MICRO-MODELING: A VISUAL DESIGN FRAMEWORK FOR COLLABORATIVE TOOLS IN COMPLEX SERVICE ORGANIZATIONS

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

By

Joe Bolinger

Graduate Program in Computer Science and Engineering

The Ohio State University

2011

Dissertation Committee:

Jayashree Ramanathan, Advisor
Rajiv Ramnath
Philip Smith
Paolo Sivilotti
Copyright by

Joe Bolinger

2011
ABSTRACT

Collaborative software plays an important role in firms that rely on teams to work together effectively and in pursuit of a common goal. They allow individuals to interact with one another in ways that were not previously possible, and as mediators of social interactions they have the potential to make an organization’s people-driven operations more visible and improve the self-management practices of teams. However, collaborative technologies have never fully realized their potential as tools for sustaining strategic self-management activities. Technology users tend to ignore features that are not directly related to their immediate day-to-day tasks, and they are often unwilling to deal with more complex interfaces or to input additional data into a system even if it would be beneficial from a collective organizational perspective. This research examines the often-conflicting relationship between the needs of individuals and groups in computer supported cooperative work (CSCW) systems and it addresses the lack of established design principles for resolving the tradeoffs that they create in practice. Two contributions are made.

First, a set of design guidelines is proposed for building collaborative software systems that can collect useful metadata from a managerial point of view in a way that is transparent to end-users. These guidelines help ensure that technology can support group-level needs, such as self-governance and self-monitoring, without requiring individual end-users to input data or perform actions that have no clear and immediate
value to them. These principles are first developed through a critical examination of two divergent areas in the literature and are subsequently assembled into a single software interface design pattern. Next, the pattern is used to develop a prototype system that is evaluated through a naturalistic user study. The results of this study provide evidence that user interfaces can be intentionally designed to capture strategically oriented metadata, and that it can be done in ways that are not perceived as distracting or irrelevant by individual end-users. However, the study also revealed that further research is needed to develop a related set of strategies for more effectively directing any collected metadata to people in ways that will encourage bottom-up self-management activities to occur.

Second, a novel research methodology is presented to coordinate this body of research. The presence of relevant, but often idiosyncratic, sociocultural influences makes objectively evaluating forms of collaborative technology a well-known challenge. Subsequently, heuristics and engineering guidelines are common in fields like CSCW because accurately predicting if collaborative technologies will be successful in new domains is difficult. Yet, there are no systematic methods for constructing heuristic knowledge from empirical data. The method proposed here addresses this gap by framing observational techniques from the social sciences within the structural foundation of software design patterns to facilitate an abductive, heuristic-building, research process. While this study illustrates one application of the methodology in practice, further research would be necessary to adequately compare it with traditional approaches.
To my growing family.
ACKNOWLEDGMENTS

Graduate school has been a unique experience and I am sincerely grateful to all the people that made it possible.

First and foremost, I am thankful to have had the pleasure of working with my dissertation advisor, Dr. Jay Ramanathan. Without your guidance I would not have made it this far. Your ambition is as inspiring as your enthusiasm for your work, and your insistence that I find my own “path” is what made these past few years so rewarding. Thanks Jay!

I would also like to thank my co-advisor, Dr. Rajiv Ramnath. The path through graduate school is long and you have always been there to help me locate the resources to succeed. Your attention to detail is impeccable, and I know that someday when I look back on this treatise and discover the inevitable typographical error I will remember how critical you were to my success in all matters, large and small.

A special thanks goes out to all of the talented students that I have had the opportunity to work with at the Center for Enterprise Transformation and Innovation and the Center for Resilience. Through various projects I have enjoyed working with Jared Zelek, Kelly Yackovich, Greg Horvath, Tom Lynch, Shweta Deshpande, Oleg Mishchenko, Shweta Singh, Emrah Cimren, Kieran Sikdar, and Mike Herold (and many others less directly). Thank you, Dr. Joseph Fiksel, Dr. Marc Posner, and Dr. Bhavik
Bakshi for supporting me in the department of Industrial and Systems Engineering. In addition, I owe a huge debt of gratitude to everyone in the Department of Computer Science & Engineering for contributing to my graduate and undergraduate education.

I must also thank IBM and Paul Kontogiorgis for their support during the final years of my studies, and I would like to say thanks to all of the IBMers that I had a chance to work with at Almaden Services Research. My mentor, Dr. Yolanda Rankin, was extremely supportive and her leadership made my time there unforgettable. I would also like to thank Dr. Jeanette Blomberg, Dr. Melissa Cefkin, Dr. Ben Shaw, Dr. Susan Stucky, and my fellow intern, Aqueasha Martin.

I wish to thank the additional members of my committee, Dr. Phil Smith and Dr. Paul Sivilotti, for their support. You are both excellent teachers, and I am grateful that I had the opportunity to learn from both of you inside and outside of the classroom.

Finally, I would like to thank all of my friends and family\(^\dagger\). My parents have supported me in every way possible and I could not have done this without their love and support – thank you. Isabel, you have been with me throughout this entire experience, first as a friend and now as my wife. Without your emotional support and constant encouragement I would never have survived such a long journey, and I may have forgotten why I set out on this expedition so many years ago. Thank you for everything and thank you for joining me on the journey that we are about to begin.

\[^\dagger\] Including our dog Lola – thanks for studying with me on all those long nights!
VITA

2005.................................................................B.S. Computer Science and Engineering,  
                           The Ohio State University  
2010.................................................................M.S. Computer Science and Engineering,  
                           The Ohio State University  
2005 to present ........................................Graduate Teaching/Research Assistant,  
                           Department of Computer Science and Engineering, The Ohio State University  
2010 ...............................................................IBM Ph.D. Fellow and Research Intern,  
                           Almaden Services Research, IBM Research

PUBLICATIONS

invisible visible within IT service delivery organizations. Proceedings of the SRII  
Global Conference 2011, San Jose, CA. 685-692.

client interactions in the context of service delivery. Proceedings of the 2011 Annual  

theory: An approach for creating integrative industry case studies in the software
FIELDS OF STUDY

Major Field: Computer Science and Engineering
TABLE OF CONTENTS

Abstract ................................................................................................................................... ii
Dedication .............................................................................................................................. iv
Acknowledgments ............................................................................................................... v
Vita .......................................................................................................................................... vii
Table of contents ............................................................................................................... ix
List of tables ...................................................................................................................... xiv
List of figures ..................................................................................................................... xv

Chapter 1: Introduction .................................................................................................. 1
  1.1 Contributions ....................................................................................................... 4
  1.2 Document organization ....................................................................................... 5

Chapter 2: Background & related work ...................................................................... 8
  2.1 Strategy & tactics ................................................................................................ 9
    2.1.1 User-centeredness ....................................................................................... 11
    2.1.2 Treating people as data-points ................................................................. 14
    2.1.3 Towards resolution-centered design ....................................................... 17
5.4.2 Remote observation ................................................................. 122
5.4.3 Individual exit interview and exit survey ................................. 123
5.4.4 Group validation session .......................................................... 124
5.5 Case analysis ............................................................................... 124
  5.5.1 Case A ................................................................................. 125
  5.5.2 Case B ................................................................................. 133
  5.5.3 Case C ................................................................................. 134
  5.5.4 Case D ................................................................................. 142
  5.5.5 Case Z ................................................................................. 151
5.6 Cross case analysis ...................................................................... 160
  5.6.1 From substantive theory to design theory ................................ 164
5.7 Pattern revision .......................................................................... 168
  5.7.1 Problem .............................................................................. 168
  5.7.2 Forces ............................................................................... 169
  5.7.3 Structure of the solution ....................................................... 169
5.8 Summary .................................................................................... 172
Chapter 6 : Conclusions & future work ........................................ 174
References ..................................................................................... 179
Appendix A : Case study protocol .................................................. 195
LIST OF TABLES

Table 1: Case study subjects

120
LIST OF FIGURES

Figure 1: Meeting customer demands................................................................. 8
Figure 2: BlackTie - Box & arrow diagraming....................................................... 56
Figure 3: BlackTie – Activity ............................................................................. 57
Figure 4: BlackTie - Work product................................................................. 58
Figure 5: BlackTie - Rule ............................................................................... 60
Figure 6: BlackTie - Activity details ................................................................. 61
Figure 7: BlackTie - Work product details ........................................................ 61
Figure 8: BlackTie - Distributed model at time T1............................................. 63
Figure 9: BlackTie - Distributed model at time T2............................................. 64
Figure 10: BlackTie - Distributed model at time T3......................................... 65
Figure 11: BlackTie - Distributed model at time T4......................................... 66
Figure 12: BlackTie - Distributed model at time T5......................................... 67
Figure 13: BlackTie - Distributed model at time T6......................................... 68
Figure 14: BlackTie - Distributed model at time T7......................................... 70
Figure 15: BlackTie - Distributed model at time T8......................................... 70
Figure 16: BlackTie - People bar ................................................................... 71
Figure 17: BlackTie – Isomorphism ................................................................. 73
Figure 18: The role of grounded theory........................................................... 89
Figure 19: Prototypes for theoretical sampling ........................................................... 96
Figure 20: The pattern-based methodology .............................................................. 99
Figure 21: Visual clustering to aid data collection .................................................... 102
Figure 22: Case A - Personal view accesses ............................................................ 126
Figure 23: Case A - Other's view accesses ............................................................... 127
Figure 24: Case A - Leader's view ........................................................................... 131
Figure 25: Case A - Another person's view ............................................................. 132
Figure 26: Case C – Personal view accesses ............................................................ 136
Figure 27: Case C - Other's view accesses ............................................................... 136
Figure 28: Case C - Student C's view of the process .............................................. 141
Figure 29: Case C - Student A's view of the process ............................................... 142
Figure 30: Case D - Personal view accesses ............................................................. 147
Figure 31: Case D - Student E's view ....................................................................... 148
Figure 32: Case Z - Personal view accesses ............................................................. 157
Figure 33: Case Z - Other's view accesses ............................................................... 158
Figure 34: Case Z - Student C's view ....................................................................... 159
CHAPTER 1: INTRODUCTION

The proliferation of technology has pushed the services sector into a dominant position in many industrialized economies (Heskett, 1987; Maglio, Srinivasan, & Kreulen, 2006; Murrell, Berg, & Einspruch, 2008; Ramanathan & Ramnath, 2009). Despite the shift, complex service organizations often struggle to operate efficiently and transparently. Complex service providers are those that handle non-routine and variable types of customer requests. These include design firms, consultancy practices, and factions within larger organizations that support primary business processes, such as information technology (IT) service providers.

Optimization is challenging in complex service organizations because the operational components that are responsible for delivering the firm’s services are not cogs or other forms of technology. They are humans working together as part of a distributed socio-technical system (Ramanathan & Ramnath, 2009; Vicente, 1999). Highly skilled personnel negotiate the unique requirements of each customer and carry out a coordinated response in order to co-create value (Prahalad & Ramaswamy, 2000). This level of variability can interfere with rigid business processes and has caused complex service organizations to rely more on dynamic collaboration practices than static operational plans (Kogan & Muller, 2006). Consequently, the empirical engineering and
optimization methods that have historically stretched the boundaries of manufacturing and agricultural yields have not been as successful in the complex service domain.

Systematic management and organizational engineering requires visibility and transparency. Nonetheless, knowledge about the kinds of interactions that occur as part of a firm’s operations, what purposes they serve, and what actors are involved is naturally distributed and constantly changing (Bolinger, Martin, & Rankin, 2011; Krackhardt & Hanson, 1993). Global consolidation of such information is challenging because individuals play distinct roles within organizations and their operations. Visibility beyond the horizon of a single individual’s own work is limited by fundamental physical and cognitive constraints (Allen, 1977) and is further restricted by the social norms and networked work practices of a highly educated workforce.

Excessive transparency and governance can deter individuals from being innovative in their work practices and create an atmosphere of unwelcome accountability (Suchman, 1995). However inconvenient and somewhat paradoxically, the basic principles of management are no different for networked service organizations than they are for other kinds of firms. One cannot manage what one cannot measure. But what aspects of social interaction should be made visible and measureable? Who should have the right to see it, and how should such operational information be disseminated to facilitate efficient, decentralized, and self-managed work practices?

In complex service organizations, collaborative software can play a critical role in enabling a coordinated and cost effective response to non-routine requests for service.
Collaborative systems allow individuals to keep in touch with colleagues, share information and raw data, and articulate real-time plans despite temporal and geographical constraints. However, in this role these systems do so much more than shuffle text and media through time and space. They become integral components of an organization, and they reshape the way people interact and the way that a firm is capable of responding to its customers (Olson & Olson, 1991). This role also places them in a unique position to instrument a firm. They have the potential to make an organization’s partially invisible network of interpersonal operations more visible. How this potential can be leveraged in collaborative systems effectively, or if it should be at all, has been a subject of much debate (Suchman, 1994; Winograd, 1994).

Designers have struggled to realize collaborative technologies that can simultaneously meet the tactical needs of end-users and the strategic needs of management. From a tactical perspective, workers need tools that will help them carry out their duties, automate dull tasks, and enhance their natural abilities. On the other hand, the presence of technology as a mediator of human collaboration yields strategic opportunities for data-driven management at all levels of an organization. Instrumenting interpersonal interactions, or simply making them more visible, and collecting data about them can be used for planning, improvement, early problem detection, and risk avoidance (Grinter, 2000; Haeckel & Nolan, 1993; Haeckel, 2003; Krackhardt & Hanson, 1993; Ramanathan & Ramnath, 2009). These “dual” roles that collaborative technology plays within an organization are well recognized, but they continue to confound one another in
practice. Their contradictory nature is often cited as a possible cause when collaborative technologies fail to meet the expectations of stakeholders (Bannon, 1995; Bowers, Button, & Sharrock, 1995; Dourish, 2001; Edwards, Bellotti, Dey, & Newman, 2003; Grudin, 1988; Grudin & Palen, 1995).

Understanding this dual nature of technology is particularly relevant in complex service organizations that have a need to decentralize for agility. Their networked environment diminishes the stereotypical distinction between managerial and technical roles and necessitates technology than can meet the dual needs of a skilled, networked, and self-managing workforce (Prahalad & Ramaswamy, 2000). The lack of successful and reusable design strategies for integrating the strategic need for visibility and the tactical need for usability is a practical deficiency in complex service organizations.

1.1 Contributions

This research examines the often conflicting relationship between strategic and tactical needs in computer supported cooperative work (CSCW) systems and addresses the lack of established design principles for resolving them in areas of practice. Two contributions are made.

First, a set of design guidelines is proposed for building collaborative tools that meet the immediate needs of end-users in a way that also supports their team’s strategic need for self-governance. These principles are first developed through a critical examination of two divergent areas in the literature and are eventually assembled into a single design
pattern. Next, the pattern is used to develop a prototype that is evaluated through a naturalistic user study and refined to reflect the lessons learned through that study.

Second, a novel research methodology is developed to coordinate the study and to evaluate the proposed pattern. Developing a new methodology was necessary because evaluating forms of collaborative technology objectively is a known challenge (Grudin, 1988). It is more complex than studying software from a single user’s perspective because the idiosyncratic differences in organizational cultures can influence social processes, including management and governance practices, which makes any kind of evaluation difficult to interpret out of context. In fact, the lack of research methods that specifically address this area of weakness is itself largely responsible for the shortage of reusable design principles for successful cooperative work systems. Heuristic forms of knowledge are a necessary component of design research because formal theory that transcends cultures is very difficult to develop. Yet, there are no systematic methods for developing heuristics. The method proposed here addresses this gap by framing observational techniques from the social sciences within the structural foundation of software design patterns to facilitate an abductive heuristic-building research process.

1.2 Document organization

Chapter 2 begins with a brief overview of social and collaborative software, defines some basic terminology, and discusses related work from two key areas of the literature. During this discussion an area of ideological divergence is identified and used to examine
the relationship between the tactical and strategic needs of organizations and what collaborative tools can support.

Then in Chapter 3, a conceptual micro-modeling framework is developed from an analysis of the divergence in the literature. It is a concrete strategy, or a design pattern, that can be used to incorporate macroscopic organizational needs in collaborative tools without neglecting the immediate needs of end-users. The thesis of this framework is that designing for organizational accountability and operational transparency is not as incongruent with these immediate needs as the literature has traditionally implied. In other words, it is a strategy for building tools that support group processes, such as self-governance, in ways that also support the needs of individual users. It minimizes the amount of extra work that a single user must do in order for the group to benefit. Finally, a prototype system based on the pattern is introduced, which becomes the basis for a naturalistic evaluation in the subsequent chapters.

Next, a hybrid research methodology is developed in Chapter 4 by integrating a form of the grounded theory method with the design pattern technique. The rationale for such a hybrid methodology is to lessen the gap between formative pilot study evaluation methods and more summative forms of evaluations, such as large-scale ethnographic field studies and well-controlled laboratory experiments. Its goal is to facilitate incremental heuristic development through a disciplined focus on the dominant social processes that emerge during pilot studies, whether or not they were anticipated. This is in sharp contrast to the more commonly accepted use of prototypes as vehicles for
confirming *a priori* hypotheses and patterns of behavior. This kind of traditional approach often fails to establish a credible level of validity because of the natural diversity of organizational cultures and a corresponding lack of macro-sociological theory that could be used to account for it.

Chapter 5 describes the results of a pilot study that follows the research methodology that was developed in Chapter 4. The data collected is discussed and a substantive theory is developed that explains the observed behavior throughout the study. Next, this newly developed theory is related to the original version of the design pattern and used to refine it by integrating it with the lessons learned. Finally, Chapter 6 summarizes this body of research and highlights unresolved issues and avenues for future work.
CHAPTER 2: BACKGROUND & RELATED WORK

Groupware is a general term for software that allows people to interact with one another through some kind of computer-mediated fashion. In its most basic forms it serves as a channel that enables individuals to communicate in ways that were not previously possible. However, in an organizational context it restructures the nature of collaboration, and successful groupware systems enable teams to work together and achieve shared goals (Ellis, Gibbs, & Rein, 1991). In complex service delivery organizations, these goals contribute to a single high-level and universal goal (Teboul, 2006) that is illustrated below in Figure 1.

![Figure 1: Meeting customer demands](image)

Negotiating, analyzing, and meeting unique customer demands is what differentiates complex service providers from other forms of organizations. As Figure 1 suggests,
direct interactions with customers, their expectations, and their perceptions of the service experience are ultimately what determines the performance of the organization (Parasuraman, Zeithaml, & Berry, 1985). Although this process begins on the “front stage” with direct customer interaction, work is often broken down and distributed along the “back stage” and out of sight of the customer. As more and more hand-offs occur complete transparency is lost and effective collaboration becomes necessary to deliver the service with respect to a customer’s unique expectations.

Groupware is valuable to these types of enterprises because it has the potential to enhance this collaborative process of value co-creation. For instance, it becomes a part of the embedded social processes of articulating and negotiating plans, monitoring outcomes, maintaining various types of short and long-term awareness, and establishing forms of common ground that make communication more effective (Carroll, Rosson, Farooq, & Xiao, 2009; Convertino, Mentis, Rosson, Slavkovic, & Carroll, 2009; Strauss, 1988). To emphasize this role in an organizational context, the term collaborative tool is used throughout this discussion (instead of more generic terms like groupware).

2.1 Strategy & tactics

The relationship between a coherent strategy and operational tactics is essential to service organizations. A strategic direction is developed, monitored, and reshaped over the long-term, while tactical action unfolds in the field and contributes to the overall strategy from a day-to-day perspective. A general rule of thumb suggests that strategies should be clear and measureable to the extent possible, and that tactics should remain
necessary and efficient or be subject to change or elimination (Kaplan & Norton, 1992). This distinction between the strategic and the tactical is quite relevant to the design of collaborative technologies, but its significance has received little explicit attention.

Collaborative tools mediate interactions between groups of people. Group interactions, taken as a whole, are more strategically oriented in comparison to the tactical actions of individuals. Optimizing technology from the perspective of an individual is not necessarily the same as optimizing for the group, and this causes multi-user technologies to play dual roles within organizational boundaries (Grudin, 1988; Grudin & Palen, 1995). On one hand, they must support the needs of individual “actors.” On the other hand, actors have personal, social, and political ties that connect one another and often lead to power struggles and the formation of alliances. This duality leads to conflicts and tradeoffs that must be addressed by designers if they are to develop successful tools for organizations.

The next few subsections briefly introduce two dominant attitudes that have influenced the design of cooperative systems. They are interesting because they represent two distinct and polarized perspectives on the dual nature of technology. The first is clearly more concerned with the needs of individuals, while the other has tended to take a “systems” perspective and focused on the strategic potential of technology with less concern for the day-to-day consequences of design. In the remainder of the chapter, a more in-depth analysis of trends in collaborative systems is provided, and the failure of
either school of thought to produce designs that achieve a proper balance between strategic and tactical concerns is exposed.

2.1.1 User-centeredness

User-centered design is a popular theme that has had a large influence on modern engineering practices and methodologies. It has proven itself to be quite successful, particularly in the context of tangible products and single-user software systems (Olson & Olson, 1991). Its fundamental tenet is that the users of technology, their needs, and their preferences, should be the primary focus of attention during the design process because they will experience the direct consequences of all decisions that are made. Hence, designers should cater to users’ limitations and desires, avoid making assumptions about their needs or wants, take care to account for and observe localized preferences and the surrounding culture, and create artifacts that are simple and enjoyable to use from the perspective of the user (which may not appear simple or enjoyable to anyone else) (D. Norman, 1993).

However, the user-centered moniker is misleading, particularly in the context of collaborative multi-user software systems. As social creatures a user’s needs are often related to other users’ needs, and these may eventually cascade into even larger community or regional needs. There is no reason to assume that these needs will overlap, and it is not difficult to find scenarios in which they clearly do not. For example, in a typical organizational hierarchy those with managerial roles make sure that things get done, and they are often evaluated on their ability to produce deliverable results. They do
not necessarily care how things are done, that is, until they cause problems (i.e. low morale, turnover, lawsuits, ethical problems, poor performance, etc.). A user-centered philosophy is necessarily a culturally-centered one as well. However, its rhetoric suggests that the immediate needs of the user should be given preference when they conflict with larger communal goals.

Fortunately, it is rarely interpreted quite so literally. In most cases it is taken for granted that the set of stakeholders in a new piece of technology is potentially larger than the set of end-users (Bowers et al., 1995; DeLone & McLean, 1992; Delone & McLean, 2003; Dourish, 2001; Grudin, 1994; Haeckel & Nolan, 1993; Haeckel, 2003; Hammer & Champy, 2006; Hatch, 2006; Kaptelinin & Nardi, 2006b; Selznick, 1943; Woods & Hollnagel, 2006). For example, design methodologies rooted in activity theory (Kaptelinin & Nardi, 2006a) are explicit about identifying key social constructs, like the division of labor, and the ensuing constraints that they place on design. Similarly, design in the participatory or Scandinavian tradition is built on the idea of involving the most complete set of stakeholders in as many design activities as possible (Kensing & Blomberg, 1998; Muller & Kuhn, 1993).

Nonetheless, there is little portable guidance on how to resolve the potentially conflicting immediate needs of users with their indirect culturally oriented needs. In comparison, there are many widely reusable examples of how to avoid common user interface issues (Preece, Rogers, & Sharp, 2001; Shneiderman, 1997). Similarly, there is a growing collection of generic computer-mediated interaction patterns (Schummer &
Together these provide generic solutions for problems such as supporting immediate situational awareness, enabling pointing or navigating in shared informational spaces, and dealing with synchronous and asynchronous styles of communication. But again, the majority of these strategies reveal much more about how to design for the tactical needs of individuals and their short-term tasks than they reveal about how to sustain the longer-term mission of a group. More importantly, they say very little about how to resolve individual and group conflicts when they arise.

Perhaps researchers have identified a practical reason for focusing more attention on the needs of individual end-users rather than the collective needs of groups. The assumption of altruism has no place in good design. This notion has been supported by a number of studies and is more colloquially known as Grudin’s law (DeLone & McLean, 1992; Delone & McLean, 2003; Grudin, 1988; Grudin & Palen, 1995):

> When those who benefit [from technology] are not those who do the work, then the technology is likely to fail, or at least be subverted. (Norman, 1993, p. 113)

It is quite an unsettling statement. Of course users should benefit from technology, but they are often not the only stakeholders. How can technology be created to the benefit of a larger society or organization without burdening, irritating, or otherwise marginalizing those individuals that actually “do the work?” Work is done for a reason, and that reason should not be any less important than any single individual’s immediate reasons for doing their part.
2.1.2 Treating people as data-points

Complex service delivery organizations need to be adaptive to survive. Competition has led them to rely on highly skilled workers that operate in decentralized and networked structures (Blackler, 1995; Haeckel, 2003; Hammer & Champy, 2006; McAfee, 2006). These individuals are increasingly expected to be skilled in an area of expertise, but also reflective, innovative, and capable of adapting their local practices to keep the organization running smoothly. This decentralized structure reduces visibility and makes the operations of the organization much like a black box. Even the most motivated individuals are limited in their ability to see beyond the horizons of their local practices and make globally optimal decisions (Bolinger et al., 2011; Martin, Rankin, & Bolinger, 2011).

Consequently, an alternative school of thought in the research literature has popularized the idea that collaborative tools can be used as a kind of sensor for data about the operations of people and their interactions. Data collected could potentially provide complex service organizations with leading indicators that could detect operational problems and help mitigate growing risks (Aalst et al., 2003; Aalst, Hofstede, & Weske, 2003; Aalst, Weijters, & Maruster, 2004; Haeckel & Nolan, 1993; Haeckel, 2003; McCoy, Schulte, Buylendijk, Rayner, & Tiedrich, 2001). This data could be electronically processed and routed quickly through the organization so that decentralized decision making is as consistent, globally informed, and data-driven as possible. In this
role, collaborative technology has the potential to make an otherwise invisible network of human interaction more visible.

As a prototypical example, consider Haeckel and Nolan’s vision of “managing by wire” (Haeckel & Nolan, 1993). They claim that information technology (IT) can provide a strategic backbone for organizations that need to adapt quickly by instrumenting the organization with very precise computer models of how a firm operates. These models could then be used to construct gauges, or even complete dashboards, that would make an organization’s day-to-day operations immediately visible for inspection. To support this claim they draw a metaphor from modern aircraft cockpits, and note that flight crews do not see the complete inner workings and complexities of the aircrafts that they command. Instead, they see simplified IT mediated abstractions of those inner workings, and they claim that because the computer has such a precise model of the plane it can provide the crew with accurate and timely information.

From the strategic perspective of management, this argument is a good fit with the notion of a decentralized and knowledge driven workforce. Management may not need complete visibility into operations (the inner workings of the plane). Instead, they can rely on human ingenuity to the degree that they can establish a reasonable amount of confidence in their indicators (the gauges and dials) to maintain a coherent strategy (the plane does not crash). However, like many metaphors it is misleading because it uses an unnatural and unequal mapping. In the case of complex service organizations, the flight crew represents only the policy makers. All the hydraulics, gears, and electro-mechanical
subsystems represent the workforce, and that typically includes many highly skilled personnel. Although people, teams, and groups can be instrumented to some degree, it is overly ambitious to assume that a computer model will somehow allow human behavior to be sensed, measured, abstracted, and electronically mediated to the degree and accuracy that a cockpit display can make the higher-level functions of an aircraft visible.

There is nothing inherently flawed with the idea of management by wire and it has also witnessed some commercial success. Still, it is as overly optimistic in spirit as the notion of user-centeredness. Collaborative technologies can collect data as individuals work together but that data is relatively superficial. For example, application or clickstream logs may be useful for detecting if a user is struggling with an interface to complete some well-defined use case. But, using that same kind of data for purposes such as determining if an organizational policy is being violated is significantly more challenging (although it has been attempted Aalst et al., 2003; Song & Aalst, 2008). In fact, Grudin’s law (given in the previous section) suggests that attempting to do so may lead to deviance or even invite subversive behavior. In situations like these individuals will find ways to conceal what they are actually doing and falsely manipulate IT mediated indicators in their favor. Or, they may find ways to work around these “sensing” systems entirely (Woods & Hollnagel, 2006).

Designers have the ability to influence the kind of data can be collected through the interfaces that they produce. However, trends like Grudin’s law suggest that the trade-offs must be examined carefully (i.e. group versus individual benefit). Instrumentation
may have the potential to increase operational transparency and facilitate management by wire as more interactions take place over the Internet or through collaborative tools. However, if people are unwilling to be treated as data-points there is little chance that such instrumentation will yield anything but distractions, noise, and irritation.

2.1.3 Towards resolution-centered design

These two competing philosophies are more representative of two extremes along a single axis than they are truly distinct. Researchers have long recognized the tradeoff between visibility, the accountability that it creates, and structuring the nature of work through technology (Abbott & Sarin, 1994; Bannon, 1995; Bowers et al., 1995; Dourish, 2001; Gerson & Star, 1986; Grudin & Palen, 1995; Herrmann, Hoffmann, Kunau, & Loser, 2002; Kueng, 2000; Suchman, 1987; Suchman, 1994; Suchman, 1995; Swenson, 1993; Winograd, 1994). On one hand, sometimes “freedom” from accountability is exactly what people need to work together successfully and to survive organizational politics (Nardi & Engeström, 1999). But, visibility is also necessary in many cases for effective collaboration and coherent decision-making.

Teboul recognized this need to balance tactical and strategic concerns in service organizations through a “power play” on the service triangle (Teboul, 2006). The service triangle refers to the three groups that are of critical important to service organizations, the firm, its employees, and its customers. Members from each of these groups must work together in order to create economic value but each group also has conflicting motives. For instance, Teboul notes that employees generally want some level of
autonomy but firms rely on budgets and norms for the purposes of efficiency and forecasting. Similarly, the technical competence of employees influences the kinds of interactions that customers and providers can engage in, but the bargaining power of customers can also shape the nature of these interactions and influence the level of training that employees may need. While Teboul’s model is specific to complex service organizations similar “power plays” are likely to exist in any form of organization (for evidence see (Kaptelinin & Nardi, 2006b)).

Organizations rely on strategic governance processes to balance the distribution of power, but it is difficult to integrate such processes in collaborative tools because governance does not directly benefit a user – to any degree. Governance is something that keeps organizational systems from falling apart, but it is not something that has any direct value to anyone beyond that purpose. How can collaborative technology platforms be designed to thrive within this kind of environment and maximize the benefits from all of these competing perspectives?

2.2 Ideological divergence in the literature

There are two important communities in the literature that have contributed to the development of collaborative tools. The first is the workflow research paradigm, which roughly embodies the “treating people as data points” perspective. Its main concern has been on the development of software for office automation through the application of formal mathematical models (Aalst & Kees, 2004). A more “user-centered” branch of research is the field of computer supported cooperative work (CSCW). It has come to
represent a focused field of study on the application of technology in organizational contexts, with the explicit goal of understanding how it can support interpersonal coordination and inform design (Grudin, 1994)².

Both of these communities have similar interests and goals, but they have remained fairly distinct from one another in both theory and practice. They represent two extremes along a spectrum of research methods. The mathematical roots of the workflow community tend to underemphasize the true complexity of social processes and their effects on technology. Similarly, the micro-sociological foundation of much CSCW research has failed to develop reusable strategies to explain how technology can be positioned into an organization beyond the “user-centered” boundary (i.e. it is easier to uncover how technology has failed in the past than it is to build theory that prevents failure in the future). Nevertheless, both traditions share a mutual frustration for transforming the results of research efforts into repeatable and successful applications in practice (Grudin, 1988).

In the following subsections, a review of each paradigm is first presented to establish points of reference. Next, a review of the literature that draws from both paradigms is discussed and, finally, a summary of some critical points concludes this chapter.

² Workflow technology somewhat predates CSCW becoming its own research field. Hence, workflow is something that CSCW researchers have studied because it is a form of collaborative software. However, this split is not artificial and there is a noticeable distinction between the workflow (i.e. process engineering) and CSCW communities.
2.2.1 The workflow paradigm

The workflow paradigm is associated with a number of closely related topics, including workflow systems, business process management (Aalst, Hofstede et al., 2003), business activity monitoring (McCoy et al., 2001), and business process re-engineering (Hammer & Champy, 2006). These topics are mostly differentiated with respect to how workflow technology is applied in practice, and not on any significant technical differences (Aalst, Hofstede et al., 2003).

The term workflow refers to software that is used to model and execute business processes, which is often based on formal models of computation like Petri Nets (Murata, 1989), logic programming (Croft, 1984), or related techniques. In their most basic form, workflow systems simply route pieces of information to users in response to events to complete a logical sequence of actions (such as sending a message to a representative when a customer places an order, rerouting that order to a warehouse when the representative finalizes it, and archiving records in a database when that order is eventually fulfilled by a distributor). Much like reusable user interface libraries have allowed programmers to decouple functional business requirements from interface code, workflow systems have helped to separate routing logic from other programmatic concerns.

The other terms denote particular methodologies for managing business processes, most often using workflow technology as the principal means. For example, business process management refers to an incremental quality improvement methodology that is
grounded in the statistically controlled Shewhart, or Deming, cycle (Deming, 1986). In comparison, business process re-engineering is a less incremental improvement methodology based on the use of well-planned, radical, and sweeping changes throughout an organization.

Other uses of workflow technology have grown out as extensions to these basic strategies in response to wide workflow adoption in industry and the increasing availability of process-oriented data (Ramanathan & Ramnath, 2009). For instance, business activity monitoring is a generic strategy for aggregating near real-time operational data and making it visible through organizational “dashboards.” Proponents claim that this kind of visibility will allow decisions to be made in less time and on the fly, which should lead to more robust and efficient business processes (i.e. management by wire).

Similarly, workflow-mining (Aalst et al., 2003; Aalst et al., 2004; Cook & Wolf, 1998) techniques have arisen in response to the vast amounts of data that these systems can log and record. This class of techniques transforms logged event streams into process-like representations that can be used as an input for organizational improvement, planning, or re-engineering activities. It also leverages the value of process-centric representations (i.e. models or traces of work) as artifacts for other forms of analysis. For instance, they can be used to help understand or reconstruct actual work practices in specific cases to carry out an audit, they can be used as aids for training, or they can be
used to verify conformance of certain behaviors for legal reasons or to demonstrate regulatory compliance (Song & Aalst, 2008).

The mechanical aspects of workflow systems are not relevant here but the concept of a process model is. It is important to distinguish the characteristic design of workflow systems from the methodologies and principles that are used to integrate them into an organization. As a basic technology, workflow is formal, precise, ubiquitous in the marketplace, and relatively uncontroversial. However as a mechanism for enhancing an organization’s effectiveness through the use of explicit process models, it has a history of controversy. For instance, Winograd (Winograd, 1994) and Suchman (Suchman, 1994) held a classic debate on the theoretical value of workflow based on Winograd’s theory of language/action and speech-acts (Winograd, 1986; Winograd & Flores, 1987).

The language/action theory attempts to explain coordination between cooperating agents though a linguistic perspective, which it does by providing a uniform model of the “structure” of coordination between agents. For instance, it recognizes the use of language for a number of cooperative purposes, including for action, clarification, possibilities (i.e. seeking to find an action to take), and orientation. The “conversation for action model,” found in (Winograd, 1986), is the theory’s primary model for describing how cooperation works.

Essentially, the model is a linguistic “protocol” that coordinating agents are assumed to follow naturally. It can be interpreted as a state diagram where speech utterances that imply action on the part of the listener, speech-acts, are represented as arrows between
the states of a complete “conversation.” Under the model, a conversation begins with an initial “request” and progresses through a series of transitions, such as another entity “promising” to meet that initial request, the request being “withdrawn,” etc., until the conversation reaches a terminal state (there are a total of nine distinct states in the original model). The model was later used as a basis for the Coordinator workflow system and the ActionWorkflow platform (Medina-Mora, Winograd, Flores, & Flores, 1992). The Coordinator system was advertised as more natural in comparison to other workflow systems that lacked such a linguistic and human-centered foundation (other systems were often based on a mechanical information-processing model rather than a linguistic one). For example, the Workflow Reference Model’s (Hollingsworth, 1995, p. 23) provides a comparable state diagram of a “process.” It serves the same functional purpose as a language/action “conversation,” but it uses very different language and embodies a much less human-centered approach (i.e. processes can “start,” “stop,” or can be “suspended” or “terminated”).

However, Suchman criticizes the linguistic interpretation of workflow and labeled the conversational model as a mathematized interpretation of reality that gave arbitrary technical meanings to otherwise “colloquial expressions.” In other words, she claims that the model is not a reflection of reality or of how people actually go about planning. Her alternative theory of situated action suggests that behavior is contextualized and cannot be framed in a universal model of negotiating plans. She argues that this artificiality is a fundamental flaw in the language/action theory, and suggests that because of it workflow
technology will fail to significantly empower organizations. Instead, she claims that it would burden them by creating additional layers of unnecessary bureaucracy and oppressive controls that result from such a synthetic and superimposed structure.

Fundamentally, her claim is that there is no uniform structure for coordination, be it grounded in a conversational protocol or otherwise. Hence, attempting to create such an artificial structure is an inevitably doomed approach. Winograd responded by citing that abstraction and artificiality is the nature of computers and noting that without some kind of formalization workflow technology could not be built, much less play a role in mediating or enhancing human collaboration. He also challenged the idea that workflow is doomed to bring about unnecessary bureaucracy by noting that organizations do, as historical fact, use artificial formalizations and bureaucratic practices as they grow to a positive effect. Thus, even if the language/action theory does not perfectly match reality, its artificiality does not render it useless as a foundation for design. It may not be an optimal model of human cooperation, but at least it is not a completely arbitrary one.

These kinds of theoretical debates of workflow technology have continued, but they have not lead to any generally accepted conclusions. No one claims to have reliable models that can explain or predict the effects that deploying workflow, or similar technologies, will have on an organization. This does not mean there is a lack of research on the topic; see (Vicente, 1999) for an integrated example of socio-technical systems analysis. However, it does imply that making workflow a widely useful technology is complex because organizations defy simplistic analytical explanation. As a result, many
have called for empirical accounts of workflow in order to understand how they may, or may not, lead to success stories in the field. Bowers noted such a need for an empirical understanding of workflow systems, and provided one of his own through an observational study of a workflow system that was introduced at a print industry (Bowers et al., 1995).

Throughout his account, Bowers noted a number of ways that the workflow system frustrated users as they carried out their assigned duties. For instance, the system imposed an unnecessary ordering of work activities that needed no sequencing, it added procedural overheads and new data entry requirements, and its control structure prevented ad-hoc practices that were entirely reasonable. Of course, the system could not physically constrain people and he noted that in order to work around some of these problems workers would sometimes step “around” the boundaries of the system to get things done. They found ways to get their own work done and they did not always use the system in ways that had been intended.

However, Bowers went on to describe how the system was also contractually mandated as part of a service agreement. The company’s customers demanded credible operational reports and there was no way to produce them without the help of this kind of technology. This conflicting scenario illustrates the most important lesson of the entire story. The sidestepping behavior was annoying to many of the workers, but without it the business was at risk (i.e. customers have power and they were demanding transparency).
Unfortunately, the system was unable to serve its dual roles as both a mediator and an accountant of work.

The inflexibility, brittleness, and general obtrusiveness in workflow has long been recognized and validated through empirical studies, like Bowers’s. However, most accounts also reveal that while parts of an organization may resist change or suffer, the structure and support provided by workflow can reduce errors, lead to quality improvements, produce more resilient business processes, increase accountability and visibility, or automate dull tasks, and all without significantly disempowering individuals. This overlap extends to such an extent that it is difficult to classify individual accounts as distinctly positive or negative (Abbott & Sarin, 1994; Bowers et al., 1995; Grinter, 2000; Herrmann & Hoffmann, 2005; Kueng, 2000; Oba, Onoda, & Komoda, 2000; Reijers & Aalst, 2005; Reijers, Song, & Jeong, 2009). As noted in (Abbott & Sarin, 1994), the historical outgrowth of workflow from more primitive technologies for brute automation has associated it with concerns over power struggles and a disenfranchised workforce, but that is more of a legacy than an accurate characterization of modern workflow. Hence, it is often associated with a partially misleading negative connotation and it has been deliberately separated from other major design trends.

In summary, workflow systems and the various methodologies for applying them in practice have been successful in many organizations. However, the inherently formal and pre-wired patterns of work that workflow systems can effectively support has left them seemingly less relevant in environments where the nature of work spans from simple to
complex, or from planned and routine to articulated and ad-hoc. Since real work is often a mix of the routine and the non-routine, workflow systems have repeatedly been shown to be useful for about half of the problem, roughly half of the time.

2.2.2 The cooperative work paradigm

Computer Supported Cooperative Work (CSCW) grew out of interdisciplinary research surrounding organizational coordination and group work. Although it has been influenced by a number of discrete disciplines, CSCW has defined itself as an area distinct from purely sociological, psychological, or technological approaches and defined itself as “a design-oriented research area (Schmidt & Bannon, 1992).” The emphasis on design provides a filter through which interdisciplinary knowledge can be applied to a common goal, ultimately leading to the construction of useful systems to coordinate work.

In setting this focus, it established “articulation work” (Strauss, 1988) as a central concept. Articulation work refers to the process of making sense of one's local environment, planning an appropriate response, and articulating that plan by coordinating, renegotiating, and monitoring its progress in order to get real work done. As a consequence, CSCW has largely disassociated itself from formalized and procedural approaches, like that of the workflow paradigm, in favor of less formal mediation strategies emphasizing shared information and virtual workspaces. By analogy, the workflow approach represents the application of technology as a means to carry out and
execute plans, while CSCW more accurately represents a philosophy of using technology as a means to articulate and negotiate such plans.

In their characterization of the field, Schmidt and Bannon point out that while process-like artifacts are useful for coordination they should not become “executable code” as they do in workflow,

...but rather heuristics and vague statements to be interpreted, instantiated, and implemented maybe even by means of intelligent improvisation. (Schmidt & Bannon, 1992, p. 26)

Here, they are pointing to a fundamental ideological difference between the two paradigms. They go on to claim that any artifacts related to planning form a “local closure” (Gerson & Star, 1986) at the time they become codified, which requires them to be reinterpreted in new contexts and not blindly reused as they might be in a workflow system. Thus, while the workflow paradigm has embraced the use of formal models of an organization, the CSCW community has responded by asserting that any such model of an organization is so far from reality that it can only be taken as a rule-of-thumb or treated as some sort of heuristic device.

This focus on artifacts as aids for coordination is why the CSCW paradigm has adopted a “shared spaces” orientation as a means for distributing such resources, in lieu of workflow’s more assertive “shared policy” perspective that directly operates on those resources. Consequently, CSCW has developed an alternative set of theories to explain the inability of human coordination to be put into an algorithmic language (Kaptelinin & Nardi, 2006) and to form a theoretical basis for cooperative system design. For example,
situated action (Suchman, 1987), distributed cognition (Hutchins, 1995), and Activity Theory are three of the commonly used logical frameworks in the CSCW literature.

Situated action and distributed cognition both emphasize the adaptive nature of humans and the influence that the local environment has on behavior. Collectively, they have motivated the use of detailed ethnographic field studies as a means of understanding the kind of idiosyncratic behavior that only seems to emerge in the context of genuine practice. For example, Malone provides such a detailed observational analysis of how office workers interact with their desks (Malone, 1983). He describes a number of individuals that used “piles” of “files” in various ways to organize their work. Focusing on some of the specific uses of the piles and files he noted that some people have difficulty conjuring up useful classification schemes for certain things, such as imminently important files. This led him to the insight that the available desktop space creates a “deferred classification” mechanism that removes the need to immediately file items away.

This principle of providing a means of deferred classification is a design principle that has seen widespread use in modern computer systems, and it is a key part of the desktop metaphor that most computer users have grown accustomed to. Nonetheless, this kind of observation without underlying theory has serious limitations. For example, critiques of the desktop metaphor are quick to point out (Freeman & Gelernter, 2007) that the translation from a physical desktop to a virtual desktop is not as straightforward as it may seem. Virtual desktops are not equivalent to their physical counterparts. They do
not necessarily yield the same functions and there may be more efficient ways to implement mechanisms, such as deferred classification. Rich observational studies are useful for seeding the creativity of designers, but they do not lay the foundations for good designs or lead to the construction of sound theory on their own.

In addition to pure observational research, the CSCW community has adopted more structured theories for explaining the role that technology plays in practice. For instance, Activity Theory (Kaptelinin & Nardi, 2006a) is a widely accepted foundation for design research. It is particularly sensitive to tools, their relationship with society, and cultural developmental processes. The focus on a communal unit, instead of the more primitive relationship between an individual and a tool, is what makes the theory a useful lens for examining the consequences of design.

Activity Theory can be summarized through a few essential concepts (Kaptelinin & Nardi, 2006b). First, an “activity” frames the conceptual unit of analysis. Activities are characterized by an “object”-orientation, which implies a corresponding motivation to act on this object (this object does not need to be physical, it is like an objective or a direction that a person wants to move in). This object-orientation frames the social, historical, and environmental context within any further understanding of behavior. For example, Engeström’s model (reproduced in Kaptelinin & Nardi, 2006a, p. 100) is an example of a concrete “activity system” based on the theory. It is a frame of reference that can be used to analyze the various elements that may interact with one another to shape group behavior. It includes a set of interrelated concerns such as “tools,” “rules,”
and the “division of labor” that influence one another to drive behavior. Most importantly, the model shows that these elements may occasionally come into conflict with one another, which necessitates change and leads to technological and sociological development and evolution of the “system” as a whole.

Unfortunately, Activity Theory is not a model that can be used in a precise predictive manner. It gives designers a lens to focus their analysis, but it does not explain how all of these interacting forces will actually influence behavior (i.e. it is a descriptive model). There are more predictive models available, such as GOMS (Card, Moran, & Newell, 1983)⁴, but they come at the cost of ignoring some of the influences that Activity Theory has recognized as relevant factors that influence social order.

In summary, the CSCW paradigm can be viewed as an intentional alternative to the technology-oriented worldview embodied by the workflow paradigm and its reliance on information-processing and algorithmic models of coordination. Perhaps it is for this difference in perspective that they have held one another in a negative light, despite their similar goal of supporting collaborative work. Nonetheless, both paradigms have had a significant influence on the design of collaborative systems in practice. The next section presents a survey of collaborative systems that have drawn on the strengths, and weaknesses, of both schools of thought.

---

⁴ GOMS (Goals Operators Methods and Selection Rules) is a human processor model that is used to break down a user’s interaction with a system into small chunks, like keystrokes or pauses, which can be quantified and used to predict an expected model of behavior. Many variants of the basic method have been developed over time.
2.2.3 Crosscutting perspectives

Articulating plans and coordinating work can be supported by computer systems at varying levels of fidelity. At one extreme, explicit plans and related artifacts can be embedded into a system to precisely “specify” the kinds of information that must be exchanged among parties to promote a similar way of working over time. An online ordering system is a good example of such a system. At the other extreme, they merely allow individuals to share certain artifacts without specifying their detailed structure or semantics (i.e. as in email). However, most collaborative software falls somewhere in between the two extremes and facilitates the exchange and maintenance of specific kinds of semi-structured artifacts. For example, systems that provide document-sharing services through a central repository or event planning on a shared calendar impose a partially standardized way of working on people, but not to the degree that would prevent them from being used by (and sold to) organizations that have different work practices.

Collaborative systems often move further toward the more “specified” end of the spectrum within particular domains. For example, programmers do not use general-purpose document sharing systems to maintain and share code. They use specialized document sharing software that has been designed to integrate with relatively common development practices, such as versioning, branching, locking/freezing, and commenting⁴. Historically, this has lead to a dual lineage of highly structured workflow-

---

⁴ For examples, see CVS, SVN, or the Git source code management systems.
like systems that are rooted in precise specifications, and less structured CSCW tools that are inspired by approximate representations of work related artifacts. Each has sought, unsuccessfully, to reach an elusive spot somewhere in the middle where technology is able to support flexible work patterns in a reliable fashion.

Bernstein characterized this diversity with the notion of a “specificity frontier” (Bernstein, 2000). The specificity frontier denotes the range of cooperative tools from those that are very structured and controlling, such as workflow and resource planning systems, to those that are very unstructured, such as email and document collaboration tools. To explain how each type of tool falls along this frontier, he stratifies the space into four categories: “providing context, monitoring constraints, planning options based on constraints, and guiding through scripts/directions” (Bernstein, 2000, p. 281). Each category represents a set of progressive features that make a tool more or less “specified.” He uses this framework to argue that because real work encompasses a range of conditions, from highly routine to highly dynamic, cooperative tools should allow moving along this frontier dynamically based on the context of the situation. For example, a workflow system that is actively guiding a user’s action should be able to gracefully degrade and take on an advice-giving role when it can no longer confidently direct that user because of unexpected circumstances or conditions.

To illustrate this concept he describes a prototype system that allows movement along this frontier. The system relies on a knowledge-based planner that can offer a user such advice when unexpected situations prevent it from making direct decisions and guiding
that user through a predefined task. Essentially, the design makes the system’s reasoning process visible to the user. Was this design successful? Unfortunately, it is difficult to say because, as with most experimental collaborative systems, researchers face significant barriers when they attempt to objectively evaluate their designs (this will be covered in more detail later in Chapter 4).

The specificity frontier is a useful characterization but it leaves many questions unanswered. Most importantly, it ignores the role that technology is playing and how that role might be changed if the specificity is actually allowed to vary. For example, Dourish notes that tools which fall on the highly specified end of the spectrum are valuable because they serve as “organizational accounting devices” (Dourish, 2001) in addition to their role as coordination aids. Dynamically changing the level of specificity may not be acceptable if automation is not the only motivation for moving into the highly specified regions along the frontier. Realistically, automation and accountability are often mutually desirable features (recall Bowers’s analysis of workflow in the print industry).

The development of experimental workflow and groupware systems that aim for a useful balance between specificity and flexibility, such as Bernstein’s, have a rich history in the literature. Examples include accounts of flexible process-oriented systems like (Bellotti, Ducheneaut, Howard, & Smith, 2003; Bernstein, 2000; Castellani & Pacull, 2002; Dourish, Holmes, MacLean, Marqvardsen, & Zbyslaw, 1996; Ellis, Kedda, & Rozenberg, 1995; Faustmann, 2000; Gottschalk, Aalst, Jansen-Vullers, & La Rosa, 2008;
Before proceeding, it is helpful to introduce a conceptual framework that can be used to frame the discussion. One such framework is Dourish’s model of the “dimensions” of workflow systems. Much like the specificity frontier, any collaborative technology can be understood within this framework on the basis of the roles and the responsibilities of the system. What makes Dourish’s framework particularly useful is that it focuses on how plans, or process-like artifacts, can be leveraged by technology in a number of distinct ways. For example, in workflow systems there is always an explicit connection between the representation of plans in the system and the expected actions of users. In
other forms of collaborative technology the system may not have this kind of end-to-end responsibility, but it may play a smaller role in the process of moving from plans to action.

Dourish’s dimensional framework\(^5\) is composed of three logical dimensions along which technology can support a work “process” (Dourish et al., 1996, p. 191). Each dimension corresponds to a set of actions that an actor will need to engage in to manipulate and make use of a logical plan. First, it must be written down in an appropriate manner so that it can be used at a later time (a). Next, some entity (human or computer) must have a mechanism for interpreting that codified model in the context of a specific situation (b). Finally, that interpretation must be somehow useful for assisting an actor in meeting an objective or completing a task (c). Point a represents the act of formalizing a particular work process, in whole or part, into a format that can be used to inform or guide work that will occur later. B represents how that explicit artifact will lend itself to interpretation and lead to a plan in the field. Finally, point c represents how authoritative an interpreted plan is or, to what degree actions can be expected to conform to the plan (i.e. if it will lead to a consistent pattern of behavior over time). For example, a very specific description may lend itself to execution with strict controls and consistent points of measurement while a looser description may not.

\(^5\) This is a broader use of Dourish’s framework but it is not contradictory to its original purpose.
Like the specificity frontier, this reference model helps to relate the workflow and cooperative work paradigms. At one extreme, to be a traditional workflow system implies that work processes are modeled to an exacting level of detail and used to direct step-by-step actions in clear and repetitive ways. The result is a detailed and contextually dependent process description, much like a computer program. It will have the potential to render a highly predictable pattern of behavior if the assumptions made during the modeling process remain unchanged (i.e. routine work).

At the other extreme, less structured collaborative systems are designed based on the assumption of a more universally relevant process model rather than a specific one. For example, many groupware systems are based upon a generic process of organizing arbitrary electronic materials into some kind of a shared repository. In this case, this underlying process can be observed through the actions of the system’s users, but it is only one part of their larger work processes. It is a perpetual cycle of adding, removing, and moving materials in to and out of that repository for reasons that are unknown to the system. As such, the model is less specific and not as dependent on as many contextual factors. It is less brittle and capable of supporting a wider variety of collaborative behaviors but it still relies on the assumption of a normative process model.

Dourish used his framework to motivate the Freeflow system, which is an example of a workflow system that has taken a step away from the rightmost extreme of the specificity frontier. In Freeflow descriptions of a process (point a) are a set of constraints on a goal rather than a step-by-step sequencing of procedures. From a representational
standpoint, the process is described in a declarative fashion rather than an imperative one. The primary advantage of a declarative representation is that it allows a process modeler (at point a) to avoid introducing artificial deterministic routes and consequently overspecifying the process without just cause. Instead, the Freeflow system monitors a set of constraints that can restrict certain kinds of actions without specifying a literal recipe-like procedure.

For more flexibility, Freeflow also allows constraints to be broken if a user explicitly indicates that they should be broken via an interactive prompt. Although these features make Freeflow somewhat more flexible than traditional workflow systems, its additional flexibility is largely a consequence of removing accidental complexities (Brooks, 1987). Work processes must still be modeled completely a priori and out of context, which is responsible for the majority of their brittleness. Similarly, allowing arbitrary constraint violations makes the system more flexible but it also lessens the ability of the system to play its dual role as an organizational accounting device, a role that Dourish himself recognized as a potentially important one.

Both Dourish’s and Bernstein’s designs changed the dynamics of workflow systems to accommodate more dynamic forms of work. However, neither addressed a gap between point a, where work is modeled a priori, and point b, where that model is interpreted for potential use in the field. Their designs are representative of a centralized, global, and bureaucratic style of management and governance. However, despite the presence of formal organizational policy, informal and locally constructed policies often
govern work practices as well (Gerson & Star, 1986; Kogan & Muller, 2006; Moran, Matthews, Vega, Smith, & Lin, 2009; Muller, Geyer, Brownholtz, & Wilcox, 2004). Collaborative systems that only provide support for implementing the high-level global policies of an organization may be limited because they do not support the kinds of localized policies that evolve to shape work practices over time. Ideally, policy should be accounted for on both a local and a global scale and these designs do not support the incremental and evolutionary nature of localized self-management practices.

Regatta, its descendant TeamWARE, (Swenson et al., 1994; Swenson, 1993) and Serendipity (Grundy & Hosking, 1998) are representative of another class of flexible approaches to workflow. Unlike the previous systems, they do target the modeling process (and the gap between points a and b) as their source of flexibility. In Regatta, the process of modeling a work process is perpetual and always subject to change. Users collaboratively model their work processes by creating diagrams through a visual interface, and the state of these processes is maintained by the system over time. This differs from traditional workflow systems because the modeling can be done by the end-users of the system rather than experts or business process analysts. In Regatta, if a user feels that the model does not suit their particular situation they have the ability to change it directly rather than having to somehow work around the inconsistency.

The semantics of a process in Regatta are similar to that of a traditional workflow system. In other words, the model is still used to script a sequence of actions. However, its design is intended to promote the formalization of work practices by giving end-users
the capability to do the modeling. It inverts the role of process design and process enactment found in traditional approaches, and it allows the system to be more flexible when unanticipated circumstances arise. However, this does lead to a new kind of tradeoff in practice.

Collaborative planning systems like Regatta place a burden on users to explicitly model their work. Maintaining a process model over time so that it remains an accurate reflection of reality that can be usefully employed in new situations requires time and effort. Users may be reluctant to engage in this extra work unless it has a clear benefit to them and their duties. Hence, this kind of approach depends on how appropriate the modeling language is for any given community of users and their ability to integrate modeling activities into their native work practices. For example, Regatta was designed to function as a standards-compliant workflow system and its visual process modeling interface requires users to model work in a very precise machine-readable manner. This can lead to over-specification (Grinter, 2000; Grudin, 1988; Herrmann et al., 2002; Kogan & Muller, 2006; Moran et al., 2009; Muller et al., 2004; Suchman, 1987) and it assumes that the work of a knowledge worker (Blackler, 1995) is easy to codify in the precise mechanical terms of traditional workflow systems, which may not always be the case.

Regatta allows end-users to model and script relatively complex sequences of actions. Other collaborative planning systems stop at the modeling phase and do not interpret or execute the artifacts that are modeled in order to reduce the level of complexity. Instead,
these systems allow users to collaboratively edit visual artifacts that are most often engineering diagrams, such as IDEF, UML, or similar domain-specific models (Dean, Orwig, & Vogel, 2000; Dennis, Hayes, & Daniels, 1999; Grundy & Hosking, 1998; Meire, Borges, & Araújo, 2007; Yu, 1995). These kinds of systems are often integrated with decision-theoretic models to support deliberative processes in a way that reduces forms of bias or incites creativity6 (Convertino, Billman, Pirolli, Massar, & Shrager, 2008; Farooq, Carroll, & Ganoe, 2005; Nunamaker, Dennis, Valacich, Vogel, & George, 1991; Valacich, Dennis, & Nunamaker Jr., 1991). For example, gIBIS (graphical Issue-Based Information System (Conklin & Begeman, 1988)) was intended to help groups reach consensus on ill-defined topics though a visual model of the tradeoffs involved and the corresponding rationale behind each decision. In these cases, the system does not use the model, or artifact of planning, for any further purposes. It is simply a visual aid that supports the process of planning but not the subsequent process of enacting that plan.

Most of the systems that have been presented up to this point have been process-oriented in the sense that they rely on logical models of work. However, at the leftmost extreme of the specificity frontier the focus shifts from a process-orientation to an activity-orientation. Activity-oriented systems do not rely on complete models of work processes. Instead, they recognize that most work entails interacting with a variety of distributed resources in order to perform logical “activities” that are part of a larger

6 Tools that are designed for strategic decision-making are more broadly known as group decision support systems.
unknown process. Hence, activity-oriented systems provide users with an environment that makes the transition between activities as seamless as possible (Bardram, 2007; Robertson et al., 2007; Voida, Mynatt, & Edwards, 2008). For example, they often allow users to group related files and desktop applications into named “activities” that can be saved and restored on demand (i.e. because the applications that operate on the files are not natively integrated).

Activity-oriented systems mediate a universal process of getting information from one computerized application into another or from one user to another. Workflow systems do this as well, but it is all that activity-oriented systems do. While a workflow system can rely on knowledge of the process to actively invoke or mediate between individual applications directly, an activity-oriented system merely provides a staging area that can help users do the mediation on their own without a significant loss of context. For instance, Moran (Moran, Cozzi, & Farrell, 2005, p. 69) states that the goal of an activity-oriented system should be to:

rely on social interaction mechanisms to control activities and to impose only as much restraint as is needed.

In this sense, activity-oriented systems are exactly like process-oriented systems. However, they refrain from organizing logical activities into any higher-level structural constructs, like a process, and delegate any further interpretation to the community at large outside the boundaries of the system.
The ActivityExplorer project (Geyer et al., 2006; Millen, Muller, Geyer, Wilcox, & Brownholtz, 2005; Muller et al., 2004) is a concrete example of an activity-oriented design. It provides users with a thread-based collaboration space. Each thread represents a logical activity and any associated items can be added to a thread in a variety of formats, such as documents, emails, and media files. These activity threads can be arbitrarily shared with others as needed to coordinate work. Fundamentally, systems like this (Dragunov et al., 2005; Kaptelinin, 2003; Voida et al., 2008) address the lack of functional integration between vendor applications that has developed as computing has become more pervasive. In some cases, the basic structure of “activities” that these systems use can be reused as templates for carrying out similar work in the future.

Activity-oriented systems are not unlike their process-oriented counterparts. They simply operate on a smaller scale. Process-oriented systems are effective tools for modeling top-down policies and meeting the needs of management, while activity-oriented systems are better suited for organizing bottom-up work practices and meeting individual needs. In organizations that rely on both global policies for resiliency and locally adapted practices for agility, neither paradigm can claim superiority over the other.

2.3 Summary

Researchers have proposed collaborative systems that dot the landscape along the specificity frontier. Nonetheless, observable accounts of real work have demonstrated a mix of routine and non-routine practices that seem to inevitably lead support systems to
over, or under, specify aspects of work. In service-oriented organizations, the contradictory needs to operate reliably, repeatedly, and efficiently while maintaining flexibility and customer responsiveness highlights a need for tools that can support the periodic shifts to and from routine and non-routine work practices. Designers have recognized this need, but a noticeable gap remains in practice.
CHAPTER 3: PATTERN PROPOSAL

Chapter 2 identified a key difference among established design trends in collaborative tools. Superficially, this difference is whether or not explicit representations of work processes are used to mediate interactions. Such explicit models are brittle and can lead to counterproductive circumstances, like being absorbed by bureaucracy or being made accountable for minute actions (Suchman, 1995). Yet, there is another important difference.

Descriptions of processes, like flowcharts or diagrams, are visual aids. They can reveal information about peoples’ intentions without necessarily steering them in predetermined ways, and they can act as a visual language during an on-going planning effort. These representations do not have to be completely accurate or authoritative to be useful, and small process-like artifacts such as these are a part of common practice (Moran et al., 2009). However, few general-purpose collaborative systems have leveraged this potential function. There is a significant difference between designs that actively support a planning process from those that simply make such a process more visible.

In workflow systems these roles are confounded in a number of ways. In some cases, the representation of a process is not intended to be visible to users at all. This allows the
system to drive the work and route pieces of information, but it leaves this visual potential unrealized and it is most appropriate for routine forms of work (i.e. the need to deviate from a fixed plan is rare). In other cases, systems like Regatta (Swenson et al., 1994; Swenson, 1993) do make the visual representations of workflow system processes visible to all parties. However, a heavy focus on enabling users to script action leads to something that is more like a visual programming language than a visual aid, and the potential for the process itself to serve as a visual means of communication is diluted by the presence of precise formal semantics.

3.1 Micro-modeling

This section introduces the micro-modeling design pattern. The intuition for “micro-modeling” comes from the potential that representations of processes have to serve as a basis for visual communication rather than automation. In this role, process representations do not enforce policies or predefined ways of working. Instead, they are approximate and diagrammatic models of actual work practices that provide the kind of organizational visibility that is required for effective and localized self-governance. The micro-modeling pattern is intended to elicit a kind of visual “process model” as a side-effect by restructuring a more common tactically oriented task. In other words, “micro-modeling” is intended to blend the act of doing some immediate task (like sharing a file with a peer) with the larger task of maintaining a visual model of a group’s collective actions (that illustrates how files are being shared, for what purposes, etc.).
Again, this kind of a mechanism is necessary because individuals with tactical responsibilities in an organization may not find any immediate value in maintaining the kind of visibility that a “process” represents. They know what it is that they are doing and why they are doing it and it takes additional time and effort for them to make this information more visible to others. Similarly, they may be under pressure to get their own work done and making their actions and intentions more visible would only amplify this pressure. Yet, the lack of cross-organizational visibility is a potential cause of error and inefficiency. The micro-modeling pattern presented here is an attempt to resolve this conflicting set of needs by minimizing the overhead of process modeling.

The pattern is defined in three parts. First, the problem that it addresses is revisited. Next, the influential forces that cause the problem are identified and, finally, the key characteristics of good solutions are presented (the next chapter will discuss design patterns more generally and each of these three components in more detail). The key aspect of the pattern is that it acts as a hypothesis-in-development throughout the course of this research. It embodies what are believed to be useful design principles. At this point, this belief has primarily come from an analysis of the literature (Chapter 2). Later, the results of a user study will be incorporated into the pattern to refine it.

3.1.1 Problem

Social interaction is a key component to collaboration in both physical and virtual spaces. However, maintaining long-term forms of awareness within a constantly evolving social context requires sustained effort. For example, in larger teams
knowledge of what others are doing, with whom, and for what purposes, becomes fractured and distributed (Bolinger et al., 2011; Bolinger, Ramanathan, & Ramnath, 2010; Martin et al., 2011). There is strategic value in consolidating this knowledge to prevent errors, reduce inefficiencies, and avoid redundancies. However, there is a significant cost associated with maintaining that kind of centralized visibility. Individuals under pressure to carry out their day-to-day duties may perceive it as an intrusive and unwelcome overhead despite any potential longer-term organizational benefits. In these kinds of environments strategic governance mechanisms for establishing visibility must be carefully designed. They must serve the best interests of an entire community without frustrating the individuals that have immediate responsibilities.

3.1.2 Forces

How information is structured, organized, and visually presented in collaborative tools is an important consideration. Often it is organized graphically into folders, directories, or through similar constructs drawn from the legacy of the desktop metaphor of other equally familiar modes of interaction. While they may be familiar, these modes of interaction are not necessarily ideal (for a catalog of similar critiques see (Kaptelinin & Czerwinski, 2007)). The structure and organization of data, documents, and messages within collaboration tools is itself a form of metadata. Yet, the role of a graphical
interface is most often thought of as solely a data input device. It is not often seen as a device for strategic data collection. However, different prompts often elicit different responses and that does influence the kind of data that can be collected. In reality, an interface is both a data input and a data collection device. It is an input device from the perspective of a user, but a data collection mechanism from the point of view of a group of users.

Consider the following example; a user adds a document to a folder in a shared file system. Does that action reveal anything useful to other users? If a prior convention had been established it might have an implied meaning but otherwise it probably does not. Now, allow the user to add an icon to each document that describes its type (i.e. presentation, report, meeting minutes, etc.) and some additional metadata can be captured. However, that makes users do extra work that they may not be willing to do regardless of the value that the additional classification icon may have to others. The key to building effective group interfaces is to design input mechanisms that can "collect" useful metadata, like the classification icon, as a side-effect and without explicitly asking a user for it.

A key force to consider is the influence that an interface will have on the kind and format of data that is captured by a system. This data must allow the user to complete

7 Collecting interface usage statistics (such as click locations, frequent navigation paths, etc.) is another form of indirect "data collection," but here the focus is on collecting information about the user or their environment (i.e. domain data) and not this kind of system usage data.
their task, and the interface should do so with respect to commonly accepted usability criteria (see (Nielsen & Mack, 1994) for example). However, an interface that is capable of capturing more data, or isomorphic data that is organized in a more useful structure, without sacrificing usability to a perceivable degree should be preferred over an interface that captures less data. In complex service organizations, any additional data that can be captured about users work practices that they would be willing to openly share and self-report could enhance incumbent social governance processes.

3.1.3 Structure of a solution

Process diagrams concisely represent information about what others are doing, for what purposes, and how the work of one individual relates to another. Very simple process diagrams, like flowcharts (i.e. box and arrow diagrams), can be used as an interface mechanism to promote visibility within a group in an intuitive way. Such diagrams do not need to actually direct work or be totally accurate representations of reality to act as visual aids and promote awareness. Rather, they merely provide a channel for process-centric communication that is not present in most collaborative tools.

Process diagrams can cater to the strategic need for group visibility when they are used as an interface metaphor in place of process-agnostic organizational primitives, like document collections, folders, or threads. Even simple box and arrow style flowcharts can reveal information about the intentions of users in ways that collections of files and other organizational constructs cannot. They have the potential to capture data that may otherwise never leave the privacy of an individual’s mind if they are used in place of
simpler organizations primitives. However, these diagrams must be integrated into
collaborative systems in ways that users will benefit from directly, and they must not lead
to overly complex interfaces.

Process-centric collaborative interfaces are ones that use the concept of a work
process as an interface metaphor. They have the following general properties:

A visual language: The graphical user interface resembles some kind of process-like
representation, such as a diagram, flowchart, or list. The visual aspect of this process
should act as a navigation aid while allowing users to carry out direct tasks, such as
sharing pieces of information. The interface should create an opportunity for users to
describe their own logical process as they interact through it, but this should not be the
primary activity that they are engaged in. “Their process” is an electronic approximation
of their mental model of their work practices to the extent that they are able to articulate
them. It may be incomplete and users may not always think of what they are doing in
terms of a “process,” but the interface should allow them to complete their tasks while
partially exposing their internal planning process.

The visual language should act as a kind of boundary object (Star & Griesemer,
1989). It must allow the representation of a process to be understandable to individuals
that are in different functional areas of an organization, with different backgrounds, or

8 However, complexity can be very difficult to quantify and it is yet another reason why
collaborative tools are challenging to evaluate objectively and why interface design trends have
changed so slowly.
distinct roles. The language must create an opportunity for users to attempt to communicate process-oriented information to others. It may be an incomplete or noisy source of data, but the key characteristic of this visual language is that it elicits information that would otherwise not be transmitted at all.

_A visual interlock:_ The role of a visual language is to facilitate the sharing of process-oriented data. However, it must not be an “extra step” that needs to be taken by a user. It should be a necessary step that is required in order to do what the user really wants to do, which is to share some piece of information that is relevant to their immediate task. Hence, it should function as an interlock that prevents, to a reasonable degree, individuals from sharing information without revealing additional metadata about what it is that they are sharing and why.

_A minimal overhead:_ Interlocks cause a perceived overhead that can frustrate people that have to work with them. Worse, interlocks do not typically have any short-term benefits. They prevent errors, minimize the consequences of accidental slips, or enforce policy. In this case, the interlock attempts to keep a communication channel open that might otherwise close. The overhead of this interlock has to be very small (i.e. perhaps on the order of seconds) compared to other readily available means for sharing information. The role of the visual language should not be to capture a large amount of additional metadata. It only needs to collect enough information to keep the types of communication that contribute to shared forms of awareness alive. The bulk of the
communication can take place elsewhere. The interface itself functions only as a catalyst.

A participatory community: A written work process is typically thought of as a singular entity. In workflow systems, and even in casual communication, when it is said that people are following a process it is implicitly singular. However, this has very little resemblance to reality (Moran et al., 2009; Suchman, 1987). Individuals may not consciously think of their actions as contributing to a coherent process even if their behavior is somewhat predictable. Those that do claim to follow a high-level process may have their own preferences and variant styles of working that are significantly different from others.

The visual language must be capable of taking input from everyone and aggregating it in an appropriate fashion. The language should resemble a process but it should not be a singular authoritative process. The goal of the language is not to create a single representation that mirrors the reality of practice. Rather, it is to make inconsistencies visible and to give everyone an opportunity to show their colleagues how they logically structure their own work in relation to others.

3.1.4 Summary

The micro-modeling pattern integrates the tactical processes of sharing information with the strategic processes of maintaining an awareness of the intentions of others. It works by creating a visual interlock mechanism that gathers process-oriented metadata as a byproduct of online interaction. It is most appropriate for long-running and
asynchronous types of distributed work where there are few other channels for exchanging process-oriented information on a regular basis. The pattern is intended to prevent missed opportunities and reduce inefficiencies in these kinds of environments, which are common in complex service delivery organizations. It is not applicable when stronger forms of situational awareness are needed to prevent serious errors or negligent action.

3.2 Prototype: BlackTie

The micro-modeling pattern itself is only a high-level set of guidelines. To demonstrate how it can be applied in practice a prototype system, BlackTie, was developed. In this section, the key features of BlackTie are presented.

3.2.1 A “shared document repository”

BlackTie mimics the basic functionality of a multi-user document repository. Like most document sharing software⁹, it supports the following set of high-level use cases:

1. User creates an individual account.
2. User logs in to their account and views a collection of “shared repositories” that they have permission to view (i.e. those that they have created or were granted access to).
3. User creates a new named “shared repository.”

---

⁹ Examples include IBM's Lotus Notes, Google Documents, and similar products.
4. User uploads/downloads/deletes a document to/from a “shared repository” that they have permission to view.

5. A user updates the permissions on a “shared repository” that they have permission to view (i.e. to allow other users to view/edit documents in it).

### 3.2.2 Box & arrow diagramming

The most direct way to implement this set of use cases is to follow the familiar file/folder metaphor where each “shared repository” is a folder (i.e. a named collection) of documents that can be shared with multiple users. However, in BlackTie each “shared repository” is a distributed “process” diagram rather than a folder. In order to share a document, or add it to the repository, a user must sketch a two-dimensional partial box and arrow style process diagram. Then, documents can be added to specific components of that diagram, which function exactly as folders do.

Users draw diagrams by clicking or dragging elements from a palette on the right hand side of the interface on to the main canvas in the center. Figure 2 shows BlackTie’s process-centric user interface after a user has navigated to an empty “process” (the call-out boxes and highlights have been added and are not a part of the interface).
3.2.3 A visual language

Users “draw” process diagrams by using primitive elements from BlackTie’s visual language. These elements include Activities and Work Products (displayed as boxes) and Rules (arrows). Activities are visual representations of work tasks, like “estimate the cost of widget X”. Users can create multiple activities in order to describe some higher-level activity, or process, that they are working towards. Note that activities do not have any real semantic meaning. They only serve as a visual means of organizing information. What an Activity represents is entirely at the discretion of the user. Distinct users may
use language differently or model the same logical work activities in different ways. An example of an Activity is shown below in Figure 3.

![Figure 3: BlackTie – Activity](image)

Activities allow users to organize chunks of information and share them with other users. Users can perform two important operations on Activities. First, they can add any number of Work Products to an Activity. Work Products represent sets of documents that are related to an Activity. They are displayed as boxes surrounding the Activity with a short line linking them. They function exactly as folders do in a traditional application
by allowing users to store and retrieve documents. An example of a Work Product is shown below in Figure 4.

The second important function that Activities provide is the ability to “assign” a set of users to them. When an Activity is created only the user that created it has permission to view it. In order for another user to see it, and all associated Work Products, they must be assigned to it by someone that already has permission to view it. Hence, Activities allow users to share chunks of a larger logical process without necessarily sharing all of

Figure 4: BlackTie - Work product
the details with everyone that they collaborate with (this aspect is discussed in more
detail shortly).

Rules allow users to draw explicit relationships between Activities, and they are
displayed as directional arrows between two Activities (or the same Activity). Their role
is also primarily as a visual indicator, like the other elements of BlackTie. However,
Rules can optionally be set to automate a few basic actions such as notifying users about
changes to connected Activities via email or by moving a document from one Work
Product to another in response to specified events.

It would have been possible to make Rules more powerful and capable of performing
other kinds of actions, but they were purposely kept simple. Recall that a major design
goal was to construct an interface that would elicit process-oriented information as a by-
product of document sharing (i.e. a diagram) while minimizing additional complexity.
Therefore, Rules were restricted to a few essential actions. An example of a Rule is
shown below in Figure 5.
Activities, Work Products, and Rules form the basis of a box and arrow style visual language. But, there is a limit to the amount of information that can be displayed as a box without requiring the user to perform excessive scrolling or zooming in order to navigate. In BlackTie, each of the three basic elements has a magnifying glass icon that can be used to expand the element into a full-screen details page (in the upper right corner). For Work Products this page allows the user to manage the documents that are housed within it. The details page of an Activity provides the user with an ordered to-do list, a list of all the documents in connected work products (providing an alternative way to navigate through them), and a widget for sending email messages to everyone assigned to that Activity. An example details page for an Activity is shown below in Figure 6 and for a Work Product in Figure 7.
Figure 6: BlackTie - Activity details

Figure 7: BlackTie - Work product details

A collection of documents (or URLs).
You may upload/download files to store them on the BlackTie server.
3.2.4 Distributed modeling

A process diagram is often thought of as a canonical and singular view of the world. For example, in workflow systems a singular description of a process is necessary for consistent and repeatable enactment of that process. Similarly, it is this tendency to use representations of processes as normative artifacts that leads to their brittleness in volatile environments and why they are not a part of most collaborative tools. However, process descriptions are neither inherently singular nor normative. They have the potential to be used as comparative artifacts that allow individuals to express their particular view of the world (i.e. how work is or should be done), and to see how their worldview may or may not differ from others.

This is the role that process diagrams play in BlackTie. Consequently, a process in BlackTie is a distributed model. It may look different from the perspective of each user, although some level of consistency is maintained when it is appropriate. This aspect of the system is best understood through an example. Therefore, consider three different users; Alice, Bob, and Cindy. At time T0, each has been given an account that they can use to access BlackTie but they have not done anything else. The following sequence of figures describes the events that take place next. Each figure shows a wireframe version of BlackTie’s interface at the same point in time from the perspective of three different users (wireframe mock-ups were used here instead of screenshots for clarity and to reduce the size of the figures).
First, Alice logs in to BlackTie and creates a new process with the name “Example.” She then creates a single Activity and gives it the name “A.” Figure 8 shows the state of the system from the point of view of each user at time T1.

![Figure 8: BlackTie - Distributed model at time T1](image)

Next, Alice assigns Bob to Activity “A” in the “Example” process. At this point Bob can log in to the system, is able to see the “Example” process in the list of processes that he can view, and is able to open the “Example” process for editing. Figure 9 shows the state of affairs just after Bob opens the “Example” process at time T2 (the blue dot on Activity A in Bob’s view is a marker that the system adds to elements that were not created by the user that is viewing them – it can be dismissed at that user’s discretion).
At time T3, Alice moves Activity “A” to a new position. Although both Alice and Bob are looking at the “Example” process it may appear in different geographic positions. Certain properties of Activities, such as the name and set of assignees, are always consistent from the perspective of all users. However, other properties, like position, are not. Figure 10 shows the state of affairs at time T3.
Then, Bob creates a new Activity “Y” in the “Example” process. At time T4 he is the only person assigned to it, which implies that he is the only person that can see it. Figure 11 shows this situation.
Bob assigns Cindy to Activity “Y” at time T5. At this point, Cindy is able to see the “Example” process in the list of processes that she can view and she opens it for editing. Figure 12 shows BlackTie’s main interface from the perspective of all three users, which are now all actively editing the “Example” process. This illustrates the distributed nature of a “single” process in BlackTie. It is more like a set of interleaved, or partially overlapping, process diagrams than a single normative flowchart.
At time T6 Cindy adds a Work Product to the Activity “Y.” The relationship between Activities and Work Products is another property that is consistently maintained throughout the process. Hence, Bob immediately sees this change in his view of the “Example” process. This is shown below in Figure 13.
This distributed nature allows the diagrams to be used in a comparative fashion instead of a normative one, and it also limits the amount of information that each user needs to process. For instance, users with different organizational roles or responsibilities may have concerns that do not completely overlap. Therefore, they do not necessarily need to see potentially irrelevant aspects of a process that are relevant to others. Similarly, it allows each user the freedom to organize and model their work in the way that makes the most sense to them (i.e. putting the same Activity in different positions). Some level of consistency, like consistent names, is needed so that the diagrams can play a role in coordination without creating confusion.
On the other hand, this distributed nature of a process in BlackTie leads to an ownership problem. For example, if Bob discovers that Activities “X” and “Y” are really representations of the same thing what should be done? Perhaps, Bob believes “X” and “Y” are the same but Alice disagrees. Who should have the ability to delete or merge Activities, and how can it be done without confusing users that may have significantly different views of the process?

To avoid confusion, elements in BlackTie can only be deleted after everyone explicitly agrees to delete them. Users can ignore things in BlackTie such as Activities to signal that they should be deleted. An element that is ignored becomes minimized from the point of view of the user that is ignoring it, but it is unchanged to others (visual cues are given to alert others if a majority of users are ignoring something that they are not). An element is only truly deleted when everyone that could possibly ignore it actually does ignore it\(^{10}\). In Figure 14 Cindy creates a new Activity at time T7, which Alice deems irrelevant and deletes at time T8 (Figure 15).

\(^{10}\) This is certainly not the only way that ownership issues could have been resolved, and the method selected here may not work well in large communities. Exploring the various ways to handle ownership issues in different communities is an interesting area of future research.
Figure 14: BlackTie - Distributed model at time T7

Figure 15: BlackTie - Distributed model at time T8
BlackTie processes are distributed because they are intended to catalyze process-oriented forms of communication that might not otherwise occur. They are not intended to yield singular or authoritative process descriptions, or to enforce particular ways of working. However, if users cannot see how their view of a process differs from another user’s then their power as comparative artifacts is substantially reduced. For this reason, BlackTie includes a “People Bar” along the top of the interface that allows users to see, but not edit, the process from the perspective of another user. This is shown below in Figure 16.

Figure 16: BlackTie - People bar
3.2.5 Reusable processes

BlackTie is intended for collaborative types of work that are largely non-routine. However, BlackTie processes are specified at a relatively high-level and in some situations it may be useful to reuse the general structure of an existing process as a template (similar to (Millen, Muller, Geyer, & Wilcox, 2005; Moody, Gruen, Muller, Tang, & Moran, 2006)). The system supports this by allowing processes to be reused (i.e. the structure is repeated but elements are reset to an original state). But, this potential for reuse is not particularly novel and it was not relevant during the pilot study (see Chapter 5). It is noted here only for the sake of completeness.

3.2.6 An isomorphic representation

Comparing the system to a functionally equivalent alternative is another way to show the influences that the micro-modeling pattern had on its design. Without the graphical interface, BlackTie implements a shared document repository with a fixed 3-tier structure with documents stored only in the bottom level of the hierarchy. Graphically, this is shown below in Figure 17.
Although users are likely to be more familiar with the file/folder design on the left hand side of the figure, BlackTie is capable of organizing information in an analogous way. However, it imposes a certain way of working on the user by using a very different organizational structure. This is not intended to make the task of organizing documents any more difficult or time consuming. An experienced user should be able to upload, download, and share documents with roughly the same amount of time and effort. But, the semantically richer construct of process as an organizational device is designed to help users understand the intentions of their peers and give them an opportunity to see how they describe their collective “process.”

**3.2.7 Summary**

BlackTie is a prototype collaborative system based on the micro-modeling design pattern. It functions as a document repository but uses a process-centric graphical user interface in place of the traditional file/folder metaphor. The process-centric interface functions as a minimally invasive interlock that prevents users from sharing information
without revealing additional metadata about their logical work processes (at least to a degree). Similarly, BlackTie processes were designed as distributed diagrams to make them more useful as comparative communication artifacts, and less useful as normative models of work or as organizational accounting devices.
CHAPTER 4: METHODOLOGY

The goals of a research study should be the primary factor in selecting the appropriate methodological tools (Charmaz, 2006; Corbin & Strauss, 2008; Creswell, 2008; Yin, 2008b). However, design-oriented research is not a traditional science and there is not a standard method. Studies of collaborative technology have used methodologies that span from the logico-deductive and axiomatic to the hermeneutic and interpretive. Regardless of method, the outcome is most often some form of heuristic because the goal of the field is to invent new technology and to understand its role within a constantly evolving society. In this chapter, the major research traditions in the CSCW community are identified and used to lay the foundation for a new kind of hybrid research methodology that frames heuristic knowledge within the structure of software design patterns.

4.1 Research traditions

Designing collaborative software is challenging (Grudin, 1988), and a large part of the difficulty stems from the pragmatic realization that group behavior is at least partially situated and inseparable from the context in which it occurs¹¹ (Allen, 1977; Carroll, Kellogg, & Rosson, 1991; Suchman, 1987; Suchman, 1994; Suchman, 1995; Vicente, aims.

¹¹ Even if grand sociological theory does exist it certainly has yet to be identified.
1999; Winograd, 1994). Software is but one component that shapes behavior and organizations often struggle to understand the intricacies of their own practices well enough to map their needs to available technology, much less design new forms of it (Matthews, Whittaker, Moran, & Yuen, 2011). As a result, engineering processes and design methodologies have received much attention in the literature.

A key component of that body of knowledge is the recognition of widely reusable heuristics, or strategies, that can be followed to reduce the risk of failure and the cost of development when an off-the-self approach to technology is simply not possible (Borchers, 2008; Nielsen, 1989; Schmidt & Bannon, 1992; Woods & Hollnagel, 2006). This knowledge is the result of research that falls under one of the three logical categories introduced below.

4.1.1 Searching for needs

A large part of CSCW research is focused on identifying the needs of people and groups in organizational settings. Field studies, experiential case reports, and other observational forms of research are the most commonly employed traditions (Schmidt & Bannon, 1992). This style of research has had a significant impact on the field for a number of reasons. First, it has shown in rich detail just how idiosyncratic and complex seemingly simple work practices can be (Bowers et al., 1995; Malone, 1983; Suchman, 1987). It is also a valuable test bed for proving design methodologies and refining
domain analysis techniques (Muller & Kuhn, 1993; Preece et al., 2001; Vicente, 1999). However, it is also indicative of a practical dilemma in much of design-oriented research.

What do these needs actually reveal to a wider audience and how do they contribute to a field of general interest? Their situated, transient, and idiosyncratic nature implies that they are rarely generalizable. Micro-sociological analysis can illustrate good engineering practices, reveal innovations that have had an effect in particular settings, and they often lead to entertaining stories. However, they are of limited value in new contexts. Revealing the unmet needs of users or the naivety of past software systems does not yield insights that will necessarily be relevant in new situations (Carroll et al., 1991; Norman, 2005; Suchman, 1987).

This has implications for researchers. Identifying needs, requirements, or tasks that can be supported by technology cannot be the sole focus of design research. They are not static concepts and something more fundamental is needed. Norman recognized this imbalance in his article “Human-Centered Design Considered Harmful,” an excerpt of which is reproduced below (Norman, 2005, p. 16):

One concern is that the focus upon individual people (or groups) might improve things for them at the cost of making it worse for others. The more something is tailored for the particular likes, dislikes, skills, and needs of a particular target population, the less likely it will be appropriate for others.

12 There has been a historical disconnect between behavioral researchers and those that actually develop new software systems due to their respective backgrounds and approach to research, but this is slowing changing (Edwards, Bellotti, Dey, & Newman, 2003).
The individual is a moving target. Design for the individual of today, and
the design will be wrong tomorrow. Indeed, the more successful the
product, the more that it will no longer be appropriate. This is because as
individuals gain proficiency in usage, they need different interfaces than
were required when they were beginners. In addition, the successful
product often leads to unanticipated new uses which are very apt not to be
well supported by the original design.

Norman’s statement was not intended to imply that understanding the unique needs of
individuals and groups is without value. Human-centered design is not really harmful,
but it is an ironic term. Understanding needs does not reveal how to meet them and,
perhaps worse, needs are not constant. They vary over time and they are often related to
or even in conflict with one another. What practical value does the publication of needs
have if they are idiosyncratic, transient, or localized, and of what value do they possess
beyond their novelty?

4.1.2 Searching for truths

Experimental methods have an equally important place in the literature. As a design
research method they are particularly well suited to two kinds of situations. First,
controlled laboratory studies have been instrumental in improving interface design from a
single-user perspective and much of this applies to multi-user systems as well (i.e. results
from the first wave of human computer interaction research (Bannon, 1995; Bødker,
2006)). However, these results often rely on generalities in the human form, like Fitts’s
law of pointing, the Gestalt principles, or the biasing effects of alternative representations
of the same information (Fitts, 1954; Zhang & Norman, 1994). When social processes
are considered, the correspondence between the laboratory and the field becomes more
difficult to establish (as does generalization (Bannon, 1995; Convertino & Carroll, 2011; Vicente, 1999)). Experiments can be controlled but they may not produce useful results if the localized nature of work practices cannot be positioned within more generic patterns of behavior.

On the other hand, experimental methods are particularly well suited to situations in which groups are large enough to command special attention. For example, regulated and high-risk groups of workers, such as soldiers and power plant operators, perform the kind of work that is worthy of special attention. The theoretical understanding of collaboration in these kinds of specialized domains is greater as a result, but that knowledge is not generally transferable to other contexts.

Collaboration in other domains does not attract as much attention, work practices are not as regulated, and they are often less accessible to researchers that are not members of the community (Edwards, Bellotti, Dey, & Newman, 2003b). Additionally, if collaborative tools support a conflicting set of user needs, as it has been consistently argued that they do, there is a risk that any measurable change could be construed as a positive one from someone’s point of view. It is sometimes clear what positive and negative social behavior is and how it contributes to organizational performance, but it is not always so obvious and the relevant influences can be surprising (Suchman, 1994).

4.1.3 Searching for uses

One of the most challenging problems in CSCW research is the availability of data. It is easier to scientifically examine technology that is widely deployed in practice than it is
to experiment with new systems. The literature is filled with thought provoking
prototypes and formative pilot studies (see Chapter 2), but there is little theoretical
distillation because the scope of the research is so tightly constrained.

The situated nature of design in practice prevents researchers from moving much
beyond face validity, regardless of their methods, because they simply lack the resources
to exhaustively validate their assumptions and their “preliminary” findings. Rigorous
forms of assessment are only possible after large-scale commercial adoption, which often
has less to do with the merit of a particular design than it does on other influences. This
distances the results of novel CSCW systems from genuine practice and it is why new
methodological approaches need to be developed.

Traditional scientific “replication” is beyond reach in most CSCW studies (Wilson et
al., 2011) yet some kind of reproducibility is necessary. If the results of pilot studies are
to be of any lasting value in a research-oriented field they must support a kind of
incremental theory building that approximates the repeatable aspect of the general
scientific method. Fortunately, this quality is exactly is what separates a good heuristic
from a bad one. The best heuristics are not always accurate, but they are correct often
enough to be useful in practice.

4.2 Methodology I

CSCW researchers play a part in developing novel and frame-breaking guidelines for
building collaborative tools. But, there is no compelling proving ground for their new
ideas. Despite the diversity in research methods none have the power to accurately
predict how new systems will function within different organizational cultures and diverse settings. To summarize, the three major research traditions are relisted below with a set of representative examples.

1. Developing design guidelines from the observation of technology that has been widely deployed at particular locations. Examples include: (Abbott & Sarin, 1994; Allen, 1977; Barrett et al., 2004; Bowers et al., 1995; Dourish, 2001; Grinter, 2000; Kogan & Muller, 2006; Kueng, 2000; Malone, 1983; Strauss, 1988; Suchman, 1994).

2. Experimentally benchmarking specific features of technology against reference points (i.e. common tasks). Examples include: (Bergman, Tucker, Beyth-Marom, Cutrell, & Whittaker, 2009; Carroll et al., 2009; Convertino & Carroll, 2011; Convertino et al., 2009; Moran et al., 2009).

3. Piloting novel technology at locations where researchers can intervene and change established work practices. Examples include: (Bernstein, 2000; Bolinger et al., 2011; Bolinger, Horvath, Ramanathan, & Ramnath, 2009; Bolinger et al., 2010; Castellani & Pacull, 2002; Dragunov et al., 2005; Geyer et al., 2006; Grundy & Hosking, 1998; Kaptelinin, 2003; Kushmerick & Lau, 2005; Medina-Mora et al., 1992; Millen et al., 2005).

From an ideal perspective each approach is complementary to the others. However, in reality they are separated by large gaps. Commercial and political forces create a rift between where new technologies can be invented and where deployed systems can be
studied in the wild. Pilot studies and controlled experiments can demonstrate novelty, but this novelty is also what makes it challenging for outsiders to interpret the results that are born in laboratories or from within foreign communities of practice. Occasionally, there is enough established sociological theory available to showcase compelling design principles, but more often than not there are simply too many relevant factors to consider (whether or not they are known or unknown in advance).

These gaps create the need for a unifying heuristic device that can support long-term and incremental reasoning. In design-oriented fields heuristics are the most valuable type of result because they go beyond describing interacting effects and point to good solutions. They may not be totally correct, but in practice having the most likely explanation available now is more useful than a perfect theory might be at a later time. However, “heuristics” tend to be tacked on to the end of otherwise rigorously established results as a best effort to explain a half-proven set of hypotheses or a mountain of qualitative observations.

New designs are like theories about the world, cognition, and social dynamics and they should be treated as such. The goal of CSCW is the development of heuristic knowledge yet the methods of researchers do not match this theory-building orientation. Instead, they rely heavily on classical theory-proving methods or simply abandon the search for any principles that might have wide applicability. This had lead to a body of engineering knowledge that is expanding in volume but not necessarily one that is becoming cohesive. Fortunately, the software engineering community has identified a
unifying heuristic device that could be used to support longer-term and incremental heuristic development if it were positioned within a theory-building scientific method.

4.2.1 Design patterns

Design patterns are concise and reusable strategies for solving frequently encountered problems in practice (Gamma, Helm, Johnson, & Vlissides, 1993). Examples of design patterns can be found on topics ranging from software engineering (Erl, 2005; E. Gamma, Helm, Johnson, & Vlissides, 1995; E. Gamma et al., 1993; Hohpe & Woolf, 2004; Schmidt, 1995), software methodology (Coplien & Harrison, 2005), usability (Bass & John, 2003; Carroll & Rosson, 2005), to interaction design (Borchers, 2008; Schummer & Lukosch, 2007).

Essentially, design patterns are heuristics. What makes design patterns unique among other types of heuristics is the way in which they are structured for dissemination throughout professional communities. Popular patterns can act much like shorthand or a technical jargon, and they can consolidate the implications from diverse research efforts into a concise and well-annotated package. Although different authors have proposed various guidelines for presenting design patterns, Alexander’s original three-part format is common to them all (Alexander, 1979, p. 253):

We see, in summary, that every pattern we define must be formulated in the form of a rule which establishes a relationship between a context, a system of forces which arises in that context, and a configuration which allows these forces to resolve themselves in that context.

It has the following generic form:
In other words, design patterns begin with an abstract description of a recurring problem that often needs to be solved in practice. This helps others recognize the real-life contexts where it can be applied. They also include a discussion of a set of conflicting forces, which are the factors that are believed to cause undesirable situations. Most importantly, at the heart of a pattern is a discussion of how to analyze the significant forces in the environment, and the presentation of one or more logical strategies for changing the context in order to put it in a more desirable state (the configuration or structure of the solution). They are often also annotated with examples and references that support the advice given in the pattern, which allow them to function as an index into larger bodies of knowledge.

Patterns are not reusable solutions, but they are representative of a hypothesis that is in development. As hypotheses they are not yet mature enough to be proven in any convincing manner but they have value because they are often the “best” available explanation. Patterns describe common problems and give guidance on how to solve them because that guidance has had some success in the past. They outline the basic structure of a good solution, but they also recognize that each situation is unique and in need of individual concessions. Design patterns are not law, and it is entirely up to someone applying the pattern in practice to apply it appropriately and to solve the problem at hand (referred to as the generative quality of a pattern (Alexander, 1979)).
4.2.2 The pattern lifecycle

Design patterns are used to distill lessons learned over time. However, patterns have never been fully integrated into the scientific method of researchers. There is no systematic method for deducing, or arriving at, a design pattern. In some senses this is acceptable. Patterns represent heuristic knowledge, not fact, which clearly cannot be logically deduced in any sensible way. Nonetheless, as hypotheses they must be generated somehow and there is a difference between good hypotheses, which are worthy of attention and resources, and poorly conceived ones, which are not.

In design-oriented fields like CSCW it is ironic that so much effort is spent attempting to verify theories (i.e. novel designs) for which there are rarely sufficient resources to prove in any convincing manner (Wilson et al., 2011). When theory is immature or the conditions necessary for the replication of a study is not clear, the empirical rigor is better spent on the process of shaping good hypotheses rather than on the process of verifying poorly stated ones (Glaser & Strauss, 1967). Design patterns represent hypotheses that cannot be proven, and as a heuristic device the method for systematically shaping them is still missing.

4.3 Methodology II

Data has two purposes. It can be gathered as evidence to generate hypotheses or it can be used to confirm or deny them. Both uses are important but the former is crucial in applied research disciplines where discovering fundamental laws is not the goal. Design researchers tend to operate in an iterative fashion by understanding user needs, building
prototypes, and validating that their designs have met those needs. Yet, that entire process is really all about theory building and identifying principles that will inform design in the future. It has little to do with verification in the scientific sense, but a large amount of emphasis is placed on validating prototypes and assumptions in situ through deductive scientific methods.

Validating prototypes in the “small” is a good practice, but to move design research forward an equal amount of attention should be given to rolling those lessons learned into genuine and enduring design principles that are relevant on a wider scale. In this section, a hybrid research methodology is developed by integrating design patterns and grounded theory with this goal in mind (see (Baskerville & Pries-Heje, 1999) for a partially related approach that combines action research with grounded theory).

4.3.1 Methodological development

The need to construct abstract theories from field studies is causing new methods to emerge in mainstream CSCW research (Muller & Kogan, 2010). One of the most dominant is grounded theory, a qualitative research methodology developed over the past five decades in the social sciences (Charmaz, 2006; Corbin & Strauss, 2008; Glaser & Strauss, 1967). It consists of a set of conceptual guidelines that researchers can follow to arrive at a coherent explanation of some data. The resulting “theory” is not a formal theory, at least in the traditional scientific sense, because it explicitly considers only a very limited set of data. This kind of theory reflects one possible explanation of an observed phenomenon that fits all of the available data, which may or may not be
generalizable to a wider set of contexts (Corbin & Strauss, 2008). Much like traditional scientific theorists, grounded theorists try to ensure that their emerging theories explain data in a way that meets certain generally desirable conditions, such as being concise, parsimonious, and pragmatic.

Grounded theory has evolved into a number of distinct and significantly different branches since it was first proposed. Here, grounded theory should be understood as the method developed by Corbin and Strauss as described in (Corbin & Strauss, 2008) with the noted inclusion of design patterns and a few other small changes. Grounded theory is not normally used to “evaluate” or “explore” new technology. It was intended to be an observational method and it is not typically used with interventions. Hence, some minor modifications had to be made here to account for its use in a design-oriented research process.

The hallmark of all forms of grounded theory is that it is a generalized method of constant comparison, and what separates Corbin and Strauss’s version from the other branches is a focus on procedure and operational transparency (for reference, other approaches place more emphasis on personal experience, intuition, or interpretive processes, which are not as useful for systematic heuristic development). Strauss’s grounded theory is not quite a mechanical method, but it does outline a set of specific techniques with the explicit goal of minimizing researcher bias and maintaining an objective empirical orientation throughout the process. It is this quality of reducing
forms of bias when analyzing the results of pilot studies that makes grounded theory particularly relevant to design research.

The key procedures of grounded theory allows for the development of a substantive theory that is tightly bound to the context being studied based on any and all available data. Much like design patterns, the developed theory is essentially a hypothesis-in-development. Such hypotheses are not definite but they make conjectures that go beyond the substantive area in which they are grounded. They are the outcome of an abductive reasoning process and such a “grounded theory” is essentially the first step towards heuristic development.

Grounded theorists often say that the method works in the opposite direction from classical deductive research strategies. However, the method is better described as a precursor to these kinds of research methods. It allows researchers to construct hypotheses in complex domains with an increased level of confidence and that is the first phase of any research program. The method is not appropriate for moving forward with verifying hypotheses as they mature, but it can help avoid wasting effort on ill-formed hypotheses that are not likely to be of much value.

Figure 18 below shows the role of grounded theory (the darker box) as it relates to a traditional line of inquiry along the top of the figure (the lighter box). The overlapping section is where grounded theory shares commonalities with deductive research strategies, and the dashed arrows emphasize where that similarity stops. The key point is that grounded theory can help construct hypotheses through contextualized data, which
makes it a particularly relevant method when background knowledge alone is not sufficient to theorize with confidence.

Figure 18: The role of grounded theory

In the CSCW domain researchers construct new hypotheses all the time. Every novel collaborative system is representative of a theory about the nature of interaction and how it will affect any given community. The “best” designs are built with an understanding of lower-level theory but they come to represent new theories as assumptions are made to fill in the holes and to build a complete socio-technical system. This aspect of CSCW research resembles the same set of concerns that Glasser and Strauss had with sociological research methods in the mid 1900’s, and it is why they felt that a theory-building method was necessary in their field. Quite simply, there are no “grand” or unifying theories of cognition and social behavior that are worthy and in need of more
empirical proof (in their case) or on which to base design (in this case). Existing theory
is not adequate to explain all facets of behavior and to predict the effectiveness of new
technology. Examining the theories that do exist is certainly not without value, but
because this theory is known to be incomplete it is equally valuable to search for rival,
complementary, and alternative explanations of such complex phenomena.

In design research studies are often used to validate assumptions. However, that
should not be the only focus of the research. Studying novel forms of technology through
pilot studies presents as much of an opportunity to observe actual behavior as it does to
evaluate expected behaviors. Design researchers can play the role of a theory-verifier by
looking for expected patterns of behavior or they can take on a theory-building role by
focusing more broadly on all of the interesting behaviors that emerge as people interact
with new technology. Which role should researchers play and what should they focus
their attention on?

There is not a unique answer but it is important to remember that CSCW is an applied
field. Testing a new design is not the same as verifying a more fundamental scientific or
behavioral theory that may have motivated certain design choices. The outcomes of a
design process are socio-technical systems that have virtually no precedent. There is
rarely, if ever, enough theory to predict how the whole system will work. Yet the trend is
to verify predictions nonetheless. Often, these predictions are based on lower-level
theories that influenced design choices as if there were a reason to assume that design is
somehow compositional in nature (recall Winograd and Suchman’s debate in Chapter 2).
Glasser and Strauss pointed out a similar lack of theory and a comparable misplacement of effort when they argued for the inclusion of theory-building research methods in sociology (Glaser & Strauss, 1967, p. 10).

Currently, students are trained to master great-man theories and to test them in small ways, but hardly to question the theory as a whole in terms of its position or manner of generation. As a result many potentially creative students have limited themselves to puzzling out small problem bequeathed to them in big theories. A few men (...) have seen through this charismatic view of the great men sufficiently to generate “grand” theories on their own. But even these few have lacked methods for generating theory from data, or at any rate have not written about their methods. They have played “theoretical capitalist” to the mass of “proletariat” testers, by training young sociologists to test their teachers’ work but not to imitate it. [sic]

Integrating design research with grounded theory is a way to separate fields like CSCW from their more fundamental influences (i.e. cognitive science, social psychology, corporate anthropology, etc.). CSCW could benefit by developing its own set of theories that explain how prominent design features influence behavior. It may be challenging to identify formal theories because of the wide range of factors that need to be considered, but heuristic theory is not beyond reach.

However, this requires that researchers break free of the implications that the more basic scientific fields have brought into CSCW. These implications should not be ignored but there is no need to verify them again in yet another location. That type of verification is misleading in any event because CSCW systems in practice are not equivalent to whatever rationale was used to develop their constituent parts. For this
reason, design researchers have more to contribute by playing the role of a theory-builder than they do as theory-verifiers.

Integrating a method like grounded theory into design research requires that some relatively minor modifications be made. But, before discussing the necessary alterations it is useful to outline a stereotypical grounded theory process to introduce the method and to expose the potential issues that might arise if it is used in a design context. Like all research a typical grounded theory study begins with a domain of interest. However, it does not begin with any specific research questions in mind or with respect to any prior work in the literature.

Grounded theory research begins with no clear goal or direction and the researcher(s) must not have a set agenda. It may appear odd to begin a project without a clear goal in mind or an awareness of what has already been done in an area but the intention of starting with a “blank slate” is to minimize bias to the extent possible. Grounded theory is for generating theory and not for confirming hunches or *a priori* assumptions. With too much background knowledge researchers may be tempted to verify their assumptions or to seek out specific sources of data. However, the generation of new theory requires the freedom to explore data, to identify the meaningful trends as they emerge, and to decide what the best use of the data actually is.

After a high-level topic is selected the next step in a grounded theory study is data collection and analysis. Data collection begins immediately and runs throughout the project. It takes place concurrently with a non-stop analysis process that is based on
constants comparisons of data with data (i.e. old data with newly collected data, data with the emerging theory, etc.). The entire data collection process is guided by a theoretical sampling technique in which what data to be collected next is determined by the data that a researcher already has. This implies that data collection methods may change over time\textsuperscript{13}. For example, if a trend emerges in the data then the researcher should try to find negative cases that could offer an alternative explanation or look for similar data that might have related, but slightly different, characteristics (often referred to as the dimensions of an emerging concept). A grounded theory study typically ends when this strategy “saturates” and the researcher is no longer able to find new data that is any different from the already collected data.

The outcome of a grounded theory study is the identification of a core “category.” This category corresponds to the most dominant social process that was observed\textsuperscript{14} and the theory explains how it was observed to function. For example, such a process might be dealing with stress at work or maintaining a leadership position in an online community. It can be challenging to arrive at a core category and to recognize when enough data has been collected because of the undirected nature of the entire process. Since clear boundaries are not established at the start of the study there is some subjectively in recognizing a “saturation” point of diminishing returns. This is one of the

---

\textsuperscript{13} This can lead to problems with institutional review boards.

\textsuperscript{14} Social processes are typically the focus in the version of grounded theory that was used here, but a study can be focused other unit(s) of analysis that are aligned with the research objectives.
most criticized aspects of grounded theory because it can be difficult to show that a resulting “theory” is credible (i.e. enough data was considered, appropriate sources of data were used, concepts and categories that were developed accurately represent the underlying data, etc.). As a final step, a review of relevant literature can be carried out to compare the newly developed theory to prior work in order to identify similarities and differences.

A common problem with doing any grounded theory study is deciding where to start, and this is the first concern that must be addressed in order to integrate the method within an intentional design process. Research questions are not identified and prior work is not reviewed at the beginning of a “pure” grounded theory study. In fact, some authors even advocate not discussing the research with anyone until a theory has been formed and the process is complete in order to limit the biasing effects of external influences (Charmaz, 2006). However, this is certainly not possible if grounded theory is to be used in the context of design research. Stakeholders’ problems must be considered at the beginning of the process and new designs should leverage the lessons learned through prior work. Design is not random and design research certainly cannot be as undirected as a pure grounded theory study might be.

The rationale for bringing grounded theory into a design context is so that it can contribute to long-term theory building through the short-term evaluation of prototypes. It addresses the common criticism that formative research methods may lead to “finding
what one is looking for,” particularly if those methods are qualitative. Fortunately, most grounded theorists recognize that no research project begins with a blank slate. For example, it is simply not possible for researchers to erase their interests and background knowledge prior to a new investigation (Charmaz, 2006; Corbin & Strauss, 2008; Glaser & Strauss, 1967) and design researchers are not any different in this respect. The key to applying grounded theory, in a design or any other context, is to ensure that there is enough freedom to shift the focus of data collection and analysis to the most interesting trends in the data however unexpected they may be. The procedures (discussed shortly) are also intended to help maintain that kind of exploratory, but empirical and objective, focus.

Evaluating prototypes can be seen as a form of theoretical sampling in the grounded theory method (also discussed shortly). It can be integrated into a generic process of building design patterns by treating the patterns as the grounded “theory.” In essence, researchers can experiment “in the small” with different prototypes based on the same logical pattern to facilitate a long-term “in the large” process of theory building. This process is shown below in Figure 19.

---

15 However, this concern is equally relevant to qualitative and quantitative forms of research (Creswell, 2008; Eisenhardt, 1989; Yin, 2008).
For example, the pattern that was presented in Chapter 3 is representative of an initial hypothesis that attempts to explain how technology can positively influence the strategic process of self-governance in small teams by creating an unobtrusive new channel for visibility. It was culled from an analysis of the literature and, at this point, it is still somewhat immature. There is an expectation that it will be incomplete or fail to account for relevant forces just as there is some hope that it might be a good place to start understanding the relationships between technology and self-governance. It is a starting point and the purpose of building and evaluating a prototype based on that pattern is to refine it and to identify any such weaknesses.

Again, the significance of grounded theory is that it can help further refine this pattern though systematic and objective data analysis. There is no clear hypothesis, or theory to be proven, about how a design based on the pattern actually works although
there was an explicit rationale for developing it. The goal is not to determine if this rationale was correct (it includes many assumptions and it is not yet worth the effort to verify). Instead, the goal is to build that missing theory of how it does actually work even if that theory will never reach full development in the pure scientific sense.

The reason for integrating design patterns with grounded theory is because the patterns decouple broader theoretical assumptions from concrete prototypes. In other words, patterns give additional structure to the grounded theory method. In a classic grounded theory study the researcher must determine what “kind” of theory to generate but this is not necessary in a design research context. Here, design patterns help structure the outcome of a study in a manner that is useful for the purposes of software design. This does not constrain the method because a generated theory can still be about anything. It merely helps to frame such theory in a well-known and useful format.

Similarly, the value of using grounded theory to move this larger pattern-building process along is twofold. First, it can help relate many similar designs and prototypes to a small set of relevant patterns. Most novel collaborative systems have similar characteristics and patterns provide a means for distillation and building a theoretically cohesive knowledgebase. More importantly, the use of grounded theory has implication for how the evaluation of prototypes should be carried out (i.e. between each dashed line in Figure 19). The goal of a design evaluation is no longer to validate the prototype and confirm a priori assumptions. Instead, the focus is now on the pattern and the constant comparison of data so that the effects of design can be further understood in practice.
The desired outcome of an evaluation in this kind of method is to produce some data that contradicts or deepens the theory embodied by the pattern being studied (i.e. looking for alternative or richer explanations of how it works). It is to disconfirm and restate rather than to confirm and verify assumptions. In comparison to the way most designs are evaluated if the a priori assumptions are confirmed then the study yields no new knowledge. Under the grounded theory approach evaluations that produced “expected” results would contribute to establishing a point of theoretical saturation or provide motivation for using other research methods (i.e. to verify a mature theory). Such results should begin to occur more often as the pattern becomes a more refined heuristic over time. But, it is the partially unexpected results that fuel the construction of theory and the development of better design heuristics.

This logical process of developing design patterns with grounded theory is illustrated below in Figure 20. Note the three elements of design patterns (from section 4.2) in the figure and their correspondence to elements in Figure 18.
Figure 20: The pattern-based methodology

The darker box in Figure 20 represents the boundaries of the pattern-based methodology. The focus is centered on refining three elements, the problem, the structure of the solution, and the forces. The “evaluation” of prototypes within the grounded theory method leads to the construction of a collection of substantive theories that explain how the pattern was observed to function when it was applied in particular settings. Over time, researchers can reconcile these isolated theories into a coherent design pattern that embodies the key lessons learned. This process of continuous
refinement and “evaluation” through rigorous empirical observation is the essence of the method. The process is not much different from a pure grounded theory study, but the structure of a “theory” is partially fixed because it is being applied to generate knowledge explicitly for the purposes of design. Note the loss of the arrow that would connect this process to the steps necessary to develop formal theory. That is beyond the scope of design patterns and it is not the purpose of design research. Formal theory is applied in domains like CSCW but it is not developed in them.

4.3.2 The method of constant comparisons

In the next few sections, some of the key procedures of grounded theory are discussed. They all contribute to its qualities as a method of using constant comparisons to generate theory and to build conceptual frameworks. There are other peripheral techniques available within the method but the characteristics discussed below are essential to the method.

4.3.2.1 Theoretical sampling

Theoretical sampling was already introduced as a central concept in grounded theory and it is where a ground theory research process begins. It differs from random and statistical forms of sampling because data is collected based on theoretical interest and the current state of the research process. A researcher begins a project by collecting an initial sample of data and then immediately begins to analyze it. This initial analysis is used to determine what data to collect next and this process of concurrent data collection and analysis continues throughout the duration of the project.
Theoretical sampling allows the project to be steered by collecting data that is most likely to help further build the theory, and also so the boundaries of the project can be established after some time (recall that there is no clear goal at the beginning to guide the research so one eventually has to be established). The recommended sampling strategy is to seek out data that is likely to rival aspects of the theory in development, expand parts of it, or expose alternative points of view. As potentially important concepts are discovered in the data a researcher must actively look for locations to collect data to build on their understanding. Sometimes this means moving to different physical locations, but often it requires adjusting one’s data collection methods (i.e. using semi-structured interviews or swapping instruments).

Diagrams and constant note taking can help facilitate the theoretical sampling process (see memo writing below). For instance, Figure 21 below is a partial excerpt from a cluster diagram that was maintained during this research to facilitate the analysis that will be presented later in Chapter 5. The figure itself is not particularly useful for presenting any results because its role was as a stepping-stone that was used to get to them. Nonetheless, it is illustrative of a technique that can help guide the theoretical sampling process (i.e. sparse clusters helped highlight areas that needed more data or further analysis and dense clusters often became important concepts in the developed theory).
Figure 21: Visual clustering to aid data collection

Through concurrent analysis and a constant comparison of the available data to the theory in development a researcher can refocus their data collection efforts as they discover the most salient factors. There is some subjectivity in this process because there may be a large number of potentially interesting avenues to explore. Time and resource constraints will undoubtedly prevent all of them from being explored and, even if they did not, the goal is not to generate a broad unfocused theory. Eventually a grounded theory study does focus on a specific area of interest. Part of the process of theoretical sampling is analogous to defining the boundaries of a traditional research project. It is simply not done at the start of the process.

Delaying the setting of boundaries gives researchers an opportunity to let the data reveal interesting trends before boundaries are set. However, at some point a researcher will have to decide upon those boundaries just as they must in any other form of research.
The lack of clear boundaries at the start of the process can make it difficult to know when to stop collecting data, define the project scope, and end the project.

As noted, researchers need to recognize a point of theoretical saturation at which time they have gathered enough data to construct a useful theory but can no longer identify new sources of data that are likely to change that theory. This is easier to do in some cases than in others. For instance, here grounded theory is applied to the process of building design patterns through prototypes, but finding places to pilot and test prototypes requires significant resources. It is unlikely that a researcher would reach a saturation point after a single field study, or even a small collection of them. Identifying good design principles is a slow and incremental process and it is exactly why design patterns were selected as a stage-gate mechanism to facilitate an incremental theorizing process.

There are also external factors that influence an investigator’s ability to effectively use theoretical sampling. Some sources of data may be too resource intensive, inaccessible to researchers, or not transparent enough to facilitate proper data collection. Or, there may be only a few locations or points in time where data can be collected. Nonetheless, this is not as problematic as it might be in theory-testing research projects. When there is not enough data to focus on a potentially interesting topic in a grounded theory project a researcher should turn to the most interesting and available data and set the boundaries of their study with these considerations in mind. The goal is to find a good theory and not the particular theory. This is highly relevant in design-oriented
research. People can interact with technology in a variety of ways, anticipated or otherwise. Often, revealing small variations or partially unexpected patterns of behavior is the most valuable part of CSCW research. It is what separates CSCW from the less applied scientific fields that influence it. Theoretical sampling should be used to focus on emergent issues so that they can be explored through theoretical development, and so that the unexpected consequences of design are not ignored.

4.3.2.2 Data coding

Coding and labeling data is a process that all qualitative researchers use to make sense out of data and to relate it with higher-level concepts. In grounded theory coding begins as soon as data is available and continues until closure is reached. There is not a single “phase” of coding because coding and re-coding is one of the key activities for ensuring that credible theory is developed (discussed in more detail shortly). It must take place throughout the entire research process so that constant comparisons can be made as a study progresses. These comparisons are made explicitly though iterative coding “phases” that complement a parallel process of memo writing (discussed next). In other words, the purpose of coding in grounded theory serves two distinct purposes. It is a way to abstract data into more abstract theory, which is its role in all forms of qualitative research. However, it is also a key part of the continual analytic comparative “method” that differentiates grounded theory from other research strategies.

There are a number of coding strategies that grounded theorists can use depending on the nature of their research, but coding must always be a continual process. A
common strategy (and the one used here) is to use multiple coding techniques to build incrementally more general concepts from the raw data and to facilitate continuous theoretical sampling. For example, the three typical coding techniques in Strauss’s grounded theory are open, axial, and selective coding\textsuperscript{16} (Corbin & Strauss, 2008). Eventually all of these techniques are used concurrently within the overarching process of making constant comparisons, but research begins with an open coding “phase.”

Open coding is simply labeling data with more meaningful terms. Often this is done at a very fine level of detail (i.e. each line, or even each word, of a piece of data is given a label). There is no fixed codebook and items are often recoded over time as reflexive and constant comparisons are made among the data and as a researcher’s emerging theory develops. During this phase the goal of coding is microanalysis. It is not to label data in some structured way or to try and fit it within any kind of conceptual framework. The goal is to inspect the data at a very detailed level, focus on alternative interpretations of it, and to summarize it to an extent.

Open coding helps identify basic concepts in the data that can later be integrated to form higher-level categories. The fine level of detail also helps build these higher-level categories as they emerge (i.e. what properties do they have, is there any variation, etc.). Open coding can be very labor intensive since the goal is to “break apart” (Corbin & Strauss, 2008) data and consider what it means at a very granular level of detail.

\textsuperscript{16} This is specific to Corbin and Strauss's version of grounded theory and it is not representative of the others.
Axial coding is the next “phase” of the coding process and its role is to build relationships between data items and codes to make clusters of data more visible. As part of axial coding, open codes are sorted or clustered into higher-level logical categories that represent trends in the data and potential relationships between them. Hence, this phase of coding summarizes the data at yet a higher level, but the purpose is to move towards a single cohesive explanation of the data. It builds logical “axes” from the data. In comparison, the goal of open coding is to identify as many individual concepts as possible and to explore alternative interpretations of data.

The goal of axial coding is to reorganize all of the pieces into high-level categories and to separate the important concepts and categories from the less important ones. Axial coding also helps a researcher define the boundaries of the study since they were not set at the beginning of the project. In most research projects there will be issues that arise which could be explored if more resources were available. Hence, a secondary goal of axial coding is to plot out the axes of the study so that appropriate boundaries can be drawn and so that opportunities for future work can be separated from the focus of the current work.

In most cases, the most important “categories” that emerge through coding are labels for key social processes and their characteristics. Again, the purpose of axial coding is to identify the properties of each category so that relationships between them are exposed and they can be integrated into a coherent logical framework. For example, this includes determining under what conditions processes occur, what actions they might lead to, and
what the consequences tend to look like. Note that grounded theory studies do not have to focus on social process. However, it is commonly recommended that they do because a focus on a single core “process” gives the researcher a way to gauge the size of their project. Constraining the scope of a developing theory into a single “process” is a strategy that can help separate interesting data from the peripheral data. Of course, it would be difficult to capture all this information in simple “codes” so memo-writing activities are used to complement the coding process (discussed in the next section).

The final “phase” of coding is selective coding and it is more related to theoretical sampling than it is to abstraction. Data is first summarized through open coding and then partially grouped through axial coding but it is not yet organized into a single theory. Selective coding is the process of determining which core “category” to focus the study on and relating it to the other categories, which may require more data collection (note that selective coding means selecting codes and it is not usually another layer of assigning descriptive labels to data as axial coding is).

Eventually, selective coding leads to the identification of the core category that becomes the focal point of the theory\textsuperscript{17}. Things that are not related to the core will not be part of the developed theory so the core category sets the boundaries of the study. Identifying the core category is an iterative process and this is why all of the coding

\textsuperscript{17} The core category is intentionally singular, but a study does not have to be restricted to a single core category. However, a single core is common so that the theory is integrated and of a reasonable scope. Again, a single social process is often recommended as the basis of a core category in order to avoid developing overly complex and disconnected theories.
“phases” eventually begin to occur in parallel. Writing memos that document the reasons for making analytical decisions is critical for keeping this process as systematic and transparent as possible.

4.3.2.3 Memo writing

Memo writing is a complementary technique to coding that a researcher engages in throughout the entire data analysis process. Writing forces interpretations to be developed on paper rather than in an investigator’s mind. Memos are a mechanism for researchers to document their internal thought processes, as a means to explicitly compare data with their interpretations, and to document the rationale behind decisions. In grounded theory memos essentially become a source of data, and that data is subjected to the same analytic method of constant comparisons as all other data. Hence, it helps to make the interpretive process more transparent and it facilitates the continual process of comparing data against emerging theory and re-examining decisions that have been made.

Memos can also serve other purposes depending on the format of the study. For example, in a design context it is useful to observe how people interact with artifacts, but it is equally useful to have them explain why they acted the way that they did. Hence, memo writing can be integrated with periodic member checking activities to ensure that a researcher’s interpretations of some observations are consistent with those of the research subjects (this is a key component to any kind of rigorous case study research (Yin, 2008)). Similarly, the role of memos makes them sort of like a rough draft of a research
report. Sorting and selecting the most important memos after the core category has been identified is often the starting point for developing an integrated theory from all of the data that has been analyzed throughout the coding process.

4.3.2.4 Closure

In a classic grounded theory study deciding when to stop is not trivial because the boundaries of a study are not clearly defined in advance. There is no mechanical way to determine when one has enough data to produce a good theory about something, and there is an inherent risk of letting the scope of the project expand to an unmanageable degree. Fortunately, this is less of an issue here since design research is not purely observational. Design prototypes are used to get feedback and, as noted, they are a form of theoretical sampling for new data. However, design prototypes are not intended to live in the wild and they are not often used on a daily basis as part of people’s everyday lives. They are research instruments and they have a fairly short lifecycle (i.e. they are developed to get some feedback from a specific group of people at a particular place and time).

Grounded theory can be integrated with the prototyping by treating each “experiment” with a prototype as a discrete phase in the research process and as a single grounded theory study (refer back to Figure 19). In other words, after each experiment the data is examined, a new core category is identified, and that is used to update the next version of a design pattern. This process can repeat indefinitely but the bounds of a single “study” are tied to exactly one experiment with a prototype. The difference
between this and a classic grounded theory study is that the bounds of each data collection step are somewhat fixed in advance and at some point it may become impossible to collect more data to develop the theory. This presents some risk because it may not be clear if enough data was considered in each pilot study to yield a well-developed theory. The inclusion of design patterns as a common structure was intended to reduce this risk by allowing the results of multiple studies to be integrated with one another over time (and it is related to the credibility of grounded theory discussed next).

Discretizing the process not only makes it easier to identify clear stopping points, but it also uses grounded theory to its full effect. Treating each pilot study independently from an analytical point of view allows the research to focus on the most important emergent behaviors in each particular case rather than on the assumptions that may be present (i.e. a new core is identified each time). Design patterns are merely a way to stitch one grounded theory to another in order to sustain a longer-term process of incremental theory building.

4.3.3 Credibility

The criterion for establishing the credibility of qualitative forms of research is something that has never been agreed upon (Charmaz, 2006; Corbin & Strauss, 2008; Creswell, 2008; Eisenhardt, 1989; Glaser & Strauss, 1967; Yin, 2008c), and it is a valid concern for grounded theory researchers and CSCW in general. Throughout this discussion it has been implicitly argued that design research methods tend to lack credibility (forms of validity, reliability, etc.). Methods range from the highly
quantitative to the purely qualitative but they are applied in the context of understanding how technology interacts with humans and how it influences their social structures. This has significant consequences (Glaser & Strauss, 1967, p. 235):

Third, and most important, a great deal of sociological work, unlike research in physical science, never gets to the stage of rigorous demonstration because the social structures being studied are undergoing continuous change. Older structures frequently take on new dimensions before highly rigorous research can be accomplished. The changing of social structures means that a prime sociological task is the exploration – and sometimes the discovery – of emerging structures. Undue emphasis on being “scientific” is simply not reasonable in light of our need for discovery and exploration amid very considerable structural changes. [sic]

If the fluidity of social structures is troubling for sociologists how can technologists produce credible research? Designing new types of collaborative systems and putting them into practice amplifies an evolutionary effect that is already present. The subject of a study, some kind of socio-technical system, will evolve long before researchers can definitively “prove” much about it or its relative value. Cultures evolve without intervention and they will certainly react to intentional technological changes. Credibility in the traditional scientific sense is relatively meaningless in design fields like CSCW because credible theory cannot survive for long. A credible theory built from reliable instruments and verified through valid inferences would have to be tightly focused on a particular social structure, and eventually that structure would simply cease to exist.

However, that does not mean that credibility should not be a concern. Design research is still in need of objectivity. When Glasser and Strauss developed grounded theory they identified four criteria that were necessary for producing a “good” theory.
First, they noted that a credible theory must *fit* the data and the audience for which it was intended\(^\text{18}\). They also believed that it must be readily *understandable* by people familiar with the topic and *general* enough to allow them to apply it in new areas, an artifact of pragmatism. But, they were also careful to note that unlike formal scientific theory a grounded theory should afford *control* to others. By control they meant that other researchers should be able to expand, revise, or otherwise modify the theory to adapt to new circumstances and incorporate new data to deal with the reality of cultural fluidity.

A grounded theory should achieve some degree of credibility by following the procedures that were outlined above, but merely following them is not necessarily sufficient to reach the four qualities that Glasser and Strauss associate with good theory. The procedures of grounded theory primarily address the first criterion of fit, or accuracy, of developed theory by limiting forms of interpretive bias (i.e. constant comparisons though coding, writing, and sampling). The remaining qualities are more subjective and they relate to how well a theory will *survive* the test of time.

Good theory is not only plausible, but it is also malleable enough to serve as the foundation for incrementally improved theory to facilitate a longer-term and abductive progression of research. If social structures do indeed change then approximate theory that can be integrated with new findings is likely to be more useful than exacting, but brittle, theory that shatters the moment people change. Design patterns complement the

\(^{18}\) The authors philosophically disagreed on how to best establish “fit” through the procedures of grounded theory, which lead to each one of them later developing their own version.
grounded theory method by structuring the outcomes of CSCW research in a format that has, to a greater degree, these additional qualities of good theory.

Finally, it is worth returning to the issue of what CSCW researchers do regardless of their methods. Do they ever prove or verify theories about collaboration and technology? In reality they do not. When qualitative studies are used to investigate design the results are often presented under the guise of “preliminary” but they are more often than not the end product of the research. Again, the fluid nature of society makes further effort unwarranted. Similarly, quantitative CSCW researchers do not apply their methods in the same spirit that experimental scientists use them in other fields where non-transient theory can be built. Rather, they import these methods because of their association with “good” science and misapply them in a way that Glasser and Strauss identified below (Glaser & Strauss, 1967, p. 235):

In sum, quantitative data are often used not for rigorous demonstration of theory but as another way to discover more theory; and qualitative data often result in a de facto conclusive analysis rather than a preliminary one. [sic]

Developing credible theory in CSCW can include both quantitative and qualitative data, but it is unlikely that credibility can be demonstrated as explicitly as it can be in other fields. Grounded theory is philosophically centered on minimizing bias in the interpretive process and that is the largest threat to design researchers. Design is an intentional process and since designs cannot be empirically proven the only scientific way to study them is to limit sources of interpretive bias and acknowledge that design
theory is approximate and constantly evolving. Design patterns complement the grounded theory technique by giving some additional structure to research results so that they are understandable by designers and so that they may be remolded to survive the test of time.

4.4 Summary

This chapter introduced a hybrid methodology for generating design patterns with the grounded theory method. The goal of this method is to facilitate long-term and incremental heuristic building in CSCW forms of research, and address the lack of attention that heuristics and theory building techniques have received in existing research methods.
CHAPTER 5: RESULTS

This chapter presents the results of applying the methodology that was developed in the previous chapter to the micro-modeling design pattern (Chapter 3). Recall that the purpose of this methodology is to facilitate theory building and to refine a design pattern. It is explicitly not intended to verify that the BlackTie prototype caused any expected patterns of behavior (see section 4.1 for a discussion of the problems with such an approach). Instead, it is intended to capture how the subjects actually behaved and to theorize about how the design of the system caused that behavior.

5.1 Context of the study

Finding a suitable location to conduct the research was a necessary first step. It can be challenging to find appropriate research sites to explore new forms of collaborative technology because care must be taken to ensure that the research objectives do not conflict with the goals of the community that will pilot the technology. Consequently, applied research is often done in laboratory settings or in the field at locations where costs can be controlled (as noted in Chapter 4). Here, the research advisor’s lab was

\footnote{For instance, this research was partially motivated by an organization that was searching for the kind of collaborative technology that was developed here. However, it became clear early in the process that this would not be a suitable location for any substantial research.}
selected for the context of the study because of these reasons. A local site allowed the risks associated with technology adoption to be managed, and it allowed timely technical support and assistance to be provided to the research subjects. A few additional local sites were considered for the study but the lab was selected because it was the only accessible location that also had two other desirable characteristics.

First, the number of students that worked in the lab was comparatively large at approximately thirty graduate students. The majority of these students had been given assignments as part of a group project spanning a full academic term, which became the basis of the study. The large project size would give those involved more time to use the prototype, reflect on its role in their work practices, and potentially provide more feedback than the other sites would have. It also allowed the study to be divided into a multiple segments through an embedded case study design. In other words, the work had been divided in such a way that the prototype could be piloted by a few different groups of students within the same environment at different times (with some overlap among group members). Finally, the larger project size made it easier to accommodate individuals that did not want to voluntarily participate in the research activities (i.e. did not wish to be interviewed but were still required to complete their assigned work).

Second, the work that the students had been asked to do had more desirable characteristics than the work to be performed at the other sites, and the schedule of that work was much more flexible. The degree of flexibility was particularly desirable because it reduced the risk of harm to all parities if the overhead of any of the research
activities had an effect on the project. For example, the schedule of the project was not fixed and that made it easier to integrate the additional research activities into the actual project work without disturbing it.

5.2 Nature of the work

The study overlapped with a “playbook” project that the directors had been preparing for because of the lab’s recent growth in size (it was not originally related to this research). The playbook was envisioned to be a collection of reference materials that students would need in order to work in the lab. This would include various kinds of information about the lab, its day-to-day operations, and the roles of students as developing researchers and leaders. For example, new students typically do not know whom to contact to get the resources that they need for their projects. Similarly, most of them do not know how to productively work towards a thesis because they have never done so before. The playbook would contain this kind of information and, ideally, it would reduce the amount of time that the lab directors spent answering questions or putting students in touch with one another.

The content was imagined to range from basic factual knowledge, like how to manage lab assets, to anecdotal reports and advice, such as how to write a research paper or a master’s thesis. The directors expected the students to work together collaboratively to assemble the playbook, and this style of ad-hoc, distributed, and creative work created an opportunity to integrate this research into the project. Most importantly, a secondary purpose of the project was to begin to bring about a more student-driven culture in the
lab. This meant getting students to see beyond their immediate tactical needs, like their own projects, and to engage in more strategic activities for the collective benefit of everyone in the lab (i.e. peer-review, mentoring, etc.). That orientation made the project a particularly relevant context for exploring the micro-modeling pattern in practice.

The content and format of the playbook was not precisely specified at the beginning of the project. Similarly, the project was not a typical part of the students’ work practices and it was the first time that such a project had been done in the lab. It was expected to be an iterative project, primarily managed and peer-reviewed by the students, and subject to occasional critique by the directors. However, it was impossible to know how the work would actually occur because there was little precedent.

The project began when the directors gave the students a list of topics to develop as “chapters” of the playbook. Each chapter was developed by a subset of students and subsequently presented during one of the lab’s regularly scheduled meetings. For each chapter, the directors gave the students a high-level description of the kind of content to be included with a bulleted list of “roles.” These roles corresponded to roles that members of the lab might play, like “scientific researcher,” “project manager,” and “special event coordinator.” The students were expected to write each chapter by describing these roles, which included collecting any relevant information that a person would need to play that role.

For example, the scientific researchers role was eventually written to include a brief description of how the lab generally conducts research, tips for new students, and links to
references on academic paper writing. Other roles, like special event coordinator, were written in a less narrative style and simply contained information about events that the lab periodically hosts and what has to be done to prepare for them (i.e. resources to purchase, where to buy them, timeline of critical events, etc.).

The lab directors initially assigned students to chapters based on student interest and their knowledge of various topics, but the students were expected to reassign themselves as necessary. The students were also given a PowerPoint template to use for creating each chapter at the beginning of the process.

5.3 Participants

A set of five cases embedded within the playbook project was selected for the study. In each case the students had approximately two weeks to complete their work and were asked to take part in follow-up research activities during the next one to two weeks. Cases A, B, C, and D each correspond to a single chapter of the playbook. Case Z corresponds to a wrap-up phase of the project where all of the completed chapters were consolidated and reorganized on a wiki system. The five cases were completed in order and one at a time. Some additional chapters of the playbook were completed during the project but were not included in the study for various reasons (i.e. lack of available resources, comparatively small chapters, etc.). The first case, A, was selected as a “pilot” case because the work was expected to be atypically simple in comparison with the others.
Fourteen graduate students and one undergraduate student voluntarily consented to participate in total. Ten were male, five were female, and twelve were international students. All but one had little to no professional work experience outside of a university setting, but many had been or were currently working on internships or projects within industry. Most of the students worked on only one chapter but some had multiple assignments that resulted in three students participating in more than one case. One student was included in cases A, C, and Z, another was in both A and C, and the third was in D and Z (but dropped out of Z). The table below shows the number of students that participated in each case.

<table>
<thead>
<tr>
<th>Case</th>
<th>Consenting Male/Female</th>
<th>In a previous case</th>
<th>Dropouts</th>
<th>Non-consenting</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>4/0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>C</td>
<td>4/0</td>
<td>2 (both from A)</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>D</td>
<td>2/3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Z</td>
<td>3/2</td>
<td>2 (1 from A, 1 from C; 1 from D)</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 1: Case study subjects

Students that did not offer their consent to participate in the study were still required to complete their project assignments, but no data was collected about them. Note that the number of non-consenting students is provided in order to get a sense of the total group size but it includes students that did not do any work, were dynamically reassigned, or had been assigned but could not be present for the work. Interviews with the students
suggested that there were approximately four students outside of the study that did work in case C. In the other cases the non-consenting students were rarely, if ever, mentioned and it is unclear if they played any significant role in the work (it is unlikely that they did but they were not asked directly because they had either not consented or were absent).

5.4 Data & operational procedures

The same operational procedures were followed for all of the cases, with one noted exception in case Z. A case study protocol was created for each case that followed a standard template in order to manage the logistics of each case (see Appendix A). The following subsections summarize the basic procedures that were used and the data that was collected.

5.4.1 Orientation meeting

At the beginning of each case, an in-person group meeting was held with all the students that had been assigned to work on that phase of the project. At this meeting, all of the students were given a brief five to ten minute introduction to the research study. They were told that it was independent, voluntary, and in addition to their normal duties on the project.

After answering any questions, all of the group members were asked for consent, via a form given in Appendix F, which stated all of the research activities, estimated time commitments, and the compensation for participating (a $20 gift card). This included taking part in one individual and one group exit interview, doing an on-line survey, using a prototype software system (BlackTie) as a part of their work, and keeping a brief
journal of their work related activities. They were given examples of the kinds of survey questions that would be asked as well as sample journal entries (see Appendix E). One of the lab directors was available to the students before this meeting to talk with the students about their assignments. However, they were asked to leave before the consent process to reduce the risk of intimidation.20

After the consent process any students that did not wish to participate were told that they could leave the meeting at their discretion (which did happen in case C). The remainder of the meeting time was used to give the students a tutorial on how to use the BlackTie software. The students were allowed to ask questions and were given an electronic copy of the training materials. The training portion of the session typically took about 45 minutes. As part of the training the students were repeatedly encouraged to use the software even if it seemed unfamiliar or of limited use to their particular work assignment and told that their role was to evaluate the software and to act as “beta-testers.” In case Z this tutorial was not given and the students were asked to get training from with their peers that had previous experience with BlackTie (they were told who had this experience).

5.4.2 Remote observation

After the orientation meeting, the students completed their project assignments away from the investigator and at their discretion. The BlackTie system was instrumented to

20 Similarly, the lab directors were never allowed to see any of the data from the study until it had been de-identified.
record and log important events during this time. Students were allowed to contact the researchers or lab directors during this period if they had problems or questions. The investigator documented and responded to all questions and problems but avoided commenting on any individual’s work, or work practices, in either a positive or negative way in order to maintain the role as a passive observer. Questions about the quality of work, how to work together, and the like were redirected to the lab directors. Throughout the study the researcher and the lab directors kept their roles as distinct as possible.

5.4.3 Individual exit interview and exit survey

After a stopping criterion was reached all of the students were asked, via email, to complete an online survey and to schedule an exit interview with the investigator at their convenience. If a response was not received within a few days the investigator sent an additional email with a suggested meeting time for the interview and a reminder to take the survey. After two such follow-up attempts any students that did not respond were considered dropouts of the study and no further communication was pursued.

The online survey was administered anonymously, via Survey Monkey, and consisted of ordinal and free text responses. The survey questions can be found in Appendix B.

The individual exit interview consisted of six open-ended questions. The first three questions were general in nature while the last three were targeted to the role of the
BlackTie system in the group’s work practices. All interviews were recorded and transcribed for analysis\textsuperscript{21}.

\textbf{5.4.4 Group validation session}

After all of the exit interviews were completed for a case a list of items was prepared to validate and fact-check with all of the group members of that case (part of the coding and memo writing process discussed in section 4.3). The list typically included items like any identified leadership roles within the group, the division of labor, overall satisfaction/dissatisfaction with the work experience, and any critical incidents. This list was discussed in a final in-person group meeting when possible. In one case multiple in-person meetings were used because a common time for all members could not be established quickly, and in another case the list was discussed through email because the academic term had ended and some students were no longer present. (Names in these lists were made anonymous with the exception of identified leadership roles).

\textbf{5.5 Case analysis}

The following sections present an overview of the outcomes of each of the individual cases. They are factual and descriptive in order to provide the reader with the appropriate context and to relive what occurred to the extent possible. Later, a cross case analysis is given to present the interpretive findings of the research.

\textsuperscript{21} There were two exceptions. In once case, the recording was lost due to a hardware failure. In another case, the exit interview was conducted through multiple emails because the student had graduated and arrangements for a face-to-face meeting could not be made in a timely fashion.
5.5.1 Case A

Case A was selected as a pilot case. Its primary purpose was to give the investigator an opportunity to test all of the research procedures and to identify unexpected problems that might put the project at risk. It was selected as the pilot because the work was expected to be simple and require much less collaboration than the other cases. Specifically, the task was to document some of the procedural responsibilities that one senior member of the lab held, such as how to book conference rooms and prepare for special events. One individual held virtually all of the knowledge that needed to be captured so the remaining students that had been assigned had few opportunities to contribute. These students confirmed that they did help format and review the content to the extent that they could after the chapter was completed, and that the senior student led the team and did the bulk of the work.

This pilot case did not reveal any major problems with the study. However, a number of minor issues were raised with some of the training materials and the BlackTie prototype. Most of these issues were related to inconsistent or vague terms that were used either in the training slides or the prototype. Members of the group also identified a number of minor usability improvements that could be made to the prototype before continuing with the other cases. For example, the way drag and drop operations behaved when an object was dragged off the edge of the screen was improved based on this feedback.
The group made a clear effort to integrate BlackTie into their work practices despite the simplicity of their task. The two graphs below show how often group members used it to view or edit their personal view of the process (i.e. to do work) or to view the process from the perspective of other team members (i.e. to see what others were doing). Each event corresponds to one session with the group’s “process” in the tool. A session corresponds to a file open operation, followed by any number of actions, and ends with a file close (either explicitly or when the browser is closed or refreshed). If users created other processes in BlackTie for demonstration or testing purposes they are not included here. Events that were clearly trivial, such as logging in but not performing any useful actions, have been excluded from these charts and all similar charts that follow.

![Personal View Graph](image)

Figure 22: Case A - Personal view accesses
The usage pattern of the first graph correlated closely with the journals and interviews from each of the group members. The team members spent a little time planning how they would work on the first day (during an in-person meeting) and then completed the work over the next few days. There was a pause after this initial period of work and then some final cleanup and review work was done the following week (mostly by A and B working in a co-located fashion). Subject B was the explicit leader of the group, did the majority of the content development, and had the most interaction with the tool. The second graph shows that BlackTie was occasionally used to view the work from another person’s perspective, but it was reported to be out of novelty and did not have any recognizable influence on the work that was performed.
BlackTie was incorporated into their work process mostly in a planning and project management capacity. Although infrequent, status updates and coordination took place through face-to-face discussions and a few emails. The group leader expressed some initial unfamiliarity with the tool but stated that it fit well with his or her normal planning routine. However, this unfamiliarly did result in some repeated effort. First, the group planned their work on paper and then the group leader mapped it into the tool, as noted in the quote below.

So, I wasn't too certain how beneficial that was, but it laid out what we were going to do and allowed me to organize my thoughts on, okay, how are we going to divide this up, which we did on paper before we started since we were not too certain about the tool to begin with.

BlackTie did not play much of a role in coordination of the work, but after the group’s first meeting there was not much communication of any kind regardless of medium. Subject A noted this during an interview:

We only all met together that one time at the very start. So, after that there really wasn't any communication between the group other than me and [the group’s leader] because we are in [physical] proximity with each other.

All group members were complacent with their work performance and reported it as either a positive or neutral experience. But, the disconnected nature of the group and the unbalanced workload lead to unmet expectations and a least-effort mode of working. Subjects A and B, respectively, noted this during their interviews below:

I would have preferred them [remaining team members] to do more. It wasn't really a negative experience; it was just annoying because I like
having feedback. I don't mind doing all the work as long as I get feedback but there wasn't even feedback from anyone other than B.

I was happy with what A did and another teammate did what was expected of him except he could have probably helped out in the slides, preparing the slides. Same thing with the rest of the guys, they could have at least helped A out on that particular aspect of it. I was glad I didn't have to do it.

This notion of not “having” to do something is an important aspect of both this case and the others. Recall that the directors’ intention of the playbook project was as a means to bring about an active student driven culture. A key component of that desired behavioral change is a kind of self-directed and peer-reviewed network of students that would allow individual students to benefit from the knowledge of their peers and reduce the reliance on the directors. Yet, that was not really occurring here (although the bland nature of this particular case made it difficult for the other students to contribute).

It is worth pausing to emphasize this aspect of all the cases. Putting the appropriate work practices and technology in place to implement such a change is what made this project relevant to the goals of this research. Implementing such a change requires establishing a social process of governance that shifts the balance of power between tactical and strategic concerns, and technology plays a partial role in that endeavor. In this case the BlackTie prototype did not play a useful role in implementing such a change. Nonetheless, understanding how technology can be designed to do so is the purpose of further building the micro-modeling pattern. Furthermore, implementing such
a change goes beyond technological factors and that is why a theory building methodology was selected.

There was one additional noteworthy observation that occurred during the pilot case. The group’s leader used BlackTie in a partially unexpected way. First, he or she planned out all of the “activities” in advance to break down the work down into smaller subtasks. This was expected because of the training that had been given. However, he or she then assigned everyone to all of these “activities” instead of reflecting the actual work assignments in the tool. The leader’s view of the process is shown below in Figure 24 as it was after the group had completed their work. It was originally broken down into about five subtasks that were distributed throughout the group (another team member added the “final” consolidation activity at the top left and the related rule arrows sometime later).
The leader stated that he or she chose to do it his way because there was an expectation that everyone would review all the content that was developed and, as such, needed exposure to all of the information. Using BlackTie’s activities to distribute pieces of work and recollect it in a shared “review” activity would have had a similar effect (and is exactly how another student altered the original process), but to this person there was no explicit review milestone. From his or her perspective, reviewing was something that should always be taking place and review work was something that everyone should be involved with. As a result, the distributed nature of the diagrams in BlackTie was more awkward than it was useful. The leader wanted a singular “view” of the work and
BlackTie’s model of a distributed and potentially non-overlapping view of the work was unnecessary. For example, Figure 25 below shows another teammate’s view of this process.

Figure 25: Case A - Another person's view
The figure above looks quite different from Figure 24, but the only actual difference is the position of the boxes. All of the activities and the information that they contain are the same. The ability to arrange them differently neither contributes to the kind of singular view that was desired by the group’s leader nor reduces the amount of information that other team members need to process. Naturally, the reason for allowing the diagrams to be distributed was to reduce the amount of information that users would be exposed to if they were working on separate tasks without totally disconnecting them from the rest of the work “process.” However, in this case the group leader did note that he or she was not certain how to use BlackTie in the context of the group’s work. So this mode of use was at least partially attributable to his or her process of learning to use a new tool. Later in case C, this person noted that their perception of a “process” changed.

5.5.2 Case B

Case B was dropped from the study immediately after the orientation meeting due to a low consent rate. Less than half of the students that were going to be working on this phase of the playbook project wanted to participate in the research activities. This low rate would not have given the investigator adequate control over technology use or sufficient access to the individuals performing the work. The students that had consented were told that the study would not continue for their group and were asked to complete their work as usual.
5.5.3 Case C

Case C was a chapter of the playbook that primarily included how-to and best practice types of information for graduate students. The topics were related to general strategies for academic writing, analyzing research reports, adhering to professional ethics, and conducting research projects. As with all the cases the lab directors made initial assignments to key topics that needed to be included. Unlike case A, these topics were not dependent on the background knowledge of any particular student. A comparatively large number of students were assigned to this chapter because it had the potential to be one of the most comprehensive chapters in terms of content. There is plenty of material readily available on the subject and, more importantly, the students had the opportunity to incorporate one another’s past experiences into the chapter.

Only about half of the students in case C agreed to participate in the research component of the project. Out of the four, two of these students were the more active students from case A. Six other students declined to participate and it is estimated that four of those six actually did some amount of work. Those that did participate in the research activities were unable to convince the others to use BlackTie as part of their work, which split the group in some sense. Nonetheless, this sort of imbalance is not uncommon in practice and made case C a useful, but unexpected, variation. Unlike case B, there was an expectation that some of the non-consenting students would not actually be a part of the work and that the consenting students would have dominant roles so case C was deemed useful for the research despite the split.
The work began much like it did in case A with an initial in-person planning meeting. This similarity was not surprising since two of the students had also been a part of case A and one of them took on an explicit leadership role that steered the group in a similar direction (although it was not the same individual that acted as the leader in case A). The initial meeting was used to discuss the topics to be included, how to best address them based on the interests of those that were present, and to divide the work. The meeting was not well attended by the group as noted by the group’s leader in the following journal entry:

[DATE OF ENTRY]:

No one has shown up for the meeting. [PERSON 1] acknowledged that s/he couldn't make it. [PERSON 2] called and said s/he would be late. We have not heard from anyone else. [PERSON 3] eventually showed up – s/he was stuck on south campus.

[PERSON 4] and [PERSON 5] showed up about an hour late. We met and had a whiteboard session brainstorming everything that we thought should go in each section. It was a pretty productive session and we met for about an hour and a half. The plan is to integrate everything into BlackTie and force the others to use the system, as it's helpful for organization. [ONE ADDITIONAL PERSON WAS PRESENT FOR THIS MEETING ALTHOUGH HE/SHE WAS NOT MENTIONED HERE]

The following day the group leader sent an email to everyone with the details of their assignments and instructed everyone to coordinate their work with BlackTie. The usage logs are given below in Figure 26 and Figure 27.
Figure 26: Case C – Personal view accesses

Figure 27: Case C - Other's view accesses
The logs correspond to the interview and journal data and indicate that two students, A and C, were the only team members that incorporated the tool into their work (A was the group leader and C was the student that had also worked with A in the previous case). The remaining two students only used the system to store their completed materials. However, these two students spent so little time actually doing their work that it would have been surprising if they had found a further use for it. Student B noted that their entire assignment consisted of about five (short) pages of text that was completed in one sitting (co-located with D) and a brief in-person review by C. Student C reported that BlackTie was more useful than it was in the previous case because it sent out an automated email reminder to him or her when student A made the initial assignments, and because it appeared to make the work more “organized.”

Once again, BlackTie was primarily used as a planning mechanism. Status updates and other work related forms of communication were infrequent and most often took place during ad-hoc face-to-face meetings when the students ran into each other in the lab. One of the students, C, noted this disconnected atmosphere during an interview as follows:

I mean, nobody complained or anything and they got the work done. The downside of it was that they did not [um] readily volunteer for stuff. So finally [A] just made assignments and everybody did the assignments s/he made. Which was fine and they got them done pretty much timely. But, there wasn't a lot of back and forth. You know, last time we worked together [and] this time it was like four independent, or four or five independent, efforts here that did it, you know. So that was strange.
In comparison, the task in case A did not require much effort and the lack of communication was not perceived as unusual even if it was somewhat frustrating to a few members of the group. It was appropriate for the work. However, in this case students A and C showed a different kind of recognition of the lack of effort by the others students. Yet, they also both reported being generally happy with the outcome. This seemed somewhat contradictory. However, all three students that were interviewed (A, B, and C; D was unable to be interviewed within a reasonable timeframe) expressed a similar interpretation of this uneasiness with the dynamics of their work. They felt what they had done was “good enough” to get them through the chapter and gain the approval of the lab directors, but not necessarily representative of high quality work by their own standards. Two of the students noted surprise or disagreement with the lab director’s expectations of the project. One of them wrote the following entry in their journal:

[date of entry]:

We received the slides from [Person 6] and [Person 7] a half an hour before [their scheduled presentation of their work] was supposed to start. They're good slides, but don't match the format. I think we should do major revisions to them before we actually release them. Again, this comes down to the format issue. For slides, they're great, but for actually communicating to new students without having to present information to them formally, the structure is less than ideal. I would like to see the [...] playbook redone or polished or implemented as a "continual integration" kind of project, with the end product being a set of wiki pages or a printable document.

Another expressed a similar sentiment during an interview:

But [lab director] was happy, which was surprising because [s/he's the person] that laid out the role responsibility form that we were all following
[laughs]. That's interesting. And we did have [lab director] review it this time too.

(Investigator): When did you have [lab director] review it?

I think [...], [A] sent it to [lab director] on Wednesday, and [s/he] commented Wednesday evening and said "innovation, innovation, innovation." So um, then we went back through and we added an innovation section, which hopefully it made [him/her] happy.

Finally, another student openly expressed dissatisfaction with his or her own work (this was in the context of a suggestion by the student to have been given three or four weeks for their assignment instead of the two that were given since the last week coincided with the last week of an academic term):

(Investigator): So, I can definitely understand that doing it during finals week isn't ideal, but do you think you would have actually needed three or four weeks? Because you said the amount of work you actually had to do was not that big.

Yeah, it wasn't that big but because I wasn't satisfied with what I did.

(Investigator): Oh, you weren't happy with what you did?

No, not very happy because it was stuffed into the slides and [...]

(Investigator): So, could you explain that a little more? Why weren't you happy with what you did? Was it just kind of a rush thing?

Yeah, rushed.

(Investigator): Okay.

Because usually when I do something I want to do it very well...
In a broad sense, BlackTie did not have much of an impact on the group’s work in either a positive or a negative sense. It did support the planning process of the students that incorporated it into their work practices, but not in any way that was clearly better or worse than other modes of collaboration (i.e. with other tools or more familiar patterns of working). They noted that the “visual” and “organized” aspect of it had some appeal, and potential to facilitate more “complex” tasks, but could not relate those qualities to the work that they were asked to do. Figure 28 and Figure 29 below show the process from the point of view of students C and A, respectively, after the work had been completed. The additional complexity of student A’s process reflects their role as a project leader.
Figure 28: Case C - Student C's view of the process
5.5.4 Case D

Case D corresponded to a playbook chapter that was similar in content to the previous case (C). The purpose of it was to assemble reference materials that would help students work on team projects with an emphasis on project management responsibilities. Hence, it provided a similar opportunity for integrating external reference material with the lessons learned by the current generation of lab students during their project
engagements. For this reason, most of the students that were assigned to this chapter by the directors had been in the lab for at least a year and had held significant roles on research projects (with the exception of one new student).

The directors expected this to be one of the more significant chapters of the playbook so six students were assigned to work on it. None of the students in case D had participated in any of the previous cases, although a couple of students would have also been in case B had it been used for the study. Unfortunately, the students did not make an effort to integrate BlackTie into their work practices despite the instructions that they were given in the orientation session. But, the factors that influenced this and the group’s general pattern of behavior are still relevant.

One of the most notable differences between case D and the previous cases was that no explicit leader took charge of the group. Like the previous groups, the work began with an in-person meeting that was used to discuss the topics to be included and to assign them to group members (one student reported themselves as being absent from this meeting). The assignments were given in pairs with the intention that one person would do the work and the other would primarily be responsible for reviewing it. The content of the chapter was incrementally integrated into a consolidated format with what the group called a “snowball” approach. As each student finished the topic that they had been assigned they would email it to everyone for review and their work would end up in subsequent versions of the chapter that others sent out to the entire group.
The snowball approach worked relatively well for the content and the work was completed on time, with one exception. Two students unknowingly developed content for the same topic and created duplicate materials. This was not discovered until moments before the group was going to present their work because it had not yet been “snowballed” (one student’s work was discarded to resolve the issue). Other than these exchanges, communication was infrequent between either members of the group or between pairs that had been assigned to work together.

A few of the group members did report briefly commenting on their teammates work, suggesting minor revisions, or reminding a teammate about a deadline. Student C was relatively active in the group and expressed his or her perception of the group’s general mode of working as follows:

Yeah, like the whole idea was that we'll have like a primary person and a secondary person. So, the primary person will do most of the stuff and might ask help from the secondary person, and the rest of the group everyone will review that work. But, that didn't happen. The plan was that but because, I mean, we presented [the results to everyone in the lab] out late and even though I sent mine really far [in advance of the deadline] I didn't get a reply. I think [A] was my second one, I don’t remember, and I was second for someone else and they didn't ask me anything.

(Investigator): Uh huh.

So, they kind of sent it late so I didn't bother to look at it either. So, I didn't and no one else did.

(Investigator): So, when you said that you guys all wanted to be responsible for reviewing it how did that actually work? [Described the snowball approach mentioned by another student] So, did you ever comment on anybody's work? Did anybody comment on yours?
Yeah, I didn't even look at them.

BlackTie was not used in any meaningful way by any of the students but most of them did attempt to use it at least once. It was used in a superficial manner that was described by one of the students (E) as follows, and later confirmed with the entire group:

(Investigator): So, when you say in parallel do you mean [pause]. What, did you do the same things through email and BlackTie? What is it you mean by that?

Uh yeah, I sent, for example, if I upload a file like a “.ppt” on BlackTie I also sent those files [by email] to other members in case they don't use BlackTie.

(Investigator): Okay, so you had the assumption that they weren't going to be using BlackTie to share files?

Yeah, I think some of them are not going to use BlackTie.

(Investigator): Now, how did you know that some of them were not going to do that?

I don't know because I don't know if they are [going to] use BlackTie. So, I assume some of them don't use BlackTie.

Some of the other students in the group echoed this lack of confidence in the system. However, members of the group should have had some reason to believe that their teammates would be using BlackTie. They were all told as a group during the orientation session that they should use it. Moreover, they had been asked not to participate unless they were willing to try and use it despite the possible overhead. Their superficial mode of using the tool was consistent among group members and it appeared to be deliberate.
It is unclear if this was due to an explicit decision that was made during their initial group meeting (i.e. to duplicate effort) or if was because of a lack of consensus (i.e. they knew to use the tool but did not agree or know how to do so). The reason for this lack of confidence could not be unanimously confirmed.

The log shown below in Figure 30 coincides with the how the group reported using BlackTie throughout their work. It was primarily used to store the artifacts that each member of the group produced, but not to share these artifacts with others or to retrieve them at a later time. Hence, most of these events correspond to times when group members finished their own work, saved the files in BlackTie, and concurrently sent them to the rest of the group via email.
The students’ use of BlackTie was trivial with respect to their work, but it was used in a manner that was consistent with the training that they had been given. Initially, one student created a small set of activities and assigned them to relevant students. Later, a few other activities were created by other students and assigned to teammates as necessary. The process from the point of view of student E at the end of the chapter is reproduced below in Figure 31.
The other students had similar views of the process, but they contained only the activities relevant to them (often just one or two). The students did not appear to struggle with the learning curve despite having no prior experience with BlackTie. They simply did not use it to drive the work and did not attribute much value to it.

There was no group leader but two of the students (E and C) took on management responsibilities at critical times. Student E was very involved at the beginning and played a large role in defining the assignments and setting up the group’s process in BlackTie. His or her initial pattern of behavior probably had a large influence on how the rest of the group chose to use the tool. Shortly before expressing a lack of confidence
that others would use the tool E also stated, explicitly, that he or she had no genuine intention of making of an effort to use it:

Generally, it's ok but for the task I think it would be better to just use e-mail.

\textit{(Investigator): And why do you think that?}

Because our task is not very big. I took your survey and I did mention that maybe for bigger projects the BlackTie may be better because when you need to manage like hundreds of files that's a better way to organize that, but for small tasks, like I just have one document, and have to go through the process like creating an activity, create work products, that seems to be too much for me if I can just simply attach a file in e-mail that may be much faster and easier for me to do that.

This attitude is a good example of the shorter-term tactically oriented focus that users of technology prefer. Level of effort is a component of usability. Yet, the style of visibility that BlackTie provided may have prevented the error that occurred during this case. But, the perceived overhead that this person associated with using it left any such potential unrealized. The technology failed, not because it was too complex, but because it was perceived as too “big” for the job. BlackTie was designed to be quick to use and most users reported it as being so but this perception of “big” remained.

This comment is also a nice example of the dichotomy between the need to manage and the need to do. From a single person’s perspective their task is never that “big,” but from an organization’s perspective it may be quite large. For instance, consider how many individual files were produced during the entire playbook project. There were approximately fifty-eight files in BlackTie after all of the cases (a small but indeterminate
number of which may be irrelevant), and there are many others that were only shared through email or other means (this study did not include all of the chapters and all of the work that was a part of the project). Months after the project ended the lab directors were not in possession of all the artifacts that had been created, the work had not been totally completed (see case Z), and no one held the responsibility for aggregating all of the content. It was a “big” project with a lot of small chunks of work.

Another interesting element of this group’s work was their federated style of management. At the start of the project the lab directors had suggested that groups have a leader to interface with them, but they did not mandate it and each group was given the freedom to manage itself. When asked why there was no explicit leadership in this case the students stated that there was no need. Students, namely A and C, temporarily took charge of the group when necessary. However, when asked about the group’s disconnected mode of working student C attributed the lack of leadership to something else:

I mean nobody wanted that extra responsibility.
Furthermore, he or she was very direct about expressing feelings of uncertainty about the value of the project as a whole and how the directors had been involved throughout the project. For instance, in a journal entry student C discussed being somewhat frustrated after the initial group meeting because they were not in total agreement with the other students about what topics to cover and how to break them down.

Though I'm in my second year, I don't know a whole lot because I never really needed to know all those [different functions or roles people have in the lab] before. I think this was also the case with our previous chapter [case B] I did not have a clue what a [role] does or [another role] does? May be having [lab director] also in BlackTie would have helped. :)

Later, in the interview this student went on to explain that he or she did not feel the work was very important, but that

But [lab director] was satisfied with whatever slides that we did, so [...] This sense of indifference was similar to that expressed by many of the students in the previous case (C). The playbook project may have had the potential to help new students understand the functions of the lab and foster a more student driven culture, but it failed to reach that potential as practiced.

5.5.5 Case Z

Case Z was substantially different from all of the previous cases. Cases A, B, C, and D were based on assembling content from various sources on a variety of topics identified by the directors of the lab. In contrast, case Z was the final phase of the project and the goal was to refine, consolidate, and archive all of that previously developed
content at an easily accessible location where it could be maintained. After some discussion, the directors asked a number of students to assemble this material on a wiki system that was already in place at the lab.

Five students took part in this phase of the project and they all consented to the research activities. The additional students listed in Table 1 as not having consented is a result of making ad-hoc assignments. He or she had been invited to participate in the project but chose not to be a part of it. Out of the five students that did participate, one had participated in cases A and C, and another had participated in case D (but eventually withdrew).

The characteristics of this phase of the project were different and it was expected that BlackTie would not be as relevant. The students had been asked to do the work on a wiki system and it would be much simpler to use it for online collaboration than an additional interface would be. Similarly, the work was not as distributed as it had been in the previous cases because there was no new creative content to develop.

Deciding how to structure and reformat the existing content on the wiki was the only thoughtful component of the work. This dialog took place during an initial in-person brainstorming meeting much like those that had been used to break down the work in previous cases. After this general structure was agreed upon the work was distributed but each individual task required only relatively menial labor. The students may have wanted to double-check each other’s work for consistency but there was little else that they
needed to do. This phase of the project was not much more than a formatting task. It was not a creative or knowledge driven process as the others had been.

The students were given the same introduction to the research and asked to use BlackTie, but because of the variation in task a change to the research procedures was made. No training session was given. Instead, the students were asked to consult with their teammates that had experience with the tool at their discretion in order to get acquainted with the system and decide how they would use it as a group. The intention of this change was so that the case could be focused on how the more experienced users would direct the other students to use BlackTie, and how their explicit directions would influence the group’s behavior.

Shortly after the assignments were made one of the more experienced students (C) met with two students (D, E) to give them instructions on how to use BlackTie (it is unclear if the remaining novice user received any training from his or her teammates). A few days later a single in-person group meeting was held where the structure of the wiki content was discussed and individual assignments were given. There was very little collaboration beyond that point and all of the students expressed some form of uncertainty with the state of affairs at the end of the project. Everyone felt that they had done their own part, but all of the students either felt the work was incomplete or were totally unaware of how it had come together. For instance, one of the students stated the following:
(Investigator): So, your part is done. What about the whole effort? Are you aware of what's going on with that and how the other [pause] how it's all coming together?

Uh, no. I didn't care too much about it but I think I need to check.

The students did use BlackTie as a part of their work, but the way in which the more experienced students gave the novice users instructions was far from expected. Student C reported training these students how to use the wiki in relation to the group’s collective task, but went on to explain BlackTie in an out of context manner:

[...] showing them how to use BlackTie I did not tell them here's how I've used BlackTie in the past so here's how I think we should use it. I said, here is what it is, here's what everything is made of, figure out how you want to use it. Like think about how you want to use it to see if they came up with a better way to use it than I have.

This would not have been surprising if student C had no experience with BlackTie, but he or she had used it in two prior cases and had been one of the most active users. Similarly, he or she had been using it in expected and consistent ways. The intention of giving experienced students the opportunity to “train” the others was to provide them with an opportunity to tell others exactly how they were expected to behave. In the previous cases some students, including this one, noted that they felt as if they had little power to direct their fellow students. This style of “training” was intended to give the experienced students some additional leverage, but this student avoided using it in such a manner. As student C continued to recall how the group worked together a recurrent theme of disconnection appeared that put it into context:
Once we got the work assignments it was pretty much autonomous. Like, I told them well if you guys have questions or concerns about anything let us know and we'll help you work through it but that never happened.

[later during the interview]

I was expecting there to be communication that ended up not being there.

(Investigator): Did you ever try to initiate that kind of conversation?

No, because I with [a class] I've been completely swamped with trying to get that working and so I didn't really have the brain power and time to devote to doing a really good job on this. So, I didn't try to prompt the discussion at all.

Another student noted a similar attitude:

(Investigator): So, you said there should have been a team leader. Why didn't you ask? You know, you could have stepped up. Right?

Yeah.

(Investigator): Any reason why you didn't want to do that?

Because I'm so busy.

As with many of the previous cases, there was a general acknowledgement that the quality of work could have been better if more collective effort and exchange had taken place. But, each student had a justification for relegating such responsibility at the expense of the group’s collective output.

The students used BlackTie in a style that was similar to how it had been used in the previous cases and it became a repository for the artifacts of completed work. Some students noted its potential to increase the visibility across the group, but this potential
did not affect the group’s behavior in any noticeable way. For instance, one student noted:

Everyone was [unclear, but repeating a previous statement about how everyone was given individual assignments]. And then we used the BlackTie and so [teammate] uploads the PowerPoint files into BlackTie system and then we know the progress of everybody.

This mode of using BlackTie was consistent but counter-productive in this case. Many of the students reformatted the content that they had been assigned and put it into BlackTie rather than on the wiki system. In other words, it essentially gave them a mechanism to by-pass their actual work. At one point, student C noticed that the materials were being added to BlackTie and not put on the wiki and sent emails to the other members of the group suggesting that they move this content to the wiki. The other students did eventually move some of the content to the wiki, but the quality of their work was noticeably poor. Student C noted this in the final entries of his or her journal:

[date of entry: 1 day before the deadline]:

More pages have gone up on the wiki after I sent an email with my finished results. I noticed that some pages don't have content from the playbook slides, which I thought was interesting.

[date of entry: final day of project]:

Not all of the pages are up -- notably, the Events pages are missing entirely -- but I think we're as done as we're going to get. I'm not satisfied with our final result, but we're out of time to work on it.

Figure 32 and Figure 33 below show the access logs from BlackTie, which shows pattern of use that is consistent with the tool being used in an archival function.
Figure 32: Case Z - Personal view accesses
Most students accessed BlackTie infrequently to store their individual work artifacts. Student C used the tool more frequently to keep their “process” in sync with their own work and, to a lesser extent, to monitor the progress of the materials (eventually realizing that effort had been misdirected from the wiki). Note that student A was not assigned to this phase of the project, although he or she did access the group’s process (it is unclear why). Also, the other student that had prior experience with BlackTie did not use it at all in this case (this was the student that stated he or she did not want to use it in case D).

Student C’s view of the process at the end of the project is shown below in Figure 34 and the other students’ views of the process were similar. All shared one activity (top left) where most of the playbook content was eventually stored. Only student C created
additional activities to break down their work, and the remaining students simply added their content to one of the work products associated with the shared activity.

Figure 34: Case Z - Student C's view
5.6 Cross case analysis

A number of common trends were observed throughout the study. The style of the work varied a bit in each group but overall it followed a fairly predictable and consistent pattern. Similarly, the groups all used BlackTie as a part of their work in a relatively consistent fashion. Case D was a bit of an exception because there was not a genuine effort to use BlackTie, but the superficial style of use that they followed was not radically different from the other groups. In addition, the students in all of the groups consistently reported the BlackTie prototype as being simple and intuitive to use, but of no noticeable value to their own work (the questionnaire results of the closed-end part of the survey can be found in Appendix C). It was less familiar but essentially served the same functional purpose as email. So, what does this experience say about BlackTie and its underlying design principles?

As the study progressed a key category emerged through the interview, journal, and open-ended questionnaire data. Initially, the focus of the research had been on visibility and how technology might create new channels for it without introducing extra effort or creating unwelcome accountability. This was the motivation behind the micro-modeling pattern and the rational for developing BlackTie. Yet, it became clear through the data that any relation between visibility and accountability was not a major concern to the students involved. The students cooperatively made explicit assignments and ensured that they were completed with or without any intervening technology. In at least two cases they did so without the authority of a recognized group leader.
But, the students never engaged in the kinds of highly interactive and collaborative modes of working that the directors had hoped for, and the features in BlackTie that were intended to support this kind of interaction went largely unused. Over the course of the study the focus gradually shifted and the process that prevented it from occurring emerged. It was not directly related to visibility or accountability, but it did help to frame those concepts with respect to how the students worked. This central process was a sort of “typical” mode of working that the labs students associated with their “student culture” (many students discussed this typical way of working directly). It explains their work habits and the role that technology played in them throughout the study, and can be called a process of maintaining relegation over delegation.

In effect, the students employed a strategy where they would relegate work when possible (i.e. dismiss or take no action upon), but occasionally delegate chunks of it to themselves or others and take action when certain conditions were reached (often by chance). For instance, all of the students can be thought of as having been at an initial baseline state where they knew that there was work to be done (i.e. the playbook as defined by the lab directors). However, while at this baseline state, a student would not perform any actual work to operationalize that shared goal. First, something had to be explicitly delegated to them. Once that tipping point was reached and a student was given an explicit assignment they would do it, often to the letter. They did not avoid any of the individual responsibilities that they were given. However, they would rarely go above or beyond that minimum level of adequacy and re-delegate new chunks of work.
For example, they generally did not seek out feedback on sections of the playbook content in development, get verification on information that was unclear or in need of expert examination/advice, clarify their assignment when the initial intention became unclear, or engage in the similar kinds of activities that are necessary for completing knowledge intensive forms of work.

Instead, they would return to that initial level of baseline inactivity and lie awaiting their next assignment. Occasionally, events would occur that would encourage some additional delegation to happen and lead to action. For instance, students would run into each other in the lab and an exchange and review of their work to date would take place. Or, their work had to be handed over to someone as a part of their task, which created an opportunity for exchange and explicitly making new commitments. Similarly, the initial kick-off meetings that most groups held lead to an associated initial burst of activity. Nonetheless, by following the relegation over delegation strategy the students tried to avoid exactly these kinds of opportunities (i.e. minimizing communication, delaying hand-offs, preferring low-bandwidth communication channels and short messages).

Leadership and personal motivation were not immune from this effect. For example, students were often aware of deficiencies in their own work or their teammate’s work, but they rarely took corrective actions. Group leaders, or even temporarily assertive individuals, ensured that defined tasks were completed but they avoided critiquing completed work in ways that would set new objectives. They were suggestive but did not delegate. Some of the students were not very interested in the project, but few were lazy.
or unwilling to do their part. With a few exceptions, students completed the work that they had been asked to do. However, they preferred to work in isolation because this mode of working supported their strategy of maintaining relegation over delegation. They did what they thought was reasonable given their explicit instructions and avoided the circumstances that might lead to a change in those instructions.

This passive attitude was dominant but not present at all times. Occasionally, students proposed innovative calls to action without being prompted. However, it was infrequent, sporadic, and it usually went unrewarded. The students were equal peers and the further any call to action was from the original instructions that they had been given by the directors the less credibility it carried. Hence, it was difficult for the students to delegate new responsibilities even when they wanted to do so (many students reported feeling powerless at various times during the project).

The trend of relegation over delegation dominated and it led to the observed pattern of working that started with a bang (lots of effort and enthusiasm) that quickly faded (very little effort after an initial meeting), much like a string of firecrackers that had been purposefully unwound prior to being lit. It is not that these students were irresponsible. They were simply self-motivated and more interested in their own immediate goals than the lab’s strategic mission. Some students in the playbook project were clearly enthusiastic, but their manner of working can be better described as doing what they personally wanted to do rather than what their group had been commissioned to do.
In this context, whether or not technology could influence constructs like shared awareness or visibility, often a focus of CSCW research, is almost totally irrelevant. There is a larger problem at hand. Using tools or implementing practices to build any kind of shared awareness would conflict with this perpetual process of maintaining relegation over delegation. It would not be welcome by the students, and they are not likely to fully leverage such tools or actively engage in such practices because it would contradict this dominant force that is a recognized part of their student culture. The process of *relegation over delegation* is a clear example of how the strategic and tactical needs of technology users interfere with one another. However, the BlackTie experiment indicates that creating new channels for visibility is not adequate to overcome this force even if the cost of that additional visibility is low.

### 5.6.1 From substantive theory to design theory

The students did not use BlackTie in its full capacity or attribute significant value to it in the context of their work. Yet, understanding how it related to this process of relegation over delegation yields insights into how the design could be improved to counteract such forces in the future.

The most interesting aspect of the students’ tendency to relegate was that although it was a dominant trend it was intermittently interrupted. For example, there were times when students did try to reach out to their peers in an attempt to improve the quality of their deliverables after the momentum of the initial group meeting had faded.
Unfortunately, these efforts were usually unsuccessful because a single student’s call to action fell on deaf ears and carried no authority.

This points to an opportunity for technology to support a kind of critical reaching out process that was also a part of the students’ larger pattern of working. The distributed aspects of the diagrams in BlackTie were intended to create a channel for reaching out (i.e. by allowing for individualized expression). However, as a passive channel it did not effectively support the entire reaching out process. It only supported the start of it, the initial call to action, and it did not help students follow through and eventually engage in constructive forms of deliberation. For instance, an anonymous student made a comment during the exit survey about how being able to view a process from one of their teammate’s point of view could be used to verify that someone was not actively responding to a call to action:

By viewing others' processes, it's easily apparent when someone isn't using the tool as it was intended, because activities and work products end up stacked on top of each other.

This student was referring to being able to see that his or her teammates were not using BlackTie to the extent that they were using it. One of the results of BlackTie’s process-centric design is that the diagrams need a small amount of routine maintenance. As new elements are added to the diagrams and shared they need to be acknowledged and occasionally visually rearranged (particularly when an activity is shared by one user causing it to appear in the top left corner of another user’s view of that process). Some students used BlackTie infrequently and as a result their view of the process would look
noticeably disorganized (i.e. overlapping boxes). This created a form of visibility but it did not help students get responses to their calls to action. After all, the person that was awaiting a response did not need to be made aware of the inaction yet again – it is the others that needed such prompting. Hence, that visibility could have been redirected to other places where it might have served more useful purposes.

For example, this “disorganization” is not difficult to detect in the diagrams. Such metadata could have been captured and displayed in various ways to encourage a response. Naturally, software cannot ensure a response takes place but it can create the conditions necessary for overcoming the dominant tendency to relegate responsibility. For instance, it could have been aggregated and displayed to an external supervisor that could have steered the group at critical moments (much like a student’s suggestion in case D). Or, it could have been displayed directly to the group via a dashboard or through periodic messages to support cycles of “coming back together” and resolving outstanding calls to action. During the playbook project students rarely avoided doing additional work when they became aware that it was needed. However, they did make a clear effort to avoid becoming aware of such situations.

One can imagine a number of similar techniques for combating this tendency to remain uninformed. For instance, the diagrams in BlackTie could have been strategically directed at particular people. In the BlackTie prototype that was developed collaborators could look at the diagrams of their peers but doing so was an extra step. They had to take the initiative and commit to an extra click of the mouse.

166
Nonetheless, there were opportunities to project those diagrams to others in ways that may not have created much, if any, extra work. For example, many of the initial screens had unused space that could have been used to show peripheral views of other users’ processes (i.e. login screen, initial navigation screen, loading/waiting screens, etc.). This kind of lightweight exposure might help avoid the costs associated with having to actively seek out another user’s process and may have had the potential to spark the process of reaching out.

Similarly, the diagrams could be explicitly directed at other users either by the system itself or by establishing complementary work practices. For example, agile software developers often use a “daily standup” meeting to discuss their plans for the day and such diagrams could be integrated into those kinds of practices with or without direct technological support.

In summary, this study provides some additional motivation for why the role of governance in collaborative tools needs research. It is not sufficient to design tools that support processes like sensemaking and building awareness if there are confounding social norms and processes that will limit their effectiveness in practice. This study revealed that designing for awareness is dependent on a willingness to be made aware, and constructed a theory about how university students develop this willingness. To the students in this study, maintaining an appropriate level of group awareness was to know that everyone had completed their assignments as given. Yet, this was not the kind of
awareness that the lab directors hoped the students would develop as part of the playbook project, and it is not the kind of awareness that can support highly effective teams.

5.7 Pattern revision

The results of the study revealed two related processes of relegation over delegation and reaching out that can be integrated into the original micro-modeling pattern first presented in Chapter 3. To keep the presentation brief the entire pattern will not be repeated here (much of it is unchanged). Instead, the revisions to each of the three components of the pattern are described below.

5.7.1 Problem

The problem has not changed. Social interaction is a key component to collaboration in physical and virtual spaces and it was clearly important throughout this study. Maintaining long-term forms of awareness within a constantly evolving social context is at the core of this pattern. However, the study revealed that what awareness means to a particular group and how forms of visibility will influence a group’s awareness depend on transient qualities of the group, such as their developmental history and cultural norms. Strategic governance mechanisms are necessary for establishing visibility across a group but care must be taken to ensure that such visibility can be usefully acted upon and that it is directed at the appropriate entity. BlackTie was successful in the sense that it added an additional layer of visibility without interfering with the immediate needs of users, but that visibility was not directed to a useful location.
5.7.2 Forces

Like the problem, the forces of this pattern have not changed dramatically. Leveraging the user interface as a mechanism to capture strategic process-centric metadata remains the key concern. The design of an interface determines the data that a user will input during their interaction with the system. In most cases there are many potential designs that could be followed to help the user accomplish a well-defined task. Care should be taken to ensure that it allows them to do so quickly and easily with respect to commonly accepted usability criteria. However, an interface should also be designed to capture as much contextual data as possible to the extent that it does not noticeably erode usability to the frustration of individual users.

5.7.3 Structure of the solution

The structure of the solution requires some significant revisions in light of the results. The general properties of a process-centric interface have not changed and still include (for details refer back to section 3.1.3):

* A visual language: For structuring a user’s interaction with the system around some high-level logical construct, like a work process or flowchart.

* A visual interlock: Leveraging the visual language as a means to accomplish a more direct task, such a sharing a document of piece of information, while capturing additional contextual metadata (i.e. not directly documenting a work process).

* A minimal overhead: Ensure that the use of a visual language as an interlock is not done in a way that noticeably undermines common tactically oriented usability concerns,
such as the time required to complete a task. The primary goal is still to support the user’s main task. Capturing additional information should be exploited to the extent possible but it cannot become a higher priority than accomplishing a user’s immediate goals.

A participatory community: A logical “process” is not a singular entity in reality and individuals may choose to imagine and model a logical work processes in different ways. Care must be taken to ensure that a process-centric design caters to their individualized notion of process.

What this framework is missing is a directionality component. Successful adoption of a software system only implies that users can carry out their immediate tasks. It does not imply that users will make use of any additional metadata that it may be capable of generating for more strategic purposes, like building shared forms awareness. This is the motivation for the following two additions to the pattern:

Adaptive visibility: The use of a visual language captures contextual metadata that can be redisplayed and/or used to compute additional metrics about how a group is working. This information must be directed to locations where it is most likely to be consumed and where it can be properly interpreted. For example, well-established groups may actively look for weakness in their operations while others may be more prone to overlooking or ignoring them at times. If users are unlikely to make direct use of the additional contextual data it should be hidden to avoid frustration and processing costs, or directed to only specific users that can make use of it such as leaders or
administrators. This may require multiple user interfaces that are each suitable for distinct roles within an organization, or adaptive techniques for dynamically adjusting the kinds of contextual information that is displayed to users.

For example, the “disorganized” metric in BlackTie could have been computed and made visible only to explicit group leaders or to external supervisors at the beginning of the project. These individuals would have a clear use for it because they would be working to steer the group and establish its working practices. Over time, the significance of an explicit leader may become less important and this metric might be more useful as a visual indicator to everyone in the group or, perhaps, a subset of the most active individuals.

A tightly coupled community: This pattern is intended for members with strong ties in a community because that is where deeper contextual metadata is meaningful. This kind of data is unlikely to be relevant on a larger scale, and this pattern should not be used to facilitate large-scale forms of collaboration. However, individuals with weak ties may need to occasionally interact with members of a more tightly coupled community through the system (i.e. infrequent users). These kinds of users will not be able to contribute any useful contextual information even if a visual interlock is present and they should have an alternate means to interact with the system.

For example, in the BlackTie study there were a few cases where students that were not involved in the project needed to provide some documents to other students. Email or other forms of document sharing were more appropriate for these kinds of exchanges.
because these students were not a part of the work process and they had no knowledge of the project team’s dynamics. They were resources, but not a part of the working community, and making the group’s actions more visible and directing additional metadata to them is not useful.

5.8 Summary

This chapter presented a grounded theory of university students’ experience with a prototype collaborative system based on the micro-modeling design pattern. The results indicated that the design failed to account for an important behavior shaping constraint in the environment (i.e. a desire to remain unaware), which helped to clarify the potential set of influences that designers need to consider when building tools that support a range of conflicting tactical and strategic functions (i.e. the pattern).

This study also showed how the general grounded theory method could be integrated into evaluative research methods. This theory-building research method should not be confused with design practices that can be used throughout a product development lifecycle, such as contextual inquires, persona development, or heuristic analysis, or with other evaluative methods (i.e. quantitative and mixed method hypothesis driven techniques). It is complementary to these practices and it could be combined with any of them. In comparison, the purpose of the methodology proposed here is to maximize the

22 However, if some of them are mixed it would be wise to do so with multiple investigators (i.e. researchers that are looking for specific patterns of behavior versus those that are trying to build an explanation of observed behaviors)
benefit of inherently non-replicable technology evaluations, or pilot studies, by building theories to explain observed behaviors rather than verifying assumptions or conclusions that were made earlier in the design process. Such a method is necessary in CSCW because of the idiosyncratic nature of many social processes, the lack of predictive sociological meta-theory, and the corresponding inability of designers to anticipate the potential range of responses that new technology might lead to in the wild.
CHAPTER 6: CONCLUSIONS & FUTURE WORK

Collaborative tools have the potential to enable complex service organizations to work more effectively. Intelligent people drive these organizations and technology has the power to enhance their ability to work together in pursuit of a common goal. However, there is a lack of portable design principles and heuristics that can be used to build collaborative software systems and successfully deploy them in practice. This research addressed this gap though two separate, but related, contributions.

First, the need for collaborative tools to support a conflicting set of tactical and strategic needs within organizations was identified as an area where design theory was lacking in CSCW. A novel micro-modeling design pattern was developed as a first attempt to build some of this missing heuristic knowledge and to work towards identifying reusable principles. This pattern was evaluated through a pilot study with a prototype that led to an improved version of the original pattern. The study revealed that the original design was of no greater value than comparable systems because it did not adequately account for the presence of a dominant and unexpected social process.

Secondly, this work addressed a deficiency in CSCW research methodologies. There are no formal methods for building heuristic forms of knowledge, and it was argued that the credibility of applying either deductive (theory-verifying) or purely observational
(theory-neutral) methods in design research is questionable. Reliable verification of theory is nearly impossible in CSCW because of idiosyncratic social behaviors and the amount of data that would be required to do so. In fact, the partially unexpected results that were found during this study are concrete examples of this problem in CSCW research. Unanticipated factors were relevant, which made the theory-building orientation of this study more appropriate than a hypothesis testing one would have been.

Similarly, pure observational studies do not advance the state of the art or adequately contribute to a body of engineering knowledge that is of wide general interest. To address this problem software design patterns were combined with the grounded theory method to form a novel theory-building research methodology. No significant changes were made to the basic canons or operational procedures of grounded theory, but there is novelty in applying it in an intentional design context.\textsuperscript{23}

The pilot study was consistent with previous research and demonstrated the challenges associated with implementing “strategic” functions in collaborative technology. When technology provides functions that do not support the immediate concerns of users they tend to be ignored despite any potential longer-term benefits. Restructuring the user interface to collect strategically oriented metadata was partially successful in the study (some users incorporated the diagrams into their practices), but the “directionality” of this metadata was not adequately considered (those that might have

\textsuperscript{23} To the best of my knowledge this kind of application has not been reported in the literature.
benefited from the diagrams chose not to look at them). This work also provides some evidence that a user interface can be intentionally used as a strategic data collection device without noticeably affecting perceived usability (supported by interview data and the survey results in Appendix C). However, more work is needed to develop effective design strategies for “directing” any collected metadata back to individuals in ways that would support their collective work practices.

There are two limitations of this work that warrant explicit consideration. Firstly, no attempt was made to evaluate the methodology that was developed. Carrying out any kind of rigorous comparative evaluation of the methodology would have been challenging because there is not a clear benchmark by which to gauge research methods. The motivation for developing the methodology was because CSCW research methods have been imported from scientific fields where proposing a theory and then empirically validating it is a genuinely useful activity. However, design principles are not equivalent to formal theory and such theories rarely predict the wide range of social and behavioral responses that new collaborative systems lead to. Hence, the method developed here was oriented towards building an explanation of how novel designs work in practice. It was not to quantify or verify the presence of an assumed explanation, which would be of little value if there were not much evidence to support the initial assumptions. These are two fundamentally different research goals and it is not trivial to compare them fairly to one another.
For example, had a traditional approach been taken here to evaluate the prototype would the results have been similar? It is likely that the inability of the system to radically change behavior would have been noticed, and perhaps even measured more explicitly. However, those indicators may have lacked accuracy and precision because unexpected influences, like the relegation process, may have been ignored. On the other hand, the critical factors may have been identified correctly a priori. The key point is that the complexity of social processes makes identifying all of the relevant factors with a high level of confidence very challenging in CSCW research. The method used here was intentionally built from the ground up to account for such unexpected influences. The unique aspect of this research is that an observational research method was employed in an evaluative capacity, and it would be hard to show that this is a universally better approach. Regardless, identifying ways to fairly compare this kind of a methodology to other research methods is a clear area for future work.

A second limitation of this work is that the design pattern was developed through a single iteration. The method was intended to be iterative but time and resource constraints permitted only one cycle. In some senses this makes the results of this study “preliminary.” However, CSCW is a field where heuristics and preliminary findings are the norm and this study is typical in that regard. The results from this research are not

24 This work has repeatedly argued against the idea of verifying that a design worked as intended because in CSCW studies unexpected, but relevant, factors are commonly identified. However, this is not always true and, as noted in an earlier chapter, there are cases where group work is regulated, consistent, or otherwise fixed enough to warrant the use of verification methods.
definitive, but they were also never intended to be. The study showcased a novel solution for a common problem in collaborative tool design, and it presented a useful critique of it in one practical real-life setting. Exploring the design pattern in alternative contexts would no doubt continue to improve the proposed design pattern but there is no obvious stopping criterion.

In conclusion, collaborative tools are becoming critical components in many types of organizations and they are particularly relevant to firms that offer complex and human-driven services. The balance of power in organizations makes designing these tools challenging because the needs of individuals or distinct social groups are often in conflict with one another. Users want tools that will make their work easier and stakeholders want tools that will make their organization stronger and more resilient. Identifying reusable strategies for successfully dealing with this trade-off is an important but underemphasized component of CSCW research. This research represents a first step towards systematically developing design heuristics in a manner that can simultaneously support the tactical and strategic needs of a technical society’s greatest asset - its people.
REFERENCES


186


APPENDIX A: CASE STUDY PROTOCOL

A.1 Subjects

The following potential subjects have been identified. If they agree to participate in the research the date that consent was obtained is noted below and their signed consent form is on file.

<table>
<thead>
<tr>
<th>ID</th>
<th>Name (until de-identified)</th>
<th>Email (until de-identified)</th>
<th>Status</th>
<th>Consented On</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A.2 Schedule & logistics

A.2.1 Initial meeting & training session

The initial meeting and training session has been scheduled for:

<table>
<thead>
<tr>
<th>Time</th>
<th>Place</th>
<th>Invited IDs</th>
<th>Attended IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A.2.2 Individual interview schedule

The individual exit interviews have been scheduled for:

<table>
<thead>
<tr>
<th>Subject ID</th>
<th>Time</th>
<th>Place</th>
<th>Journal (Y/N)</th>
<th>Completed (Y/N)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
A.2.3 Group interview & journal collection

The final group meeting has been scheduled for:

<table>
<thead>
<tr>
<th>Time</th>
<th>Place</th>
<th>Invited IDs</th>
<th>Attended IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A.2.4 Online survey collection

The survey administered for this case is:

<table>
<thead>
<tr>
<th>Survey URL</th>
<th>Password</th>
<th>Collector ID</th>
<th>Date Opened</th>
<th>Date Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A.3 Data collection procedures

At the site the following tasks need to be completed. Audio and video recording will be performed as noted below if the site permits it. Every time audio or video recording is used verbal consent to perform the recording should be obtained from all present parties.

1. If the subjects’ work is performed under the supervision of any management a meeting is held with them first to discuss the costs/benefits of the research and set the expectations of both parties. The study does not proceed without their unanimous verbal consent.

2. At a group meeting all of the subjects are given a brief introduction to the research and asked for consent, via a form.
3. The subjects are told about the intervention (BlackTie) and trained to use it. The subjects are reminded that they will be interviewed at some point in the future and required to keep a private journal.

4. The subjects will complete their work away from the investigators on their own. Before beginning a mutually agreed upon stopping criteria will be established to set the boundaries of the case. Journals will not be visible to the investigators at any point during this time. The subjects may contact the researchers if they have problems or questions during the study. The investigators will document problems and probe the subjects to provide as much detail as possible while taking care to avoid criticizing the subjects’ actions or rationale in any way.

5. When the stopping criteria are met the subjects will notify the investigators. The investigators should periodically check with subjects in case they forget to provide notification.

6. Subjects will be given a short online survey and then individually interviewed in person at the completion of the case study. A tentative interview schedule should be determined at the start of the data collection process and reevaluated as the study progresses.

7. The investigators will analyze the data and prepare findings. Subject may be asked for additional clarifications during this period.

8. The investigators will present their findings to the subjects at a group meeting and ask for discussion and verification of the results.
A.4 Reporting guidelines

Near the end of each case a group discussion session will take place where the investigators reveal their preliminary findings to the subjects within the case. The purpose of this session is threefold. First, it will help the investigators detect if they have misinterpreted data and enhance the reliability of the study. It will also yield some additional retrospective qualitative data. Finally, it will give the subjects an opportunity to raise any concerns about the results that will be shared.

The investigators will create a brief report in preparation for this meeting. This report should adhere to the following guidelines:

1. Individuals are not identified – Individuals should not be explicitly identified, just like they would not be in a published report. Of course, it is very likely that within such a small group individuals will be able to make inferences about one another fairly easily. Regardless, the investigators should avoid identification to maintain neutrality and, in some cases, to avoid response bias when seeking validation for a particular claim.

2. Key results are clear, concise, and bulleted – The preliminary results should be packaged in a discussion friendly format. The group session should not require the subjects to read a report before or during the session (although they may choose to do so). Consider the size of the group and the most interesting results when crafting the materials to be used for the session (i.e. slide deck).
3. Reiterate investigators’ role as researchers and designers, not stakeholders or sales people – The investigators should never appear to have a stake in the any positive or negative results (avoid non-neutral wordings).

4. Reiterate the preliminary nature of the findings – The subjects should be aware that this preliminary report is not intended to be shared with anyone outside the case, and that they will have access to all future reports or publications if they are interested.

5. Reiterate follow-up information – Remind the subjects how they can contact the investigators after the session is over.
APPENDIX B: ONLINE SURVEY QUESTIONS

1. How satisfied are you with the work performance of each item below during the study?
   a. Your OWN Performance
      • Very Satisfied (5) Satisfied (4) Neutral (3) Dissatisfied (2) Very Dissatisfied (1)
   b. Your TEAM's Performance
      • Very Satisfied (5) Satisfied (4) Neutral (3) Dissatisfied (2) Very Dissatisfied (1)

2. Explain how your team was successful or unsuccessful working together.
   • Open-Ended Response

3. How often did you use BlackTie as a part of your work?
   • Frequently (4) Sometimes (3) Rarely (2) Never (1)

4. Please explain why you did not use BlackTie as part of your work.
   • Open-Ended Response
5. How useful did you find BlackTie as a part of your work? - I found it
   - Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

6. How likely would you be to use BlackTie to do some similar work in the future? - I would be
   - Very Likely (5) Likely (4) Neutral (3) Unlikely (2) Very Unlikely (1)

7. Was there anything you really liked about BlackTie?
   - Open-Ended Response

8. What did you like least about BlackTie?
   - Open-Ended Response

9. How often did you use BlackTie to view the processes of others (using the People Bar at the top of the screen)?
   - Frequently (4) Sometimes (3) Rarely (2) Never (1)

10. With the People Bar, how useful was it to see the diagrams your teammates created?
    - It was
      - Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

201
11. Please explain why you liked or disliked viewing your teammates processes with the People Bar.

   • Open-Ended Response

12. When you were using the People Bar did you ever find any of your teammates diagrams surprising or confusing? Is so, please explain why.

   • Open-Ended Response

13. How often did you use BlackTie to coordinate with your teammates using ACTIVITIES rather than coordinating with them ONLY though email or some other means?

   • Frequently (4) Sometimes (3) Rarely (2) Never (1)

14. How useful were Activities to you? - They were

   • Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

15. Please explain why you liked or disliked Activities.

   • Open-Ended Response

16. How often did you use BlackTie to share documents or data (using WORK PRODUCTS) rather than sharing them ONLY through email or some other means?

   • Frequently (4) Sometimes (3) Rarely (2) Never (1)
17. How useful were Work Products? - They were

- Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

18. Please explain why you liked or disliked Work Products.

- Open-Ended Response

19. How often did you use BlackTie to create rules or reminders (using RULES)?

- Frequently (4) Sometimes (3) Rarely (2) Never (1)

20. How useful were Rules? - They were

- Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

21. Please explain why you liked or disliked Rules.

- Open-Ended Response

22. How often did you ignore or delete elements (Activities, Work Products, or Rules) when they were shared with others?

- Frequently (4) Sometimes (3) Rarely (2) Never (1)

23. How useful was ignoring or deleting elements? - It was

- Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)
24. Please explain why you thought ignoring or deleting elements was useful or not useful.

- Open-Ended Response

25. How useful do you think that the diagrams you and your team created in BlackTie would be if you

a. did similar work in the future?

- Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

b. had to explain your work to someone else not familiar with it?

- Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

c. wanted to maintain an ongoing awareness of what your team was doing?

- Very Useful (5) Useful (4) Neutral (3) Slightly Distracting (2) Very Distracting (1)

26. Any other comments or feedback that you would like to share?

- Open-Ended Response
## APPENDIX C: ONLINE SURVEY QUESTION RESULTS

<table>
<thead>
<tr>
<th>Question</th>
<th>Total Median (Mode)</th>
<th>A</th>
<th>C</th>
<th>D</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>How satisfied are you with the work performance of each item below during the study?</td>
<td>4 (4)</td>
<td>4 (4)</td>
<td>5 (5)</td>
<td>4 (4)</td>
<td>4 (4)</td>
</tr>
<tr>
<td>Your OWN Performance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How satisfied are you with the work performance of each item below during the study?</td>
<td>4 (4)</td>
<td>3 (x)</td>
<td>4 (4)</td>
<td>4 (4)</td>
<td>4 (4)</td>
</tr>
<tr>
<td>Your TEAM’s Performance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How often did you use BlackTie as a part of your work?</td>
<td>3 (3)</td>
<td>4 (4)</td>
<td>3 (2)</td>
<td>3 (3)</td>
<td>3 (3)</td>
</tr>
<tr>
<td>How useful did you find BlackTie as a part of your work? - I found it</td>
<td>4 (4)</td>
<td>4 (4)</td>
<td>4 (4)</td>
<td>3 (3)</td>
<td>4 (4)</td>
</tr>
<tr>
<td>How likely would you be to use BlackTie to do some similar work in the future? - I would be</td>
<td>3 (3)</td>
<td>3 (3)</td>
<td>4 (4)</td>
<td>3 (4)</td>
<td>4 (3)</td>
</tr>
<tr>
<td>How often did you use BlackTie to view the processes of others (using the People Bar at the top of the screen)?</td>
<td>3 (3)</td>
<td>3 (3)</td>
<td>3 (3)</td>
<td>2 (2)</td>
<td>3 (3)</td>
</tr>
<tr>
<td>With the People Bar, how useful was it to see the diagrams your teammates created? - It was</td>
<td>4 (4)</td>
<td>3 (x)</td>
<td>3.5 (4)</td>
<td>3 (3)</td>
<td>4 (5)</td>
</tr>
<tr>
<td>How often did you use BlackTie to coordinate with your teammates using ACTIVITIES rather than coordinating with them ONLY though email or some other means?</td>
<td>3 (3)</td>
<td>3 (3)</td>
<td>3.5 (4)</td>
<td>2 (2)</td>
<td>3 (3)</td>
</tr>
<tr>
<td>How useful were Activities to you? - They were</td>
<td>3 (4)</td>
<td>4 (4)</td>
<td>4 (4)</td>
<td>3 (3)</td>
<td>3.5 (4)</td>
</tr>
<tr>
<td>How often did you use BlackTie to share documents or data (using WORK PRODUCTS) rather than</td>
<td>3 (4)</td>
<td>4 (4)</td>
<td>4 (4)</td>
<td>2.5 (3)</td>
<td>3 (3)</td>
</tr>
<tr>
<td>Question</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>sharing them ONLY through email or some other means?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How useful were Work Products? - They were</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>How often did you use BlackTie to create rules or reminders (using RULES)?</td>
<td>2</td>
<td>2</td>
<td>1.5</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>How useful were Rules? - They were</td>
<td>3</td>
<td>4</td>
<td>2.5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>How often did you ignore or delete elements (Activities, Work Products, or Rules) when they were shared with others?</td>
<td>1.5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>How useful was ignoring or deleting elements? - It was</td>
<td>3.5</td>
<td>2</td>
<td>4</td>
<td>x</td>
<td>3</td>
</tr>
<tr>
<td>How useful do you think that the diagrams you and your team created in BlackTie would be if you did similar work in the future?</td>
<td>4</td>
<td>3</td>
<td>4.5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>How useful do you think that the diagrams you and your team created in BlackTie would be if you had to explain your work to someone else not familiar with it?</td>
<td>4</td>
<td>4</td>
<td>3.5</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>How useful do you think that the diagrams you and your team created in BlackTie would be if you wanted to maintain an ongoing awareness of what your team was doing?</td>
<td>4</td>
<td>4</td>
<td>3.5</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>
APPENDIX D: EXIT INTERVIEW QUESTIONS

A.1 General

1. Summarize your experience working with your colleagues over the past [timeframe]. Was it positive or negative and why?

2. How did this experience differ from your past experiences doing similar kinds of work?

3. Were there any surprising or particularly memorable events? Please describe them.

A.2 Targeted

4. Explain how you used the diagrams in BlackTie in your work. Did you find them useful or distracting? [Ask them to log into the system and show you]

5. How would you compare BlackTie to other collaborative systems, including email, which you have used in the past?

6. What would make BlackTie more useful to you?
APPENDIX E: EXAMPLE JOURNAL ENTRIES

A.1 Example 1

Asked Joe to get information on aardvark's dietary needs and I am awaiting his response.

Started searching for other animal diet information myself.

A.2 Example 2

Consolidated info about leaf cutter ants and edited Wikipedia entry. Worked alone.

A.3 Example 3

Meet with Joe & Rajiv to discuss cleaning the fish tank. Joe agreed to get in touch with OSU maintenance for help.

A.4 Example 4

Put together some info on puffer fish and sent it to Jay & Rajiv for approval. Meet with Joe to discuss possibility of getting more fish for the fish tank - he is not sure and will check OSU's fish policy.
The Ohio State University Consent to Participate in Research

Study Title: Case Study Based Exploration of a Computer Supported Cooperative Work Tool based on Individualized Process Modeling and Federated Process Representations

Researcher: Ramanathan, Jayashree

This is a consent form for research participation. It contains important information about this study and what to expect if you decide to participate.

Your participation is voluntary.

Please consider the information carefully. Feel free to ask questions before making your decision whether or not to participate. If you decide to participate, you will be asked to sign this form and will receive a copy of the form.

Purpose:

The purpose of this study is to observe the effects of a novel computer-based collaboration tool in a realistic work environment.

Procedures/Tasks:

You and your colleagues will be asked about the work you plan complete together and asked to use a new computer-based collaboration tool to help you coordinate that work. You will receive training for using the system and given remote access privileges during
a group meeting. Then you will be asked to leave the research site and complete your work as normal using this system. During your time away you will be asked to keep a journal of your work experience, and you will be allowed to contact the researchers if necessary (i.e. for technical support). When your work is complete, you should notify the researchers so that you can return your journal and participate in an exit interview. You will also be invited to a group session at the end of the study to learn about the results of the research.

Duration:

The duration of the study will depend on the kind of work you and your colleagues do. You are encouraged to discuss your time and resources concerns with the research team. At a minimum, you will be asked to participate in one 2-hour training and orientation session, one individual and one group interview session lasting approximately 2-4 hours in total, and to take a 20-minute online survey. Please be aware that you may leave the study at any time. If you decide to stop participating in the study, there will be no penalty to you, and you will not lose any benefits to which you are otherwise entitled. Your decision will not affect your future relationship with The Ohio State University.

Risks and Benefits:

The risks associated with this study are no greater than those in everyday life. This research will contribute documented cases that may be used to derive useful design guidelines for computer-based collaboration tools with similar/dissimilar characteristics relative to the one used in the study. As an exploratory evaluation of a novel system, it is also a necessary first step for future research efforts that may attempt to explain or predict specific behaviors or influences that arise as a result of using the system to coordinate work.

Confidentiality:

Efforts will be made to keep your study-related information confidential. However, there may be circumstances where this information must be released. For example, personal information regarding your participation in this study may be disclosed if required by state law. Also, your records may be reviewed by the following groups (as applicable to the research):

- Office for Human Research Protections or other federal, state, or international regulatory agencies;
- The Ohio State University Institutional Review Board or Office of Responsible Research Practices.
In addition, this study involves the use of online tools. Although every effort to protect confidentiality will be made, no guarantee of Internet security can be given as, although unlikely, transmissions can be intercepted and IP addresses can be identified.

**Incentives:**

You will be given at $10 gift card to a bookstore at the end of the study. If you complete and return a journal to the researchers at the end of the study your gift card will be upgraded to $20.

**Participant Rights:**

You may refuse to participate in this study without penalty or loss of benefits to which you are otherwise entitled. If you are a student or employee at Ohio State, your decision will not affect your grades or employment status.

If you choose to participate in the study, you may discontinue participation at any time without penalty or loss of benefits. By signing this form, you do not give up any personal legal rights you may have as a participant in this study.

**Contacts and Questions:**

For questions, concerns, or complaints about the study you may contact Jayashree Ramanathan (email: jayram@cse.ohio-state.edu, phone: REMOVED FOR PUBLICATION).

For questions about your rights as a participant in this study or to discuss other study-related concerns or complaints with someone who is not part of the research team, you may contact Ms. Sandra Meadows in the Office of Responsible Research Practices at 1-800-678-6251.

If you are injured as a result of participating in this study or for questions about a study-related injury, you may contact Jayashree Ramanathan (email: jayram@cse.ohio-state.edu, phone: REMOVED FOR PUBLICATION).

**Signing the consent form**

I have read (or someone has read to me) this form and I am aware that I am being asked to participate in a research study. I have had the opportunity to ask questions and have had them answered to my satisfaction. I voluntarily agree to participate in this study.
I am not giving up any legal rights by signing this form. I will be given a copy of this form.

<table>
<thead>
<tr>
<th>Printed name of subject</th>
<th>Signature of subject</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AM/PM</td>
</tr>
<tr>
<td>Date and time</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Printed name of person authorized to consent for subject (when applicable)</th>
<th>Signature of person authorized to consent for subject (when applicable)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AM/PM</td>
</tr>
<tr>
<td></td>
<td>Date and time</td>
</tr>
</tbody>
</table>

**Investigator/Research Staff**

I have explained the research to the participant or his/her representative before requesting the signature(s) above. There are no blanks in this document. A copy of this form has been given to the participant or his/her representative.

<table>
<thead>
<tr>
<th>Printed name of person obtaining consent</th>
<th>Signature of person obtaining consent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AM/PM</td>
</tr>
<tr>
<td>Date and time</td>
<td></td>
</tr>
</tbody>
</table>