their abstract names.

3.2.2 The Need for Immediacy

Tools providing such visualizations of run-time data should allow for a feeling of immediacy to encourage frequent use. To that effect, two characteristics should be met. Firstly, visualization tools have to be integral parts of the programming environment. Developers would welcome a tool carrying them from method source code to the visualization of an actual run of the same method by means of one click. Secondly, response times have to be low. Visualized run-time information has to be available within some hundreds of milliseconds rather than minutes [28]. However, immediacy must not hamper the level of visual detail.

We intend to support program comprehension by reducing the effort of accessing run-time information. We aim to encourage developers to use our tools frequently. Developers shall be able to avoid guesswork and validate assumptions by inspecting actual run-time information instead. The main question that our work addresses is how to make dynamic analysis results available to developers immediately.

3.2.3 Immediacy Through Interactivity

Our interactive approach to dynamic analysis enables immediacy. Traditional approaches are time-consuming as they capture comprehensive information about the entire execution up-front. Low costs can be achieved by structuring program analysis according to user interaction. More specifically, user interaction allows for dividing the analysis into multiple steps: A high-level analysis followed by on-demand refinements. This distinction reduces the overhead to provide visualizations of run-time information while preserving instantaneous access to detailed information.

3.2.4 Step-Wise Run-Time Analysis

Splitting the analysis of a program's run-time over multiple runs is meaningful because developers typically follow a systematic approach to understand program behavior. For example, in our scenario (Sect. 3.2.1), the developer first uses the presented call tree to gain an initial understanding (1). Later on, the developer identifies execution paths that lead to the population of the list of observers by inspecting relevant state (2). More generally, program comprehension is often tackled by exploring an overview of all run-time information and continuing to inspect details.

This systematic approach to program comprehension guides our approach to dynamic analysis: Run-time data is captured when needed. (1) A first *shallow analysis* focuses on the information that is required for presenting an overview of a program run. For example, method and receiver names are sufficient to render a call graph as presented in Sect. 3.2.1. Further information about method arguments or instance variables are not recorded. (2) As the user identifies relevant details, they are recorded on-demand in additional *refinement analysis* runs. In our example, the developer clicks on the `observers` variable to see registered clocks. Information about instances contained in the list are recorded in a separate run triggered by user interaction.

This interactive approach to dynamic analysis requires the ability to reproduce arbitrary points in a program execution. In order to refine run-time information in additional runs, we assume the existence of entry points that specify deterministic program executions. For our implementation, we leverage test cases as such entry points, as they commonly satisfy this requirement [21]. However, our approach is applicable to all entry points that describe reproducible behavior.

3.2.5 Less Effort Through Step-Wise Analysis

Splitting run-time analysis and refining the results on-demand reduces the effort for providing an initial overview, as well as comprehensive details. The amount of required data for generating a run-time visualization to support an initial overview is limited compared to the information that is generated in an entire program run. The data on method activations is sufficient to render the call tree in our example. More specifically, the overhead for collecting method name and receiver information is significantly less than performing a full analysis. A full analysis includes recording exhaustive information before each state change in the execution of a program. In contrast to performing a complete analysis up-front, minimizing the collected data imposes a reduced overhead with respect to the execution of the instrumented program.

User interaction with the initial overview can be leveraged to minimize the overhead of refinement analysis. As the user expresses interest in individual objects at explicit points of the execution, required information is loaded on-demand in additional analysis steps. Such a refinement step involves recording of object state at the specified point in execution. While recording object state may be time-consuming in general, we limit the extent of data collection: a refinement step imposes a minimal overhead by focusing on a single object at a particular execution step. This means that refinement analysis is hardly more expensive than execution without instrumentation.

Our approach divides the effort for dynamic analysis across multiple runs. The information required for program comprehension is arguably a subset of what a full analysis of a program execution can provide. While our approach entails multiple runs, the additional effort is kept to a minimum, especially when compared to a full analysis that has no knowledge of which data is relevant to the user. We reduce the costs by loading information only when the user identifies interest. This provides for quick access to relevant run-time information without collecting needless data.

Our tool PathFinder (Fig. 6) realizes the described interactive approach to dynamic analysis. It is integrated into the Squeak Smalltalk IDE following our objective of achieving a feeling of immediacy. PathFinder demonstrates the feasibility of our approach.[4]

[4]A screencast is available online at http://www.hpi.uni-potsdam.de/swa/projects/pathfinder/

4 Summary

In this chapter, we have reported on two improvements that are based on key concepts in design practices. We argued that programming involves design in several respects. Developers constantly prepare the program to reduce complexity whenever possible so that future coding activities remain feasible. This gives reason for investigating the transfer of design knowledge and its application to the methods and tools for programming tasks.

First, our idea of *continuous selective testing* (CSP) and its implementation in the Squeak/Smalltalk programming environment relieves developers from manually selecting and executing tests. Based on actual modifications, a selected set of tests is executed transparently in the background, reporting instantly on the effect of the applied changes with respect to the overall set of tests to be run. Our test selection technique is based on dynamic analysis and thus does not require a statically typed language for offline processing. It is the first approach to test selection that benefits from of run-time type information to reduce test sets.

Second, our interactive approach to collect and present run-time data helps developers to understand program behavior. We argued that user interaction can be leveraged to distribute dynamic analysis across multiple runs. Our combination of dynamic analysis and user interaction reduces the effort for providing an initial overview of a program's execution. Refinement steps provide relevant details on-demand and are associated with much lower costs. With PathFinder we have shown that our approach can enable immediate access to run-time views for code entities at the push of a button.

References

1. T. Apiwattanapong, A. Orso, and M.J. Harrold. JDiff: A differencing technique and tool for object-oriented programs. Automated Software Engineering, 14(1):3–36, 2007.
2. T. Ball. On the limit of control flow analysis for regression test selection. ACM SIGSOFT Software Engineering Notes, 23(2):134–142, 1998.
3. T. Ball. The Concept of Dynamic Analysis. In ESEC/FSE, pages 216–234, 1999.
4. V.R. Basili. Evolving and packaging reading technologies. J. Syst. Software, 38(1):3–12, 1997.
5. K. Beck. Test-driven Development: By Example. Addison-Wesley Professional, 2003.
6. J. Bible, G. Rothermel, and D.S. Rosenblum. A comparative study of coarse- and fine-grained safe regression test-selection techniques. ACM Transactions on Software Engineering and Methodology, 10(2):149–183, 2001.
7. J. Brant, B. Foote, R.E. Johnson, and D. Roberts. Wrappers to the Rescue. In ECOOP, pages 396–417, 1998.
8. Y.F. Chen, D.S. Rosenblum, and K.P. Vo. TestTube: A system for selective regression testing. In Proceedings of 16th International Conference on Software Engineering, 1994, pages 211–220, 1994.
9. T.A. Corbi. Program Understanding: Challenge for the 1990s. IBM Systems Journal, 28(2):294–306, 1989.

10. N. Cross. Designerly Ways of Knowing: Design Discipline Versus Design Science. Design Issues, 17(3):49–55, 2001.
11. M. Fowler. Refactoring: improving the design of existing code. Addison-Wesley Professional, 1999.
12. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison Wesley, 1995.
13. A. Goldberg and D. Robson. Smalltalk-80: The Language and its Implementation. Addison-Wesley, 1983.
14. T. Gschwind and J. Oberleitner. Improving Dynamic Data Analysis with Aspect-Oriented Programming. In CSMR, pages 259–268, 2003.
15. M.J. Harrold, J.A. Jones, T. Li, D. Liang, A. Orso, M. Pennings, S. Sinha, S.A. Spoon, and A. Gujarathi. Regression Test Selection for Java software. In Proceedings of the 16th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications, pages 312–326. ACM New York, NY, USA, 2001.
16. M.J. Harrold and A. Orso. Retesting Software During Development and Maintenance. Frontiers of Software Maintenance, 2008. FoSM 2008., pages 99–108, 2008.
17. S. Huang, Y. Chen, J. Zhu, Z.J. Li, and H.F. Tan. An optimized change-driven regression testing selection strategy for binary Java applications. In Proceedings of the 2009 ACM symposium on Applied Computing, pages 558–565. ACM New York, NY, USA, 2009.
18. D. Ingalls, T. Kaehler, J. Maloney, S. Wallace, and A. Kay. Back to the Future: The Story of Squeak, a Practical Smalltalk Written in Itself. In OOPSLA, pages 318–326, 1997.
19. K. Beck and C. Andres. Extreme Programming Explained: Embrace Change. Addison-Wesley, 2004.
20. A.J. Ko, R. DeLine, and G. Venolia. Information Needs in Collocated Software Development Teams. In ICSE, pages 344–353, 2007.
21. G. Meszaros. XUnit Test Patterns: Refactoring Test Code. Prentice Hall, 2006.
22. M. Perscheid, B. Steinert, R. Hirschfeld, F. Geller, and M. Haupt. Immediacy through Interactivity: Online Analysis of Run-time Behavior. In 17th Working Conference on Reverse Engineering, pages 77–86, Beverly, USA, 2010. IEEE.
23. R. Hirschfeld, B. Steinert, J. Lincke. Agile software development in virtual collaboration environments. In Larry Leifer Hasso Plattner, Christoph Meinel, editor, Design Thinking – Understand, Improve, Apply, pages 197–218. Springer, 2011.
24. G. Rothermel and M.J. Harrold. Analyzing regression test selection techniques. IEEE Transactions on Software Engineering, 22(8):529–551, 1996.
25. G. Rothermel and M.J. Harrold. A safe, efficient regression test selection technique. ACM Transactions on Software Engineering and Methodology (TOSEM), 6(2):173–210, 1997.
26. G. Rothermel, M.J. Harrold, and J. Dedhia. Regression Test Selection for C++ Software. Software Testing, Verification & Reliability, 10(2):77–109, 2000.
27. D. Saff and M.D. Ernst. Reducing wasted development time via continuous testing. In ISSRE, page 281, 2003.
28. B. Schneiderman. Designing the User Interface. Addison-Wesley, 1992.
29. B. Steinert, M. Haupt, R. Krahn, and R. Hirschfeld. Continuous Selective Testing. In XP, pages 132–146, 2010.