MAKING SENSE OF CONFLICT IN DISTRIBUTED TEAMS:

A DESIGN SCIENCE APPROACH

A Dissertation in
Information Sciences and Technology

by

Guangxuan Zhang

© 2016 Guangxuan Zhang

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

December 2016
The dissertation of Guangxuan Zhang was reviewed and approved* by the following:

**Heng Xu**  
Associate Professor of Information Sciences and Technology  
Dissertation Advisor  
Chair of Committee

**Sencun Zhu**  
Associate Professor of Computer Science and Engineering & Information Sciences and Technology

**Zhenhui (Jessie) Li**  
Assistant Professor of Information Sciences and Technology

**Daniel Cardoso Llach**  
Assistant Professor of Architecture

**Andrea Tapia**  
Associate Professor of Information Sciences and Technology  
Director of Graduate Programs, College of Information Sciences and Technology

*Signatures are on file in the Graduate School
ABSTRACT

Conflict is a substantial, pervasive activity in team collaboration. It may arise because of differences in goals, differences in ways of working, or interpersonal dissonance. The specific focus for this research is the conflict in distributed teams. As opposed to traditional teams, participants of distributed teams are geographically dispersed and rely heavily on computer-mediated communication. Understanding and managing conflict is a crucial task in these settings because it is at the crux of many other managerial issues (e.g., trust, leadership, and knowledge transfer). Conflict in distributed teams hides in asynchronous communication among team members. While this causes extra burdens in detecting and understanding conflict, it also makes it possible to rebuild conflict scenarios for the purpose of making sense of, and managing conflict in these teams.

The objectives of this study are to make progress toward an initial design theory in the form of a novel conflict detection and analysis approach; to implement a faithful instantiation of this evolving theory; and to employ it to gain insight about the conflict phenomenon in the new setting. The work is organized and reported in three essays.

Essay 1 develops a meta design of the approach (CM2) and describes its implementation as a novel conflict management system. The design builds on a new construct – the Vignette (vivid yet analytical, theory-laden descriptions of past experiences) – that describes conflict situations. A meta-model developed on basis of prior kernel theories underpins the conflict vignettes (the proposed conflict management approach) and CM2 (the implementation). The essay also describes an initial empirical pilot investigation that serves as the impetus for the study, and demonstrates the anticipated use of CM2 with a detailed use scenario.

Essay 2 is aimed at automating the process of extracting conflict information from computer-mediated communication (CMC) data, such as emails and instant messages. The work focuses on the basic action in conflict – argument – and develops an automatic argument
detection solution. Drawing on the argumentation theory, I propose a model for argument
detection composed of features that reflect five categories of argumentation functions including:
announcement; reasoning; modality; transition; and, affect, along with another set of language
features that are informative for recognizing arguments. The evaluation results show that the
model achieves higher accuracy and recall in detecting arguments in message sets compared to
baseline models.

Essay 3 extends prior design efforts to explore the conflict phenomenon in distributed
settings. The work creates instruments to measure conflict elements as specified in the meta
model and based on automatically detected argument information. Following a grounded theory
approach, I employ the instruments to analyze 23,094 conflict situations in Bugzilla, an open
source software development community. The analysis results reveal patterns associated with the
occurrence of conflict, participants and their behaviors, and the temporal evolution of conflict. I
then interpret these findings with the help of prior work to develop theoretical propositions that
explain conflict in crowd collaboration settings that can provide directions for future research.
# TABLE OF CONTENTS

List of Figures ......................................................................................................................... viii

List of Tables ........................................................................................................................... ix

Acknowledgements .................................................................................................................. x

Chapter 1 Introduction and Motivation ......................................................................................... 1
  Research Background .............................................................................................................. 1
  Research Setting and Problem Statement .............................................................................. 2
    Distributed Teams ................................................................................................................. 3
  Conflict and Conflict Management ......................................................................................... 6
  IT Support in Conflict Management ....................................................................................... 8
  Research Objectives and Questions ....................................................................................... 11
  Structure of This Proposal ...................................................................................................... 12

Chapter 2 Literature Review ...................................................................................................... 13
  Conflict in Organizations ....................................................................................................... 13
    Defining Conflict .................................................................................................................... 14
    Conflict Antecedents ............................................................................................................ 15
    Conflict Outcomes ............................................................................................................... 17
  Managing Conflict in Organizations ....................................................................................... 19
    Conflict Process ..................................................................................................................... 19
    Conflict Management Strategies ......................................................................................... 23
    Conflict Management Actions .............................................................................................. 25
    Summary and Implications .................................................................................................... 27
  Conflict and Conflict Management in Distributed Teams ....................................................... 28
    Defining Distributed Teams .................................................................................................. 28
    Managerial Issues in Distributed Team .............................................................................. 30
    Conflict Antecedents in Distributed Teams .......................................................................... 34
    Conflict Outcomes in Distributed Teams .............................................................................. 38
    Managing Conflict in Distributed Teams ............................................................................. 40
    Summary ................................................................................................................................. 43
  IT Support for Conflict Management ....................................................................................... 44
    Communication-Driven Systems ......................................................................................... 44
    Decision-Driven Systems ...................................................................................................... 47
    Summary ................................................................................................................................. 50

Chapter 3 Research Method ....................................................................................................... 52
  Design Science Research ........................................................................................................ 52
  A Design Science Research Approach ................................................................................... 53
    Phase 1 - Problem Definition ............................................................................................... 56
    Phase 2 – Solution Design .................................................................................................... 59
    Phase 3 – Theory Formulation and Communication ............................................................ 60

Chapter 4 Essay 1 – CM2: A Conflict Management System ..................................................... 62
  Chapter Summary .................................................................................................................. 62
  Background and Motivation .................................................................................................... 62
Prior Work and Kernel Theories ................................................................. 64
Conflict and Conflict Management ............................................................ 64
Case-based Reasoning for Managing Conflicts .......................................... 66
Kernel Theories ....................................................................................... 68
A Foundation for the Design – Conflict Vignette ......................................... 69
Defining Vignettes .................................................................................... 69
A Meta-Model of Conflict Vignette ............................................................. 70
CM2: A System for Reasoning with Conflict Vignettes .................................. 72
Vignette Base ............................................................................................. 73
Vignette Extraction ..................................................................................... 74
Vignette Retrieval and Representation ......................................................... 75
An Instantiation of the CM2 System ............................................................ 75
Illustrative Example ................................................................................... 76
The Scenario without CM2 ......................................................................... 77
Re-visioning the Scenario with CM2 ............................................................ 77
Conclusion ................................................................................................ 79

Chapter 5 Essay 2 – Argument Detection in CMC: A Theory Based Approach .... 82
Chapter Summary ....................................................................................... 82
Background and Motivation ....................................................................... 82
Related Work ............................................................................................... 85
Argumentation Theory .............................................................................. 85
Argument Detection Models ..................................................................... 86
Design Framework ..................................................................................... 87
Argumentation Theory Based Features ....................................................... 88
Language Features ..................................................................................... 90
Experiments and Evaluation ..................................................................... 92
Dataset ......................................................................................................... 92
Feature Extraction and Adopting TAD Model ............................................. 93
Experiment Setting .................................................................................... 93
Evaluation Metrics .................................................................................... 94
Results ........................................................................................................ 94
An Illustrative Example ............................................................................ 96
Conclusion ................................................................................................ 98

Chapter 6 Essay 3 – Understanding Conflict in Crowd Collaboration: Conflict Measurements
Design and Application ............................................................................. 100
Chapter Summary ..................................................................................... 100
Background and Motivation ..................................................................... 100
Related Work .............................................................................................. 103
Research approach .................................................................................... 105
Site Selection ............................................................................................... 106
Data Capture ............................................................................................... 107
Argument Detection .................................................................................. 108
Conflict Repository .................................................................................... 108
Conflict Identification ............................................................................... 109
Analyses and Interpretations .................................................................... 109
Occurrence of Conflict .............................................................................. 110
Participants in Conflict ............................................................................. 113
Process of Conflict .................................................................................... 116
Interpretation and Theoretical Propositions .............................................. 119
Occurrence of Conflicts ................................................................. 119
Participants .................................................................................. 121
Conflict Process ........................................................................... 123
Conclusion ..................................................................................... 124
Chapter 7 Conclusion .................................................................... 126
Bibliography .................................................................................. 129
LIST OF FIGURES

Figure 2-1 A General Model of Conflict ................................................................. 15
Figure 2-2 Conflict Management Strategies ........................................................... 23
Figure 3-1 Methodological Overview of the Proposed Research ........................... 54
Figure 3-2 Organization of the Design Efforts ....................................................... 60
Figure 4-1 Design Problem, Kernel Theories, and Meta-Requirements ................. 68
Figure 4-2 A Meta-model for Conflict Vignette ..................................................... 72
Figure 4-3 System Architecture – CM2 ................................................................... 73
Figure 4-4 Screenshot of CM2 – Main Window with Search Results ..................... 76
Figure 4-5 Screenshot of CM2 – Conflict Vignette Details ................................. 79
Figure 5-1. Framework of Theory based Argument Detection Model ..................... 88
Figure 5-2. Visualizing an Argument, A Bug Discussion on Bugzilla .................... 97
Figure 6-1. Research Approach .............................................................................. 106
Figure 6-2. Number of Messages Generated every Month ...................................... 110
Figure 6-3. Plot of Peak Conflict Levels ................................................................. 111
Figure 6-4. Causes of Conflict .............................................................................. 112
Figure 6-5. Third Parties in Conflict Threads ....................................................... 113
Figure 6-6. Contender in Bug Threads .................................................................. 115
Figure 6-7. Collaborators in Bug Threads ............................................................... 115
Figure 6-8. Types of Participants in a Thread .......................................................... 115
Figure 6-9. Duration of Threads with Conflicts ..................................................... 117
Figure 6-10. Duration of Threads without Conflicts .............................................. 117
Figure 6-11. Escalation Levels across Thread Duration ......................................... 118
Figure 6-12. Ending Conflict Levels in Threads .................................................... 119
LIST OF TABLES

Table 1-1 Managerial Issues in Distributed Teams .......................................................... 4
Table 2-1 Definitions of Conflict ......................................................................................... 14
Table 2-2 Organizational Conflicts and Their Causes ......................................................... 16
Table 2-3 Conflict Outcomes ............................................................................................. 18
Table 2-4 Conflict Management Actions .......................................................................... 26
Table 2-5 Use of Communication-Driven Systems and Their Influences ....................... 45
Table 2-6 Existing Decision-Driven Systems .................................................................... 48
Table 3-1 Phases and Elements Suggested in Design Science Research Methods .......... 55
Table 3-2 A Class of Design Problems ............................................................................. 57
Table 3-3 Selected Kernel Theories .................................................................................. 58
Table 3-4 Meta-requirements ............................................................................................. 59
Table 3-5 Design and Research Plan ................................................................................ 60
Table 4-1 Applicability of CBR for Conflict Management ................................................. 67
Table 4-2 Comparison between Case and Vignette ......................................................... 70
Table 4-3 Constructs in Conflict Vignettes ....................................................................... 71
Table 5-1. Argument Detection Models in Prior Work ...................................................... 87
Table 5-2. Comparing the Performance of TAD Models against Baselines .................... 95
Table 5-3. Summary of Sub-threads in a Bug Discussion on Bugzilla ............................... 97
Table 6-1. Conflict Elements Identified in Prior Work ....................................................... 104
Table 6-3. Research Propositions ...................................................................................... 121
ACKNOWLEDGEMENTS

My PhD journey and this dissertation would not be possible without supports from many people.

First and foremost, I want to thank my advisors Dr. Sandeep Purao and Dr. Heng Xu. I am very grateful for the best mentorship that I received and would benefit me for a lifetime. Both of them provided me enormous support that allowed me to overcome many obstacles in my journey. Their advice in regard to both my research and my career have been priceless.

I would also like to express my gratitude to my committee members, Dr. Sencun Zhu, Dr. Zhenhu Li, and Dr. Daniel Cardoso Llach. I thank all of them for spending time with me and providing me suggestions regardless of time and location constrains.

I also want to express my special appreciate and thanks to Dr. Yilu Zhou. Without her support in data mining and machine learning, I could not have completed the system design in such a short period.

I also want to thanks all IST faculty members who ever taught or mentored me. Special thanks are given to Dr. Irene Petrick and Dr. Xiaolong Zhang.

I want to thank my parents Qiliang Zhang and Huamei Liu for their unconditional love and support from the first day I went to school.

Last but not least, I want to deeply thank my wife Xiaoxia Ma and my daughter Emma Zhang. They make my life much more meaningful and beautiful.
Chapter 1

Introduction and Motivation

*If two people on the job agree all the time, then one is useless. If they disagree all the time, then both are useless.*

- Dale Carnegie

Research Background

Conflict is a pervasive phenomenon in the work place (De Dreu & Gelfand, 2008). Conflict refers to a process that begins with an incompatibility between interdependent parties (Putnam & Poole, 1987; Rahim, 2010; Thomas, 1992) that may arise because of differences in goals, differences in ways of doing things, or because of interpersonal dissonance (Jehn & Mannix, 2001; Rahim, 2010). It is then manifested as behavior that seems to adversely affect the parties in the conflict as well as the surrounding organization. Scholarly work sometimes describes conflict as the “dark side of the workplace” (Raver & Barling, 2008) because it produces tension and antagonism, and distracts conflict parties away from their assigned tasks – which in turn reduces team performance and individual satisfaction (Hackman & Morris, 1975; Pondy, 1967; J. A. Wall Jr & Callister, 1995). However, it has been recognized that conflict also has functional, positive impacts, and these impacts have been getting more attention. This work shows that a moderate amount of conflict can, in fact, encourage different perspectives, stimulate innovation and creativity, and improve decision making (Levine, Resnick, & Higgins, 1993; Nemeth & Staw, 1989; Tjosvold, 1997; Tjosvold & Johnson, 1977) while preventing teams from succumbing to stagnancy, mediocrity, and group-think (Nemeth, 1986). These potential positive
influences of conflict – along with the realization that conflict is inevitable – suggest that managing conflict is a critical problem in organizations.

Contemporary responses to this problem range from “best managerial practices” that seek to avoid conflict and resolve its negative effects (APM, 2006; PMI, 2004), to popular writings that delineate the functional impacts of conflict, propose productive interaction, and suggest negotiation strategies (Eisaguirre, 2002; Joni & Beyer, 2009; Weiss & Hughes, 2005). The scholarly community has contributed to the discussion with investigations of conflict, generic strategies for conflict management, and by describing the factors that matter in conflict process (Jehn & Mannix, 2001; Rahim, 2010; Thomas, 1992). While some social systems, including organizational structures and conflict report and resolution mechanisms, have been proposed for conflict management (Yarn, 2014), the application of information technologies has not received as much attention from the scholarly community, and even less from researchers from the information system communities.

Research Setting and Problem Statement

This research deals with conflict in the workplace, as opposed to the conflict that occurs inside families or between social groups. In particular, I select distributed teams as the focal setting for the research because managing conflict is crucial for realizing the potential of these teams and preventing project failure (Hinds & Mortensen, 2005; Montoya-Weiss, Massey, & Song, 2001). The availability of archived communication data between team members also provides the opportunity to explore conflict management approaches in these settings.
Distributed Teams

A distributed teams is defined as a group of people who share a common purpose, carry out interdependent tasks, work across space and time, and interact primarily through computer-mediated communication (Cramton, 2001; Majchrzak, Malhotra, & John, 2005). Driven by the global economy and enabled by information and communication technologies (ICTs), distributed teams have become a popular organizational form in contemporary companies, especially in high-tech industries that employ knowledge workers scattered around the world (Berry, 2011a; McDonough III, Kahn, & Barczak, 2001). Examples of distributed teams include new product development teams (McDonough III et al., 2001), global software development teams (Espinosa & Kraut, 2002; Vlaar, Fenema, & Tiwari, 2008), and open collaboration communities (Filippova & Cho, 2015; J. Wang, Shih, & Carroll, 2015).

In spite of the fact that an increasing number of projects are conducted through distributed teams, these teams are often found to be difficult to manage and fall short of performance expectations (Hinds & Mortensen, 2005). Previous studies show that distributed teams are subject to coordination problems (Cramton, 2001), crises of trust (Jarvenpaa & Leidner, 1998), unhealthy team dynamics (Armstrong & Cole, 2002), and severe conflicts between team members (Mannix, Griffith, & Neale, 2002). While these problems are also observed in collocated teams and can be explained by generic models, it has been argued that separate studies are required for distributed teams because of their unique characteristics and dynamics (Hinds & Mortensen, 2005). To help realize the potential benefits of distributed teams, many studies have been conducted to investigate the salient managerial issues that tend to be found in distributed teams, such as developing trust between team members, establishing leadership, enhancing knowledge transfer, and managing conflict (See Table 1-1).

The present research focuses on conflict and conflict management in distributed teams for three reasons. First, conflict is inevitable in these teams. Distributed teams are composed of
members who are drawn from different departments or organizations; have various knowledge and cultural backgrounds, and who often know very little about each other at the outset. Conflict arises when they make decisions about the goals, strategy, and schedule for their work (Cramton, 2001), and when misunderstandings are not successfully resolved through electronic communication (Armstrong & Cole, 2002).

Table 1-1 Managerial Issues in Distributed Teams

<table>
<thead>
<tr>
<th>Issues</th>
<th>Descriptions</th>
<th>Approaches</th>
<th>Example References</th>
</tr>
</thead>
</table>
| Trust           | Trust building is inhibited by geographical, temporal, and technical constraints in distributed teams | • Initial trust  
• Trust transition  
• Reward | (Rousseau et al., 1998)  
(Berry, 2011a)  
(Kuo & Yu, 2009) |
| Leadership      | Leadership becomes particularly complex in distributed environments, (e.g. physical location and number of leaders, process used to determine leaders, the presence and control of followers). | • New roles  
• Initial pressure | (Ocker et al., 2011)  
(Zigurs, 2003)  
(Yoo & Alavi, 2004) |
| Knowledge Transfer | It is hard to achieve knowledge sharing and integration in distributed teams due to stickiness of information, unique local working norms, and limited opportunities for discussion and debate. | • Shared passion  
• Rapport  
• Knowledge repository | (Lee et al., 2014)  
(Oshri et al., 2008) |
| Conflict        | Conflict in distributed teams tends to be detrimental and resist resolution due to lack of shared identity and context, and because of information diversity. | • Swift trust  
• Spontaneous communication  
• Coordination Mechanisms | (Cramton, 2002)  
(Hinds & Mortensen, 2005)  
(Mannix et al., 2002) |

Second, managing conflict is at the crux of addressing other managerial issues in distributed teams. Previous studies have identified a variety of managerial issues that tend to be salient and unique in distributed teams. Conflict has been repeatedly identified as one of the most important concerns in solving other managerial problems. For example, conflict inhibits the development of trust in distributed teams. Severe and escalated conflict between distant team members can weaken their confidence in the project and ferment tension between them, which in
turn can prevent the development of trust (Berry, 2011b; Kuo & Yu, 2009). Conflict also causes obstacles to *knowledge transfer* in distributed teams because it impairs rapport and communication among team members (Kotlarsky & Oshri, 2005; Lee, Bassellier, & Faraj, 2014). Monitoring and resolving conflict therefore represents an important task that the *leader* of a distributed team must consider in order to maintain optimal team functionality, as well as building morale and relationships within the team (Mannix et al., 2002; Ocker, Huang, Benbunan-Fich, & Hiltz, 2011).

Third, evidence shows that conflict is not only detrimental in distributed teams, but particularly difficult to cope with when compared with collocated teams (Bergiel, Bergiel, & Balsmeier, 2008; Cramton, 2001; Hinds & Mortensen, 2005; Montoya-Weiss et al., 2001). Several factors make managing conflict in distributed teams challenging. One is that members in distributed teams tend to misunderstand each other and take a competitive stances because of a lack of shared identity and shared context (Espinosa & Kraut, 2002; Mortensen & Hinds, 2001). Another factor is that conflicts in distributed teams might not be easily handled by traditional organizational structures and managerial approaches (De Dreu & Gelfand, 2008; Rahim, 2010). For example, “appeal to authority” (a traditional way to solve conflict between peers) is invalidated in distributed teams since they consist of members from different departments who have different reporting lines. The third reason is that conflicts in distributed teams are hidden in asynchronous and discontinuous electronic communication streams. Constrained by time and attention, participants are likely to overlook conflict information, to lose track of the progress of a given conflict, and to take inappropriate, even counterproductive action (Kahneman, 2011; Thomas, 1992).

In response to these challenges, a variety of approaches and mechanisms have been proposed to manage conflict in distributed teams. Examples includes developing swift trust between team members (Mannix et al., 2002), creating transative memory within the team (Hinds & Mortensen, 2005), and enforcing coordination mechanisms (Montoya-Weiss et al., 2001).
While informative and useful, these approaches are subject to three limitations. First, they focus on preventing negative conflict, but have nothing to say about embracing and exploiting productive conflict; some level of conflict is necessary to elicit different perspectives, stimulate creativity, and improve decision making in distributed teams. Second, these approaches are rarely tied to the generic conflict management knowledge accumulated in previous studies. Therefore, the effectiveness of these approaches is in doubt since they do not directly tackle problems known within conflict management practice. Third, these approaches fail to incorporate IT support. Ironically, although technology provides the foundation for distributed teams, advanced applications (such as analytical tools for supporting conflict management) have rarely been investigated and tested to deal with conflict in distributed teams. In the absence of such applications, the difficulty in tracking and understanding conflict increase exponentially as conflict parties are overloaded with electronic communication data.

**Conflict and Conflict Management**

Organizational conflict has been intensively studied in the fields of organization, psychology, and communication (De Dreu & Gelfand, 2008; Putnam & Poole, 1987; Thomas, 1992). These studies have generated a series of theories that help to understand and manage conflict in organizational settings.

Understanding a conflict starts with examining its *cause*. Prior work suggests three broad types of conflicts: task, process and relationship (Jehn & Mannix, 2001). *Task* conflicts arise because of incompatible viewpoints and opinions pertaining to essential ideas of the task, such as its goal, logic, and value. *Process* conflicts are triggered between people who have different viewpoints and opinions pertaining to duty, resource delegation, and schedule. *Relationship* conflicts can be traced to adverse emotions and feelings between people. Understanding the cause of a conflict is important because it can both predict the outcome as well as suggest strategies for
mitigating the consequences. For example, prior studies show that task conflicts have the potential to lead to functional outcomes while process and relationship conflicts often lead to dysfunctional consequences (Jehn, 1995).

Conflict also needs to be understood and managed as a process. Conflict may be conceptualized following a sequential model, or a more nuanced, punctuated model. The punctuated process model emphasizes events that occur as a conflict unfolds. In line with the punctuated equilibrium theory, the punctuated process model describes the conflict process as alternating between stasis and transition (Gersick, 1988; Putnam, 2004). Here, a transition refers to a short period of time when a conflict shifts substantially in terms of its intensity, or its progress toward resolution. A transition is characterized by the behavior and emotions of the parties in the conflict. The conflict escalation theory suggests that a conflict may undergo a combination of two types of transitions – conflict escalation and conflict de-escalation (Pruitt, 2008). Conflict escalation occurs when a conflict is intensified with mechanisms such as more extreme strategies, emotional involvement, issue generalization, or through an increased number of participants. Conflict de-escalation occurs when conflict parties re-assess the conflict situation and change their language and behavior to more moderate and collaborative.

Knowing the roles played by conflict parties is a precondition for understanding their behavior. The parties involved in a conflict can be broadly classified into two categories: principal conflict parties and third parties. The term ‘principal conflict parties’ refers to individuals or groups who are interdependent with one another in the pursuit of their interests in spite of the incompatibilities between them (Putnam & Poole, 1987). The term ‘third parties’ refers to individuals or groups who intervene in the conflict either to fulfill their duties or because they have a stake in the conflict (Pruitt & Kim, 2004). Understanding a conflict situation, thus, requires recognizing the perspectives and actions of both principal conflict parties as well as third parties who are affected by the conflict.
Prior work also outlines high-level conflict management *strategies* to characterize the general intention of the principal conflict parties toward a conflict and the approaches they use for coping with the conflict (Thomas, 1992). Scholars have conceptualized conflict management strategies in a space defined in two dimensions – concern for self and concern for others. Combining these two dimensions results in five styles or strategies that the principal conflict parties can use: collaborating (H-H), compromising (M-M), yielding (L-H), contending (H-L), and avoiding (L-L) (Olekalns, Putnam, Weingart, & Metcalf, 2008). A conflict management strategy, thus, represents a composite construct that characterizes the general orientation of a conflict party, not a specific action.

Conflict management is a task full of uncertainty. Although existing conflict, and conflict management theories provide basic guidelines for understanding and coping with conflict, specific conflict management approaches that can help practitioners put these theories to practical use are still needed. For example, conflict management strategies provide high-level explanations of conflict behavior. However, there is lack of discussion about how to apply these strategies in practice. One important reason is that conflict party behavior is influenced by a very large number of internal and external factors. The relationship between these factors and the resulting behaviors cannot be captured in structured and unambiguous rules. The absence of this kind of knowledge has lead to a gap between research and practice.

**IT Support in Conflict Management**

It is possible to conceptualize managing and resolving conflicts – like other cognitively demanding tasks – as a task that may benefit from technological support. In spite of the inherently messy and often unpredictable nature of conflict, scholars have suggested such uses of information technology for conflict management. These prior scholarly efforts have developed
two genres of approaches for managing organizational conflicts – communication-driven and decision-driven.

*Communication-driven* approaches are aimed at facilitating interactions between the conflict parties and supporting information processing activities. The implementations of these approaches are often directly adopted from group decision support systems (GDSS) (Chiravuri, Nazareth, & Ramamurthy, 2011; El-Shinnawy & Vinze, 1998; Sambamurthy & Poole, 1992) or Negotiation Support Systems (NSS) (Delaney, Foroughi, & Perkins, 1997; Johnson & Cooper, 2009; Wolfe & Murthy, 2006). These systems are adopted to provide features such as anonymous posting (El-Shinnawy & Vinze, 1998; Sambamurthy & Poole, 1992), procedural enforcement (Chidambaram, Bostrom, & Wynne, 1990), problem visualization (Chiravuri et al., 2011), team organization (Miranda, Bostrom, & Robert, 1994), and communication channels (Anson & Jelassi, 1990; Wolfe & Murthy, 2006). These features have been shown to contribute to encouraging collaboration, reducing conflict, and facilitating conflict resolution, although they are not entirely free of negative impacts (Chiravuri et al., 2011; Johnson & Cooper, 2009).

However, these GDSS and NSS approaches tend to support deliberations within groups that are collocated, and are not primarily designed for longer-duration decision-making or negotiation. Although they may (arguably) be extended for distributed teams, their applicability remains open to question.

*Decision-driven* approaches refer to those that provide useful information to conflict parties to better understand the conflict at hand and to make decisions. Implementations of these approaches provide features such as visualization (Flack & Summers, 1971; Hine & Goul, 1998; Taratoukhine, 2002), and identifying potential and optimal resolutions (Fraser & Hipel, 1981; Klein, 1991; Madani, Rouhani, Mirchi, & Gholizadeh, 2014). Like the features outlined for the first category (communication-driven approaches), these features have also been shown to contribute to conflict resolution (Hamouda, Kilgour, & Hipel, 2006; Klein, 1993). However, these approaches are often aimed at only a small component of the overall conflict process (e.g. the
decision phase), and do not help users in tasks such as detecting and tracking conflicts or exploring different conflict resolution strategies. Much like the approaches in the first category, it is possible to argue that the approaches in the second category may be extended to cover more conflict phases and new conflict situations. However, their applicability also remains limited and open to question.

The two genres of approaches, thus, have some inherent limitations. Each provides only a fractional and static view of conflict. Neither incorporates an underlying model that can represent the entire conflict process. These critiques are not surprising because both approaches were not originally designed to support the management or resolution of conflict. Rather, they have been appropriated for conflict management from their original domains – coordination and decision-making. Because of this, they do not directly account for or conceptualize ideas such as events that occur as a conflict situation unfolds and the different behaviors of conflict parties. As a result, these systems cannot provide users the ability to understand diversity across conflict situations, or to make effective conflict management choices based upon them.

The limitations described above shows that the design of conflict management systems and approaches remains understudied in the literature. One key obstacle constraining the development of two genres of approaches and other novel approaches, to my observation, is the difficulty in locating data sources that capture a wide range of conflict information. In the traditional conceptualization, the negotiations through which conflict arises and is resolved are conducted via fleeting verbal dialog between the conflict parties. It is not only technically difficult to capture these negotiations into usable data, but also socially inappropriate to do so as conflict is usually considered as an embarrassing and sensitive topic (Jehn, 1997; Raver & Barling, 2008). Without data support, the development of conflict management approaches has been concentrated on well-understood conflicts, processes, or solutions of that could be codified and optimized. Social and behavioral factors have been excluded from the design. However, the generation of CMC data in distributed teams provides an opportunity to overcome this obstacle
and to design new conflict management approaches that can be used not only to broaden the reach of current conflict management practice, but to address the specific problems arising within distributed teams.

**Research Objectives and Questions**

The present research opportunity lies at the intersection between conflict studies and distributed team studies. As detailed in the literature review, managing conflict is a critical task in distributed teams because these teams are more subject to detrimental conflicts. Understanding and managing conflict is necessary to prevent the negative effects of these conflicts, and is at the crux of addressing other cooperative issues within teams. Although CMC tools lay the foundation for distributed teams and generate a large amount of rich data in practice, the data has not been exploited to support participants and researchers in making sense of conflict because of a lack of analytical tools. From the other side, prior conflict studies have generated a great deal of knowledge and many models relating to conflict and conflict management in organizations, but, the design of conflict management systems has rarely incorporated them. This is a result of the fact that system design is hindered by lack of data sources that can capture the process of conflict (such as the events that occur as a conflict unfolds, and the action taken by conflict parties to tackle the conflicts). I contend that investigating novel artifacts to extract and make sense of conflicts that occurred in distributed teams will contribute to both fields of study.

The objective of this study is to **design, implement, and apply a design theory in the form of a novel conflict detection and analysis approach that can support people in making sense of conflicts that occurred in computer-mediated communication.** To fulfill the objective, this research will be conducted to answer a set of research questions.
RQ1: What is the information necessary to understand a conflict? How should the information be organized and presented to promote sense-making in conflict situations?

RQ2: How to extract conflict information from communication data?

RQ3: How to apply the proposed approaches to analyze conflict situations?

Structure of This Proposal

This proposal is structured as follows. In Chapter 2, I first review literature about conflict and conflict management to develop a generic and systematic view of conflict, followed by the discussion about its uniqueness in distributed teams. I then turn to an investigation of existing approaches that have been proposed for conflict management in order to explore shortcomings in the system design. In Chapter 3, I describe design science research (the methodology used for this research) and discuss the philosophical foundations of the methodology. After that, I outline the detailed research plan that was followed to organize the present research. In Chapters 4 to 6, I develop three essays, each of which answers one of the research questions. Specifically, Chapter 4 describes the proposed novel conflict management system (CM2) and its implementation. Chapter 5 describes a methodology aimed at automating the process of extracting conflict information from raw communication data. Chapter 6 presents the results of the analysis that I conducted, and extends prior design outcomes to understand the conflict phenomenon in Bugzilla, an online collaboration community. In Chapter 7, I conclude the dissertation with a discussion of the research outcomes.
Chapter 2

Literature Review

Conflict, as a pervasive organizational phenomenon, has been intensively studied in the fields of organization, communication, and psychology. Meanwhile, as information and communication technologies become indispensable in contemporary organizations, systems and applications that can be used for conflict management have been proposed and tested in information systems and computer science studies. In this chapter, I will first review the literature on conflict conducted in these three fields to develop a systematic but generic view of conflict in the organizational setting. This is followed by an investigation into conflict that occurs in distributed teams. I will then turn to a discussion of existing approaches that have been designed and used for conflict management that reveal possible shortcomings in the system design.

Conflict in Organizations

Conflict occurs both within people and between people in organizations, (i.e., intrapersonal conflict and interpersonal conflict). Intrapersonal conflict occurs when one individual is engaged in two or more mutually exclusive activities (Bazerman, Tenbrunsel, & Wade-Benzoni, 1998). Examples include work-family conflict and role conflict. The former occurs when one needs to choose between the two attractive activities (Pratt & Rosa, 2003). The latter arises when one make a decision between what he or she wants to do and what he or she should do (Floyd & Lane, 2000). Participants of distributed teams are observed to be more likely to encounter intrapersonal conflicts due to role ambiguities and difficulties in managing time across projects and between work and family (Pratt & Rosa, 2003; Rutner, Hardgrave, & McKnight, 2008).
The focus of this study is *interpersonal conflict*, which involves two or more individuals who experience some kind of incompatibility between one another. Based on the organizational level at which this kind of conflict occurs, it can be further classified as *intragroup* (within a group), *intergroup* (between groups), and *interorganizational* (between organizations).

**Defining Conflict**

Conflict results from incompatibilities occurring between people. The concept of conflict has been defined from various perspectives as: a state; a cognitive awareness; a type of behavior; and as a process that involves some kind of incompatibility between two or more parties. See Table 2-1.

<table>
<thead>
<tr>
<th>Perspective</th>
<th>Definition</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>State</td>
<td>Conflict is a state in which the goals, behaviors, and conditions for different parties are inherently incompatible</td>
<td>(Smith, 1966) (Tedeschi, Schlenker, &amp; Bonoma, 2009)</td>
</tr>
<tr>
<td>Cognition</td>
<td>Conflict is an awareness in the parties involved of discrepancies, incompatible wishes, or irresponsible desires.</td>
<td>(Jehn &amp; Mannix, 2001)</td>
</tr>
<tr>
<td>Behavior</td>
<td>Conflict is a type of behavior which occurs when two or more parties are in opposition as a result of perceived incompatibility</td>
<td>(Litterer, 1966)</td>
</tr>
<tr>
<td>Process</td>
<td>Conflict is an interactive process that begins with and manifests incompatibility between interdependent social entities.</td>
<td>(Putnam &amp; Poole, 1987) (Thomas, 1992) (De Dreu &amp; Gelfand, 2008) (Rahim, 2010)</td>
</tr>
</tbody>
</table>

This study adopts the process view of conflict considering that this perspective is sufficiently broad to cover a variety of conflict issues and events (Thomas, 1992). The process view of conflict highlights three essential elements (De Dreu & Gelfand, 2008; Putnam & Poole, 1987). First, *interaction* between conflict parties is the way that conflict forms, manifests, and
resolves. Second, *incompatibilities* between social entities (pertaining to goals, behavior, and feelings, and which vary across organizational issues) are the underlying cause of conflict. Third, those social entities are *interdependent* in the pursuit of their interests, and therefore incompatibility matters to them.

According to the Process View, a conflict can be depicted in a general model as illustrated in Figure 2-1. The model shows three key topics involved in the lifecycle of conflict. From beginning to end, they are: the causes of conflict; how a conflict unfolds as conflicts parties interact with each other; and conflict outcomes. In the rest of this section, I will review findings pertaining to conflict antecedents and outcomes. Studies about conflict process will be reviewed in the next section.

![General Model of Conflict](image)

**Figure 2-1 A General Model of Conflict**

**Conflict Antecedents**

Every manifested conflict evolves from one or more latent conflicts (Pondy, 1967). Previous studies reveal three types of conflicts, each of which is connected to a group of underlying causes, see Table 2-2.

Task Conflict refers to incompatible viewpoints and opinions pertaining to essential ideas about the task, such as its goal, logic, or value. Economic theories tell us that organizational units depend on one another to obtain positive outcomes and to share a finite number of scarce resources. Task Conflict arises when choices that deploy scarce resources to benefit the interests of one unit hurt the interests of another unit (Rusbult & Van Lange, 2003). Task conflict occurs at all organizational levels. For example, individuals within a design team might disagree on what is
essential for a new product (Cronin & Weingart, 2007); departments value their own sub-goals during information system development (Robey, Farrow, & Franz, 1989); and organizations compete for resources and market share (Molnar & Rogers, 1979).

Table 2-2 Organizational Conflicts and Their Causes

<table>
<thead>
<tr>
<th>Type</th>
<th>Underlying Causes</th>
<th>Organizational Level</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task</td>
<td>Scarce Resources</td>
<td>Intragroup</td>
<td>(Cronin &amp; Weingart, 2007)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Intergroup</td>
<td>(Robey et al., 1989)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interorganizational</td>
<td>(Molnar &amp; Rogers, 1979)</td>
</tr>
<tr>
<td>Process</td>
<td>Cognitive Consistency</td>
<td>Intragroup</td>
<td>(Montoya-Weiss et al., 2001)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Intergroup</td>
<td>(Meissonier &amp; Houzé, 2010)</td>
</tr>
<tr>
<td>Relationship</td>
<td>Positive Self-image</td>
<td>Intragroup</td>
<td>(Janssen, Van De Vliert, &amp; Veenstra, 1999)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Intergroup</td>
<td>(Fiol, Pratt, &amp; O’Connor, 2009)</td>
</tr>
</tbody>
</table>

Process Conflict is triggered between people who have different viewpoints and opinions about how a task should be conducted. The causes of Process Conflict can be traced back to the idea that people are reluctant to change their ideas because of a need to preserve the cognitive consistency and social validity of their opinions, beliefs, understandings, and interpretations (Brehmer, 1976). Process conflict is often discussed at the intragroup and intergroup level. For instance, members within a team may disagree about work schedules, project pace, and specifications for time spent on tasks (Montoya-Weiss et al., 2001). At the intergroup level, for example, IT and Business departments have often been observed to dispute over the definition and execution of tasks they must complete (Meissonier & Houzé, 2010).

Relationship Conflict refers to adverse emotions and feelings between people. Humans strive for positive self-image. Relationship Conflicts arise when people attempt to derive or maintain a positive self-image in opposition to others via competiveness or even hostility (Festinger, 1954; Tajfel & Turner, 1986). Relationship conflict often appears alongside the other two types of conflict, occurring during conflict party interactions that have other (Task or Process) conflict involved. For example, managers tend to develop negative feelings about each
other if their ideas are not accepted or considered in the decision making process (Janssen et al., 1999). Relationship conflict can also develop between groups. When a feud exists between two groups in an organization, negative images and stereotypes can develop between them. Fiol et al. (2009), for example, describes a relationship conflict that occurred between the medical staff and the administrative team in a hospital.

Latent conflicts set the stage for larger potential conflicts, but whether or not a conflict breaks out is determined by a combination of factors. Four groups of factors have been identified in the literature, they are the conflict parties: characteristics (Baron, 1989; Davis, Kraus, & Capobianco, 2009; Lindeman, Harakka, & Keltikangas-Järvinen, 1997; Thompson & Hrebec, 1996), relationships (Ben-Yoav & Pruitt, 1984; Mayer, Davis, & Schoorman, 1995; Pruitt, 2008; Varshney, 2003), organizational context (Han & Harms, 2010; Kriesberg, 2007; Pruitt & Kim, 2004; Zartman, 1995), and social context (Tinsley, 2001).

**Conflict Outcomes**

Early conflict theorists tended to regard conflict as the “dark side of the workplace” (Raver & Barling, 2008). Focusing on negative impacts, conflict has been viewed as a source of personal or organizational cost (Boulding, 1962) because it produces tension and antagonism, and potentially distracts conflict parties away from performing their assigned tasks, in turn reducing team performance and individual satisfaction (Hackman & Morris, 1975; Pondy, 1967; J. A. Wall Jr & Callister, 1995). The negative impacts of conflict has been described in a number of empirical studies (Gladstein, 1984; V. D. Wall Jr & Nolan, 1986).

The functional, positive impacts of conflict have been getting more empirical attention recently, though the concept had been recognized in theory for a long time (Pondy, 1967; R. E. Walton & Dutton, 1969). Early studies suggest that on one side, a moderate amount of conflict, handled in an appropriate way, can encourage different perspectives, stimulate innovation and
creativity, and improve decision making (Levine et al., 1993; Nemeth & Staw, 1989; Tjosvold, 1997; Tjosvold & Johnson, 1977). On the other side, too little conflict may lead to stagnancy, mediocrity, and group-think (Nemeth, 1986).

Jehn’s (1994, 1995) differentiation between task conflict and relationship conflict advances the discussion on the beneficial functions of conflict. Her initial effort (Jehn, 1994) reveals that while Relationship Conflict is negatively associated with group performance and satisfaction, Task Conflict is positively associated with group performance. Based on the argument, a significant number of empirical studies have been conducted to test the impacts of conflict on decision quality, team creativity, team cohesion, and individual satisfaction, see Table 2-3.

Table 2-3 Conflict Outcomes

<table>
<thead>
<tr>
<th>Focus</th>
<th>Construct</th>
<th>Level</th>
<th>Empirical Evidence</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Decision-making</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task conflict</td>
<td>Individual</td>
<td>Support (+)</td>
<td>(Baron, 1991)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group</td>
<td>Support (+/-)</td>
<td>(Amason, 1996)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Janssen et al., 1999)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(De Dreu, 2006)</td>
<td></td>
</tr>
<tr>
<td>Relationship conflict</td>
<td>Individual</td>
<td>Support (-)</td>
<td>(Jehn &amp; Mannix, 2001)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group</td>
<td>Support (-)</td>
<td>(Janssen et al., 1999)</td>
<td></td>
</tr>
<tr>
<td>Creativity</td>
<td>Task conflict</td>
<td>Individual</td>
<td>Support (+)</td>
<td>(Nemeth &amp; Chiles, 1988)</td>
</tr>
<tr>
<td></td>
<td>Group</td>
<td>Partially Support (+)</td>
<td>(De Dreu, 2006)</td>
<td></td>
</tr>
<tr>
<td>Team Cohesion</td>
<td>Task conflict</td>
<td>Group</td>
<td>Support (+/-)</td>
<td>(Tekleab, Quigley, &amp; Tesluk, 2009)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Schwenk, 1990)</td>
</tr>
<tr>
<td></td>
<td>Relationship conflict</td>
<td>Group</td>
<td>Support (+)</td>
<td>(Tekleab et al., 2009)</td>
</tr>
<tr>
<td>Individual</td>
<td>Task conflict</td>
<td>Individual</td>
<td>Support (-)</td>
<td>(Amason, 1996)</td>
</tr>
<tr>
<td>Satisfaction</td>
<td>Relationship conflict</td>
<td>Individual</td>
<td>Support (-)</td>
<td>(De Dreu &amp; Gelfand, 2008)</td>
</tr>
</tbody>
</table>

A closer look at the empirical evidence, however, shows a more complicated situation: contradictory findings across the studies for the impact of conflict. One explanation is that the impacts of conflict are moderated by a variety of contextual factors, such as organizational
conditions (Putnam, 1994), interdependence between conflict parties (Janssen et al., 1999), trust between conflict parties (Simons & Peterson, 2000), team norms (Jehn, 1995), and overall social conditions (Bisseling & Sobral, 2011). Another plausible explanation is that the impact of conflict is influenced by conflict expression, (the manner in which a conflict is expressed) (Weingart, Behfar, Bendersky, Todorova, & Jehn, 2014). Conflict expression is defined in two dimensions: Directness and Oppositional Intensity. Directness relates to the explicitness of the opposing positions being conveyed. And oppositional intensity refers to the degree of strength, force, or energy with which opposition is conveyed. In theory, a conflict tends to result in positive outcomes if it is communicated unambiguously and directly between conflict parties who are not entrenched in positions or subversive in actions.

**Managing Conflict in Organizations**

It is no surprise that managing conflict is a perpetual topic in the literature of organizational conflict. Instead of simply eliminating or preventing conflict, managing conflict in organizations involves a series of organized efforts directed toward limiting the negative aspects of conflict while promoting the positive aspects of conflict. In this section, I will review studies pertaining to three considerations – the process, strategy, and action – that are key for conflict management.

**Conflict Process**

In line with the process view of conflict, conflict management cannot be accomplished in one stroke. Instead, it requires a set of continuous efforts which include monitoring, understanding, planning, and acting. The goal of these efforts is to both understand the process of conflict and to develop a holistic view of the conflict. To assist people in understanding
organizational conflict, previous studies have contributed knowledge about the process of conflict and transition moments in the process.

**Process Models**

Two types of process models have been proposed, the Sequential Process Model and the Punctuated Process Model. The *Sequential Process Model* represents the mainstream view of conflict process models. Relevant models imply that a complete conflict episode will go through all stages in the sequence one after another, and that the parties must complete a series of tasks at each stage before shifting to the next one (Olekalns et al., 2008). For instance, Pondy (1967) depicts conflict processes as five stages: (a) latent conflict; (b) perceived conflict; (c) felt conflict; (d) manifest conflict; and (e) conflict aftermath. Although as an early model, this process model fails to depict events and conflict party behaviors at each stage, it does present a perspective for identifying the key questions that must be addressed at each stage. In order to present a more fine-grained picture of conflict, Thomas (1992) presents another process model, which takes the perspective of a single party in a dyadic conflict. Briefly, the model posits that a conflict begins with a party’s awareness of conflict, followed by the development of *thoughts and emotions*, followed by a formulation of *intentions* with respect to coping with the conflict. The party’s intentions are then enacted in the form of observable *behaviors*, and the other party then reacts to these behaviors. An interaction loop is established when the other party’s reaction leads to changes in the first party’s emotions and thoughts, and then intentions and behaviors. When interaction within a given conflict stops, some set of outcomes occurs. Similar models are also called *Stage Models* or *Stage-Oriented Models* (Lim & Murnighan, 1994; Olekalns & Smith, 2000).

More recently, an alternative view – concerning moment-to-moment changes in conflict, and abrupt transitions in strategy use – has received more attention. The relevant studies have
contributed to the formulation of the *Punctuated Process Model*. These models echo the Punctuated Equilibrium Theory developed in the field of organization, which depicts group process as an alternation between, “stasis and sudden appearance – long periods of inertia, punctuated by a concentrated, revolutionary periods of quantum change (Gersick, 1988).” The Punctuated Process Model suggests that a conflict in progress may be punctuated by abrupt changes in either the parties themselves and/or the context. According to Putnam (2004), the parties may reach new understandings of the conflict and their situation as the conflict unfolds, and may change their conflict strategies and behaviors accordingly. Focusing on contextual influences, Druckman (2001) finds that external events can lead to abrupt departures in the process during international negotiations.

However, unlike the Sequential Process Model, a well-defined Punctuated Process Model has not yet been proposed. Attention has been focused on investigating turning points in the process (Druckman, 2001, 2004; Kolb, 2004); transformation in understanding the conflict (Putnam, 2004); and, the improvisation of strategies and behavior (Harding, 2004). Referring to the Punctuated Equilibrium Theory, the Punctuated Process Model for organizational conflict can be depicted through the “stage 1-transition-stage 2” pattern (Gersick, 1988).

**Conflict Transition**

Both Sequential Process Models and the Punctuated Process Model indicate that conflict parties change their thoughts, emotions, intentions, and behaviors during their interactions with others. These changes lead to *transitions* in the nature and state of the conflict.

A conflict tends to *escalate* if it is not handled appropriately in the preceding episode. Conflict escalation signals an increase in the intensity of the conflict as a whole (Pruitt & Kim, 2004). Although the use of more extreme tactics can occur separately on either side, they affect the conflict as a whole because escalation on one side is usually mirrored by the other parties.
Conflict escalation begins when the conflict is intensified by all parties in ways that are sometimes exceedingly difficult to undo. According to Pruitt and Kim (2004), conflict escalation can occur in one or a combination of the following transformations: 1) movement from milder to more extreme strategies, 2) transformation of a small event requiring less attention and resources to a larger one, 3) broadening of a specific issue to a general issue, and 4) involvement of a few people to the involvement of many. Conflict escalation is usually induced from and prolonged by a combination of factors. According to Kriesberg (2007), it is determined by the issues, the chosen strategies, changes in the parties, interaction between the parties, and the general social context.

In many cases, conflict escalation is detrimental because it: 1) leads to the breakdown of working and personal relationships; 2) impedes creative thinking; c) encourages a distributive, zero-sum attitude toward conflict; 3) impedes concern for the organization as a whole; and 4) leads to conflict avoidance, which leaves important issues unresolved (Pruitt, 2008). However, it has been observed that conflict escalation can also produce functional outcomes. For example, mild escalation can help to reveal the importance of the issue/s underlying the conflict, encourage reactions such as information gathering, lead to a rethink of activities or policies, or prompt structural changes (Skarlicki & Folger, 2004). Escalation can also help reveal the interdependence between various parties and promotes the development of “logrolling” solutions (Pruitt, 2008). Furthermore, the fear of escalation can galvanize stakeholders to pay more attention to the conflict (Bies, Tripp, & Kramer, 1997).

The obvious resolution of conflict escalation is de-escalation. A constructive de-escalation begins when conflict parties perceive that they are trapped in a “hurting stalemate” (Kriesberg, 2007), in which neither party anticipates that the existing balance of power will change enough to enable it to triumph, even though the existing situation is highly unsatisfactory. However, a perceived stalemate does not directly result in the de-escalation of conflict, actions such as contact and communication, cooperation on other issues, and unilateral conciliatory
initiatives are necessary to initiate de-escalation (Pruitt & Kim, 2004). More possible actions can be found in the work of Kriesberg (2007).

**Conflict Management Strategies**

Conflict behaviors are governed by a hierarchy of decisions and choices. From broad to specific, they are: attitude toward the conflict, conflict strategies, and conflict tactics. Of these, conflict strategies – which represent the general intention of a given party (Thomas, 1992) – is the most richly explored area in the literature pertaining to conflict management.

A mature model comprised of five strategies has gained wide acceptance despite some variation in terminology (Olekalns et al., 2008; Rahim, 2010; Thomas, 1992). The model is constructed across two dimensions: concern for self and concern for others. The first dimension describes the degree to which a conflict party attempts to satisfy his/her own interests, while the second dimension describes the degree to which a conflict party wishes to satisfy the interests of opponents. Combining these two dimensions results in a matrix for positing and comparing five ‘styles’ or ‘modes’ of conflict strategy, which are described below and shown in Figure 2-2.

![Figure 2-2 Conflict Management Strategies](image)

**Figure 2-2 Conflict Management Strategies**
Collaborating. This mode requires a high level of concern for both self and others, and relies a great deal on integrative attitudes. Conflict parties that use this strategy spend a great deal of time identifying the interests of all parties and developing solutions that appeal to all parties. Collaborating can be a joint strategy, in which multiple parties work together to plan and realize a solution.

Yielding. This style indicates a low concern for self but high concern for others. It is also known as accommodating or smoothing because it is associated with attempting to gloss over differences and to play down conflict. It need not imply total capitulation, but it entails self-sacrifice to satisfy the interests of others.

Contending. This strategy highlights concern for self at the expense of concern for others. It usually represents as attempts to prevail or win one’s position through forcing behaviors (Thomas, 1992). The party that uses this mode can often be identified with strong distributive (zero-sum) attitudes and strong intentions to force others to yield.

Avoiding. This style indicates a low level of concern for both self and others, and entails not engaging in the conflict at hand. This strategy is usually expressed in one of two forms: inaction, or postponing the issue until a better time; and withdrawal, or ignoring the conflict and withdrawing from the conflict situation (Pruitt & Kim, 2004).

Compromising. This strategy seeks to achieve a moderate, if incomplete, satisfaction of the concerns of all parties. This mode is understood to produce a middle-ground strategy. It addresses a conflict more directly than Avoidance, but doesn’t explore it as deeply as the collaborating style. It requires parties to give up more than in the contending mode, but less than in the Yielding style (Rahim, 2010). Parties that employ this strategy usually take a distributive attitude, no matter whether the conflict is distributive or integrative in nature.

The five conflict strategies are not necessarily mutually exclusive, rather are often combined in practice (Olekalns et al., 2008). Research shows that a sequence of conflict strategies is more frequently employed in a conflict episode rather than a single strategy. One party may
choose a different follow-up strategy based on the other party’s response to the initial strategy (C. Conrad, 1991) and the feasibility of reaching previous agreements (McCready et al., 1996).

Research also shows that people tend to use combined strategies to maximize effectiveness. For example, it has been demonstrated that a collaborative style is typically combined with a contentious style (van de Vliert, Euwema, & Huismans, 1995).

**Conflict Management Actions**

While strategies determine the general plan for coping with conflict, specific maneuvers and actions are needed to implement a conflict management strategy (Thomas, 1992). In contrast to the consensus on conflict management strategies, studies on conflict management actions are diverse in terms of their focus and in terms of implemented strategies (see Table 2-4).

Reviewing these studies shows several findings. First, actions pertaining to collaboration and contention have received more attentions because both strategies preserve the basic interests of the executors, and are thus appealing to practitioners. Tactics for applying the contending strategy vary from attacking to entrenching. Attacking tactics involve actions that aim to force opponents to retreat by criticizing their claims, threatening, or other aggressive manners (Pruitt & Kim, 2004; Thomas, 1992). By contrast, entrenching tactics focus on defending one’s own interests and claims. Examples include rationalizing self-claims, ingratiating oneself, and coercive commitment, i.e., taking one-sided action (Pruitt & Kim, 2004).

The collaborative strategy is the most popular one mentioned in previous studies, either explicitly or implicitly. Building and maintaining a collaborative relationship between conflict parties is the intermediate goal implied in conflict management. Previous studies have discussed organizational structures that encourage collaboration (Joni & Beyer, 2009), suggestions for establishing reciprocity between parties (Uzzi & Dunlap, 2012), and principles for interacting with opponents (Weiss, Donigian, & Hughes, 2010). It is also recognized that implementing the
collaborative strategy may backfire. The initiators are subject to the risk that other parties may misinterpret or take advantage of their actions when the initiators, as a starting point, reveal information or make an initial concession (Pruitt, 1983). In order to pursue a collaborative strategy while avoiding the inherent risks, Pruitt (1983) suggests that two sets of collaborative tactics must be combined: signaling a firm commitment to one’s basic interests while signaling flexibility about the final shape of an agreement and concern for the interests of others.

<table>
<thead>
<tr>
<th>Study</th>
<th>Focus</th>
<th>Implemented Strategy</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Pruitt, 1983)</td>
<td>Collaborating tactics</td>
<td>Collaborating</td>
<td>Signaling both firmness and flexibility.</td>
</tr>
<tr>
<td>(Thomas, 1992)</td>
<td>Contending tactics</td>
<td>Contending</td>
<td>Change the other party’s rationale; change the other party’s normative reasoning; and change the other party’s emotions.</td>
</tr>
<tr>
<td>(Pruitt &amp; Kim, 2004)</td>
<td>Contending tactics</td>
<td>Contending</td>
<td>Inggratiation, promise, persuasive argumentation, shaming, tit-for-tat, threat, coercive commitment, nonviolent resistance, and violence.</td>
</tr>
<tr>
<td>(Von Glinow, Shapiro, &amp; Brett, 2004)</td>
<td>Communication in conflicts</td>
<td>Avoiding</td>
<td>Instead of direct talking, use visual and aesthetic approaches to express emotions in culture-based interpersonal conflicts.</td>
</tr>
<tr>
<td>(Weiss &amp; Hughes, 2005)</td>
<td>Conflict escalation</td>
<td>Compromising, and contending</td>
<td>Managing conflicts at the point of conflict: provide people with criteria for making trade-offs. Managing conflicts upon escalation: make escalation joint, equal, and transparent.</td>
</tr>
<tr>
<td>(Joni &amp; Beyer, 2009)</td>
<td>Prioritizing and engaging in conflicts</td>
<td>Collaborating and contending</td>
<td>Principles for prioritizing conflicts: make it material, focus on future, and pursue a noble purpose. Principles for engaging in conflicts: make it a sport, not a war; set up a formal structure; and turn pain into gain.</td>
</tr>
<tr>
<td>(Weiss et al., 2010)</td>
<td>Negotiation in conflicts</td>
<td>Collaborating and contending</td>
<td>Principles for negotiation: get the big picture; uncover and collaborate; elicit genuine buy-in; build trust first; focus on the process.</td>
</tr>
<tr>
<td>(Uzzi &amp; Dunlap, 2012)</td>
<td>Interacting with opponents</td>
<td>Collaborating, contending, compromising, and yielding</td>
<td>Method for interacting with opponents: redirection, reciprocity, and rationality.</td>
</tr>
</tbody>
</table>
Second, suggestions for conflict management actions are often developed on the basis of multiple conflict management strategies, echoing the argument that conflict strategies are not mutually exclusive (Olekalns et al., 2008). For example, Weiss and Hughes (2005) suggest that the supervisor should have two duties when intervening into the subordinate’s conflict. One is to provide criteria for making trade-offs at the point of conflict. The other is to make escalation joint, equal, and transparent if the conflict cannot be solved locally. More often, the collaboration strategy and contention strategy are advocated together (Joni & Beyer, 2009; Weiss et al., 2010). This can be explained by the argument that a conflict is more likely to be solved in a constructive way only if all parties are encouraged to express their opinions directly and explicitly (Weingart et al., 2014).

A third finding concerns the nuance involved in the interactions between conflict parties. Although we can identify some generic principles for conflict behaviors, such as redirection, reciprocity, and rationality (Uzzi & Dunlap, 2012), it is impossible to develop a finite number of formal rules to define what action is proper at what situation. For instance, direct and open talking is often encouraged in conflict management. Actions have been suggested for building a circumstance that encourages discussion (Weiss et al., 2010). However, Von Glinow, Shapiro, & Brett (2004) suggest that this is not always true. Their study shows that people face the difficulty of “finding wording” when they are emotionally involved in a conflict that occurred in multicultural teams. In these situations, the avoiding strategy seems to be more effective than other talk-intensive strategies.

**Summary and Implications**

Conflict studies in the fields of organization, communication, and psychology have developed a systematic view of conflict and conflict management along with a large number of
theories and models that explain why conflict occurs in organizations, how they unfold, and what the outcomes are. These theories are important for guiding system design in terms of how IT systems can reflect the phenomenon and how to provide support to conflict management. However, as I will discuss later, this kind of knowledge has rarely been incorporated in existing conflict management approaches and systems. A gap exists between the study of conflict itself, and studies that attempt to understand conflict management system design.

**Conflict and Conflict Management in Distributed Teams**

The first two sections discuss the generic knowledge and theories about conflict and conflict management in organizations. This section focuses on studies pertaining to distributed teams with the purpose of developing a close view of distributed teams and identifying focal theories about conflicts in these teams.

**Defining Distributed Teams**

Changes in the global market and advances of information and communication technologies are motivating and enabling distributed work. An increasing number of organizations have begun to adopt and rely on distributed teams driven by impetuses such as integrating human resources after mergers and acquisitions, acquiring external talents and expertise, and establishing a presence in multiple markets (Bergiel et al., 2008; McDonough III et al., 2001; Vlaar et al., 2008). Working in distributed teams may also be chosen by some employees who favor working with partners with different skills and backgrounds or who are looking for more flexible working time and schedule (Berry, 2011a). The capacity to apply and manage distributed teams is felt to be critical for organizations to succeed in the new global economy (Hinds & Kiesler, 2002).
A distributed team refers to a group of people who share a common purpose, carry out interdependent tasks, work across space and time, and interact primarily through computer-mediated communication (Cramton, 2001; Majchrzak et al., 2005). By the strict definition, distributed teams are the same as traditional, collocated teams in that team members are organized to complete a set of interdependent tasks to achieve a shared goal. However, distributed teams are distinct from traditional ones in two important ways. First, members of a distributed team are geographically and temporally dispersed. They work in locations that are remote from each other and may not in the same time zone. Second, distributed teams rely heavily on information and communication technologies to bridge the distance between members and to accomplish tasks, as opposed to traditional teams that are built upon face-to-face interactions between collocated team members.

Conventionally, distributed teams are often referred to as geographically distributed teams or global teams, highlighting the physical distance between team members (Montoya-Weiss et al., 2001; Potter & Balthazard, 2002). Recent studies claim that the requirement of geographic distribution can be removed from the definition by considering that distributed teams may be composed of members who collocate in the same site, but rarely meet in person since electronic communication is becoming a norm in contemporary organizations (Berry, 2011a; Mortensen & Hinds, 2001). However, in line with Hinds and Bailey (2003), I contend that physical distance between team members, together with use of computer-mediated communication (CMC), represent the two fundamental attributes of distributed teams. Other characteristics, such as different cultural/knowledge backgrounds, informational diversity, and limited shared experiences with one another (Cramton, 2002; Jarvenpaa & Leidner, 1998), can be derived from the primary attributes.
Managerial Issues in Distributed Team

Distributed teams have become a popular organizational structure in contemporary organizations, especially those of high-tech industries, that employ knowledge workers scattered around the world (Berry, 2011a; McDonough III et al., 2001). While distributed teams bring advantages in leveraging distributed talents and reducing time and cost, these teams are often found to be difficult to manage and fall short of performance expectations (Hinds & Mortensen, 2005). Previous studies show that distributed teams are subject to coordination problems (Cramton, 2001), crises of trust (Jarvenpaa & Leidner, 1998), unhealthy team dynamics (Armstrong & Cole, 2002), and severe conflicts between team members (Mannix et al., 2002). While these problems are also observed in collocated teams and explained by generic models, it has been argued that separate studies are required to address these problems because of the unique characteristics and dynamics of distributed teams (Hinds & Mortensen, 2005). In line with this thinking, a many studies have been conducted to investigate managerial issues in distributed teams, such as team empowerment (Kirkman, Rosen, Tesluk, & Gibson, 2004), collaboration know-how (Majchrzak et al., 2005), norms of CMC (Lee et al., 2014), and temporal coordination mechanisms (Montoya-Weiss et al., 2001). In the rest of this section, I will concentrate on the three issues that have been most frequently mentioned and that seem to be salient for managing distributed teams.

Trust

Interpersonal trust within a team refers to the extent to which a person is confident in, and willing to act on the basis of the words, actions, and decisions of others (Mayer et al., 1995). Collective trust is established in a team when team members share a belief that others in the team will fulfill their commitments and not accept opportunities to take advantages of others.
(Rousseau, Sitkin, Burt, & Camerer, 1998). Unsurprisingly, increased trust is often found to have constructive impact on team functioning, such as increasing task performance, improving team satisfaction, advancing relationship commitment, and reducing stress (Costa, Roe, & Taillieu, 2001; Mayer et al., 1995). Some argue that trust is especially important in distributed teams because it is the “glue” that binds dispersed team members and the “fuel” that ignite collaboration and propels teams toward success (Altschuller & Benbunan-Fich, 2010; Berry, 2011a; Griffith, Mannix, & Neale, 2003).

Building trust in distributed teams, however, is more challenging than in collocated teams (Berry, 2011a). Although trust between team members can be built upon a variety of foundations (Kuo & Yu, 2009; Rousseau et al., 1998), trust building is inhibited by geographical, temporal, and technical constraints in distributed teams. For example, calculus-based trust is based on members’ assessment of the outcomes and costs in maintaining the trust relationship. Relational or affect-based trust derives from repeated interactions between team members (Rousseau et al., 1998). Both types of trust can be gradually developed in collocated team as team members become familiar with one another through formal or informal face-to-face communication. But these trust-building methods cannot be used, or only used sparingly in the distributed environment (Berry, 2011a). To reap the benefits of distributed team, different approaches to build and maintain trust in distributed teams have been proposed. For example, Kuo and Yu (2009) contend that organization need to pay particular attention on the initial trust and its transition to calculus-based trust and later knowledge-based trust. Zolin et al (Zolin, Hinds, Fruchter, & Levitt, 2004) suggest that the reward for developing and maintaining trust should be sufficiently large to counterbalance the uncertainty and risk involved in working with strangers.
Leadership

Leadership refers to another important variable influencing the development and effectiveness of distributed teams (Ocker et al., 2011). Team leaders, either designated or emergent, are those in teams who use their problem solving skills and social control techniques to monitor project progress and team performance and take actions to solve problems (Mannix et al., 2002). In addition to task-related functions, leaders are also responsible for providing social-emotional support to maintain the morale and relationships within teams (Ocker et al., 2011). The presence of strong leadership is found to be positively associated with team success since team leaders contribute to defining a clear and engaging direction, forming rapport, and acquiring adequate resources (Hackman & Walton, 1986). To leverage leadership, a variety of models have been created to explain the traits of effective leaders, leadership behaviors and styles, and contingent/situational model of leadership emergency (J. Zhang & Faerman, 2007; Zigurs, 2003).

Leadership becomes particularly complex in the distributed environment. As opposed to collocated teams, distributed teams face new concerns, such as the number and physical location of the leaders, the processes used to determine leaders and the new roles they play in distributed teams, and their ability to be present and monitor the followers (Ocker et al., 2011; Zigurs, 2003). Although there has been a limited amount of investigation, new models of leadership are being accumulated as more empirical studies are conducted in distributed teams. For example, when studying emergent leaders in on-line class projects, Yoo and Alavi (Yoo & Alavi, 2004) find that leaders in these team display three roles: initiator (starting team communication), scheduler (setting up the temporal rhythm of the project), and integrator (integrating team member work into final deliverables). In a study of designed leaders, Weisband (Weisband, 2002) suggests that to set the stage for later interaction, leaders in distributed teams need to initiate pressure about task demands at the outset while showing awareness and consideration of others.
**Knowledge Transfer**

One of impetuses that drives organizations to apply distributed teams is the desire to integrate the distributed knowledge of workers from different departments or across organizational boundaries to create novel outcomes (Vlaar et al., 2008). In spite of differences in the innovative levels required by the tasks, distributed teams are often composed of members with different expertise, and are often tasked with topics that are complex and not well circumscribed (Lee et al., 2014). Sharing and transferring knowledge among them, therefore, plays an particularly important role in the pursuit of creativity and team performance (Finholt, Sproull, & Kiesler, 2002; Hollingshead, Fulk, & Monge, 2002). Knowledge transfer refers to a process through which the knowledge of one unit is acquired and reapplied by another unit (Oshri, van Fenema, & Kotlarsky, 2008). The literature so far has provided various explanations about how knowledge is transferred in collocated teams, (e.g. knowledge codification and socialization) (Nonaka, 1994). While electronic communication has the potential to facilitate knowledge transfer, it has been demonstrated that distributed teams face challenges in achieving knowledge sharing and integration due to a number of factors, such as the stickiness of information exacerbated by the diversity of local contexts (Cramton, 2001), unique local working norms (Oshri et al., 2008), and limited opportunities for remote team members to discuss, debate, and explain divergent opinions and perspectives (Kotlarsky & Oshri, 2005). In response, previous studies have investigated various factors that are associated with knowledge transfer in distributed teams. For example, Lee et al. (2014) find that a shared team’s passion toward the goals and shared norms of IT use help to facilitate knowledge sharing in knowledge-intensive distributed teams. Oshri, Kotlarsky, and their colleagues suggest that both human-related issues (e.g. rapport) and technique-related issues (e.g. encoding, storing, and retrieving processes involve in knowledge transfer) need to be considered to improve knowledge transfer in distributed teams.
In addition to the three managerial issues discussed above, conflict refers to another critical issue in distributed teams (Hinds & Mortensen, 2005; Mannix et al., 2002). I contend that managing conflict is at the crux of addressing other managerial issues in distributed teams since conflict has been repeatedly identified as one of the important concerns in solving other managerial problems. For example, conflict is one of the factors that inhibits the development of trust in distributed teams. Severe and escalated conflicts between distant team members could weaken their confidence in the project in general, and specifically in their project partners, fermenting tension between them, which in turn prevents the development of calculus-based and affect-based trust (Berry, 2011b; Kuo & Yu, 2009). Moreover, conflict also results in challenges to knowledge transfer because it impairs the rapport and communication among team members (Kotlarsky & Oshri, 2005; Lee et al., 2014). Thus, monitoring and resolving conflict becomes an important task that leaders of distributed team must consider in order to maintain team functionality, as well as to build morale and relationships within teams (Mannix et al., 2002; Ocker et al., 2011).

In the reset of this section, I will review studies that focus on conflict in distributed teams. By doing so, I aim to understand on a deep level whether conflicts in distributed teams are triggered in a manner different from traditional teams, their likely outcomes, and what are the existing approaches for coping with these conflicts

**Conflict Antecedents in Distributed Teams**

Instead of exploring new fundamental causes of conflict, relevant studies have focused on identifying antecedents that are unique to distributed teams. In particular, the two essential attributes of distributed teams, physical distance and use of CMC, have been repeatedly found to account for the conflict that occur in distributed teams (Hinds & Bailey, 2003). In the context of distributed teams, distance is a multidimensional concept. The distance between participants can
be measured not only by physical distance, but also time difference, organizational distance (different department, functions, and levels), and cultural diversity (Armstrong & Cole, 2002). Computer-mediated communication, or CMC, refers to a variety of communication technologies that replace face-to-face interactions among team members (Altschuller & Benbunan-Fich, 2010). It includes but not limited to instant messaging, emails, and online meeting rooms. It has been claimed that distance and CMC lead to a variety of immediate causes of conflict, such as demographic diversities, cultural differences, lack of familiarity and friendship between members, and difficulties in established trust (Berry, 2011a; Griffith et al., 2003; Hinds & Bailey, 2003; Mannix et al., 2002). My review will focus on three antecedents that have been frequently mentioned and thus seem to be prominent.

**Lack of Shared Identity.** Lack of shared team identity has long been claimed as a key facet of distributed teams which not only causes but also exaggerates conflict situations (Armstrong & Cole, 2002; Mannix et al., 2002). Social identity theory suggests that individuals’ sense of who they are is based on their group membership (Tajfel & Turner, 1986). Participants within a team tend to share a single identity when they have similar backgrounds, close functional experiences, or learn about each other over time (Berry, 2011a). However, distributed teams face challenges to develop shared identity since they are often groups of strangers who rarely meet in person. For example, an distributed product development team described in (Majchrzak, Rice, Malhotra, King, & Ba, 2000) is composed of engineers and specialists who are drawn from several companies, possess various expertise, favor different collaboration styles, and have no previous collaboration experience with each other. Participants in such teams feel little common social identity when the team is newly formed. While distributed teams might reach shared identities over time, the use of CMC further slows down this process because CMC is depersonalized and less likely to facilitate the exchange of relational information (Walther, 1995). In the absence of shared identity, distributed teams are more likely to run into conflict situations when opinions collide and misunderstandings arise (Hinds & Bailey, 2003). These conflicts tend
to burst in a sudden way and escalate quickly since team members take competitive rather than cooperative stances against each other. They tend to evaluate other’s behaviors negatively and respond with aggressive behaviors (Mortensen & Hinds, 2001).

**Lack of Shared Context.** A shared context is established within a team when its members have access to the same information and share the same tools, work processes, and work norms (Hinds & Mortensen, 2005). Shared contexts lead to shared mental models (Espinosa & Kraut, 2002) and collaborative know-how (Majchrzak et al., 2005). Shared mental models refer to organized knowledge that members share, such as about the task, each other, goals, and strategies. These models help team members develop accurate explanations and expectations about the task and member behavior, which in turn helps them coordinate implicitly (Espinosa & Kraut, 2002). An individual’s collaboration know-how is knowledge about how to communicate one’s ideas and integrate them with the ideas of others, including how to coordinate one’s actions and work with others in the team (Majchrzak et al., 2005). This know-how helps to coordinate communication practice and clarify the interpretation of information, including both the information itself and the meaning behind the information (Cramton, 2001). Members of a distributed team are likely to occupy different contexts since they have limited opportunities to take a close look at what their colleagues are doing, identify dissimilar work processes, and resolve coordination problems (Hinds & Mortensen, 2005). Moreover, some of the features of CMC, such as lack of contextual information and feedback lag, inhibit the development of shared mental models and collaboration know-how (Cramton, 2001). In the absence of a shared context, it has been observed that distributed teams are more likely to run into process conflicts, and sometimes also task conflicts (Espinosa & Kraut, 2002; Hinds & Mortensen, 2005). For example, Cramton (2001) observes a conflict over whether or not schedule online "chat" sessions between team members of two organizations. The conflict stemmed from differences in the organizational contexts, in each of which the effectiveness of online chat was valued differently.
**Informational Diversity.** Geographic dispersion and the use of CMC leads to information diversity in terms of the information available to team members and the ways in which they process information (Cramton, 2002). Members of a distributed team are scattered across multiple sites and possess different information about the project. They are usually aware of what happens at their own site, but have only partial information about what is going on at remote sites. Although CMC tools are used to eliminate informational differences, they are subject to some deficiencies. One problem is that communication via CMC reduces nonverbal and social cues, such as facial expression, tone of voice, and gestures (Potter & Balthazard, 2002). Project information – especially contextual information – may be lost in CMC channels (Cramton, 2002). The other problem is that CMC is prone to technical and human errors. As shown in a case study, communication via CMC is a kind of “leaky process,” through which some team members are not informed of key decisions and information about the project if they are left off email distribution lists (Armstrong & Cole, 2002). Moreover, informational diversity in distributed teams can also be attributed to differences in information processing. Because of geographic dispersion and use of CMC, members of distributed teams might perceive the same sets of information in different ways since they face pressures to process parallel topics, respond to each topic at different times, and weight these topics differently (Cramton, 2002). Information diversity caused by both of these reasons has been observed to increase misunderstandings among team members, and further leads to task conflict (when different perspectives cannot be recognized and resolve) and process conflicts (when members are excluded from messages) (Hinds & Bailey, 2003). Although it does not directly account for relationship conflicts, information diversity is argued to cause attribution error and ferment tension between members when task and process conflict occurs (Cramton, 2002).

Distributed teams refer to a type of social system consisting of a multitude of elements that influence one another reciprocally, making it impossible to clearly define cause and effect, or to predicate their pattern in the long term (Zimmermann, 2011). From this non-linear system
perspective, it is important to note that the antecedents mentioned above can also be viewed as the outcomes of conflict since they are entangled with one another. For example, weak team identity is also interpreted as the result of relationship conflicts in some studies (Earley & Mosakowski, 2000). Acknowledging differences in perspective, my review focus on the one-way relationship between the antecedents and conflict. Moreover, a particular conflict occurring in distributed teams usually involves multiple antecedents since these antecedents are interrelated. For example, information diversity can cause lack of shared identity and shared context.

Meanwhile, weak shared identity can also inhibit communication within the team, and then prevents the development of shared context and exacerbates information diversity.

**Conflict Outcomes in Distributed Teams**

It is a widely accepted view is that conflict is an adverse factor in distributed teams. For example, Hinds and Bailey (2003) analyze the features of distributed teams and three types of conflicts in these teams and conclude that all conflicts are generally detrimental because they cause confusion, absorb member’s time and effort, inhibit coordination, and enlarge gaps between members at different sites. The negative view of conflicts can be explained by two reasons. One possibility is that consistency is viewed as a critical prerequisite of distributed team success (Berry, 2011a). Another is that distributed teams are diverse by their nature. Instead of seeking diversity, distributed teams need to strive for common ground about expected task processes and outcomes. Because conflict tends to be severe and resist resolution in these teams, it might not be the favored choice to improve team performance and team cohesion (Bergiel et al., 2008; Hinds & Mortensen, 2005).

The negative impacts of conflict can be explained by attribution error. Attribution is the process by which people make inferences about the cause of events (Weiner, 1985). Essentially, two types of attribution can be made on basis of the locus of activity: dispositional attribution (or
internal attribution) and situational attribution (or external attribution). The former contends that the behavior or outcome is determined internally, or by the disposition of the actor, and the latter contends that it is determined by characteristics of the situation (Cramton, 2002). While actors by themselves tend to make situational attribution and observers tend to make dispositional attribution, attribution errors occur when one overweighs the dispositional determinants of behaviors and events (L. Ross, 1977).

It has been argued that members of distributed teams are prone to attribution errors due to the differences in information, information processing, cognitive load, and temporal contiguity (Cramton, 2002). When they encounter misunderstandings and conflicts, they are likely to make dispositional attribution about their remote colleagues, and therefore, mix task conflicts with relationship conflicts. As mentioned earlier, while the positive effects of conflict are tied to task conflicts, relationship conflicts are often harmful to the team (Jehn, 1994; Nemeth, Connell, Rogers, & Brown, 2001). The entanglement between task and relationship conflicts makes conflicts in distributed teams more likely to escalate and to be hard to cope with. As these conflicts are accumulated within the team, they impair relationships and cooperation among team members, distract members from tasks, and finally compromise team performance.

Due to the two concerns mentioned above, conflict has been assumed to be negative in distributed teams (Hinds & Bailey, 2003; Mannix et al., 2002). Nevertheless, a small number of empirical studies have been conducted to test the effect of conflict in distributed teams, but the results are contradictory. In a study of 24 product development teams located across five companies, Mortensen and Hinds (2001) find that both task and relationship conflicts detract from performance in these teams, and further that the negative impact is greater in distributed teams than that in geographically collocated teams. They explain that relying heavily on CMC makes it more difficult for distributed teams to resolve task conflicts effectively and therefore makes them less able to reap the benefits of these conflicts. Another study by Montoya-Weiss et al. (2001) shows that conflicts can result in both positive and negative impacts depending on how
they are managed. In an experiment conducted with 35 distributed teams, they find that conflicts lead to negative effects on team performance when they are handled via avoiding and compromising conflict management behaviors, while competitive and collaborative conflict management behaviors are positively related to team performance. The results also show that, temporal coordination mechanisms, such as process structure imposed to intervene and direct the pattern, timing, and content of communication in a team, help to weaken the negative effects caused by some conflict management behaviors.

Instead of directly investigating the impact of conflict, some case studies imply that conflict has less influence on team performance when it is controlled within a limited scope and managed appropriately (Majchrzak et al., 2000; Peters & Manz, 2007). For example, Majchrzak et al. (2000) observed an inter-organizational product development team over a 10 month period. They found that the team ran into process-related conflicts since the electronic collaborative technology brought new norms of communication that were incompatible with existing organizational structures with which the members were familiar. The team then took three adaptation steps to resolve the misalignment between the pre-existing organizational environment, the group, and the technologies, and to resolve the conflicts derived from the misalignment. In the end, the team appeared to be less influenced by these conflicts and was successful at achieving the design goal.

Managing Conflict in Distributed Teams

Taking conflict as an adverse factor, a variety of conflict management approaches have been proposed to prevent conflict or to alleviate the negative effects of conflict in distributed teams. The three specific approaches that have been stressed in the literature are developing swift trust, promoting spontaneous communication, and enforcing temporal coordination mechanisms.
**Swift Trust.** Trust, as the “glue” that binds teams and propels them toward success, has been claimed to be especially important in distributed teams (Altschuller & Benbunan-Fich, 2010; Berry, 2011a; Griffith et al., 2003). As opposed to traditional teams where members have sufficient time and face-to-face interactions to develop trust in their work, it is argued that *swift trust* is more critical for distributed teams (Mannix et al., 2002). Instead of developing gradually, swift trust is assumed based on members’ previous experiences regarding the project, and expectations derived from social categories or stereotypes about other members. It is later verified or adjusted accordingly (Jarvenpaa & Leidner, 1998; Zimmermann, 2011). Swift trust, although it appears to be fragile and temporal, does help to ignite team work and contributes to the development of knowledge- or action-based trust that lays the foundation of team collaboration (Kuo & Yu, 2009). In relation to conflict management, distributed teams that have enacted swift trust are more likely to develop long-term trust and shared identity, and tend to be more willing to interpret conflict as task based (Altschuller & Benbunan-Fich, 2010; Zimmermann, 2011). This trust will in turn prevent the occurrence of severe conflicts and help team members to take more cooperative actions when misunderstandings and conflicts arise. To improve conflict management via swift trust, Mannix et al. (2002) contend that distributed teams should: be organized in ways that members have clear roles and well-defined specialties; value the reputation of their group of origin; and be assigned to tasks that require moderate levels of interdependence among members. More specifically, Jarvenpaa and Leidner (1998) provide nine suggestions that are aimed at facilitating and maintaining swift trust in distributed teams. The suggestions range from communication behaviors, such as communication with enthusiasm and timely response, to actions behaviors, such as resolving technical and task uncertainties at the outset, and taking a somewhat phlegmatic approach to crisis.

**Spontaneous Communication.** Physical distance decreases closeness and affinity between team members, and the use of CMC further slows down the development of social ties and mutual awareness. Both together result in weak shared identity and information diversity that
has the potential to ferment conflict in distributed teams (Mortensen & Hinds, 2001). Improving communication, therefore, is claimed to be foundational for managing conflict in distributed teams (Espinosa & Kraut, 2002; Maznevski & Chudoba, 2000). One of approaches to do so is spontaneous communication, which refers to informal, unplanned interactions that occur among team members via CMC (Hinds & Mortensen, 2005). Spontaneous communication is not unique to distributed teams. More recent studies show that conversations neither about topics on meeting agendas, nor on specified problems or specific queries for expertise frequently occur in face-to-face meetings (Dossick & Neff, 2011). While these conversations (also called as messy talk) is perceived as inefficient, they increase familiarity between members, lead to unexpected discoveries, and ultimately improve team effectiveness (Dossick & Neff, 2011). In distributed teams, spontaneous communication has also been discovered to compensate for a loss of meaning in CMC, to increase awareness of other’s ways of thinking and current states, and to build bonds between distant team members (Hinds & Mortensen, 2005). These results of messy talk can neutralize the negative effects caused by geographical dispersion and the lack of social cues in CMC. In line with this thinking, distributed teams are suggested to encourage and facilitate spontaneous communication among members. Example tactics include holding more synchronous communication, such as online-meetings, and to maintain a certain level of flexibility in the agenda (Maznevski & Chudoba, 2000). Meanwhile, social exchanges, such as conversations about member hobbies, weekend activities, and families, should be encouraged in all electronic communication, especially when the team is newly formed (Jarvenpaa & Leidner, 1998)

Temporal Coordination Mechanisms. Electronic communication, by its nature, tends to be lean and asynchronous (Mortensen & Hinds, 2001; Saunders & Ahuja, 2006). As opposed to synchronous interactions wherein verbal and nonverbal cues can be used to regulate the flow of conversation, convey subtle meanings, and provide immediate feedback, asynchronous communication flows from the norm that multiple topics are under discussion at any given time, and that every team member responds to these concurrent topics at different times (Ocker, Hiltz,
Turoff, & Fjermestad, 1995). While this norm allows multitasking, it can also increase the chance of information overload and loss, and thus increase the chance of conflict due to discontinuous and disjointed communication (Potter & Balthazard, 2002). To solve the problem, Montoya-Weiss et al. (2001) suggest that distributed teams require a deliberate creation of norms that are organized as temporal coordination mechanisms to regulate communication and work. Example mechanisms includes revealing team members’ initial positions, imposing intermediate tasks, and setting time limits for response and tasks (Montoya-Weiss et al., 2001).

Summary

Conflict has been recognized as one of challenges that make distributed teams difficult to manage and likely to fall short of team performance. A number of studies have been conducted to investigate conflict and conflict management in these teams based on two of their attributes, geographical dispersion and used of CMC. In general, the literature shows that conflict not only tends to be prevalent, severe, and detrimental in distributed teams, but also fester longer and resists resolution. As team members work from different sites and primarily rely on technologies to communicate and work with one another, they have little time or opportunity to develop shared identity and context, and possess different information about the project and one another. Under such circumstance, distributed teams are more likely to run into conflict situations when misunderstandings and differences in objectives, strategies, and processes arise. These conflicts tend to escalate and ultimately result in detrimental effects, since they distract team members from tasks, increase divergence, and cause tension and dissonance inside the team. In order to prevent and solve conflicts, approaches such as building swift trust at the outset, encouraging spontaneous communication, and enforcing temporal coordination mechanisms, have been suggested.
Relevant studies add knowledge about conflict and conflict management in the particular organizational setting of the distributed team. While they identify some factors that make conflicts in these teams difficult to manage, their findings still fall within the framework of the generic view of conflict. Basic components of conflict and conflict management behavior can be adopted to analyze conflict in distributed teams and accordingly to develop management approaches.

**IT Support for Conflict Management**

Information and communication technologies have caused revolutionary change in organizations, and specific technologies to support conflict management have been developed and tested for decades. To the best of my knowledge, the earliest study can be traced back to 1971 when Flack and Summers proposed a computer program that could visualize and resolve conflicts in water resource planning. Since then, a variety of technologies have been tested and designed for conflict management. Reviewing these studies shows that two types of systems – Communication Driven Systems and Decision-Driven Systems – have been proposed, applied, and tested. In this section, I begin by reviewing studies relevant to Communication-Driven Systems, followed by a review of studies that concentrate on Model Driven Systems.

**Communication-Driven Systems**

Communication-driven systems provide functions intended to facilitate interaction between conflict parties, such as information collection and sharing. Due to their affinity to Group Decision Support Systems (GDSS) and Negotiation Support Systems (NSS), none of the identified studies propose new designs for the systems. One exception is the work by Anson and
Jelassi (1990) which describes a framework consisting of a set of modules that can be combined to support integrative bargaining processes between conflict parties.

The relevant studies have focused on investigating the application of existing GDSS and NSS systems and evaluating their influence on conflict and conflict management (see Table 2-5).

A quick look reveals that these technologies appear to be constructive for conflict management, although both beneficial and detrimental influences appear to coexist.

Table 2-5 Use of Communication-Driven Systems and Their Influences

<table>
<thead>
<tr>
<th>Authors</th>
<th>Date</th>
<th>ITs</th>
<th>Findings</th>
<th>Features and Influences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chidambaram, Bostrom, &amp; Wynne</td>
<td>1990</td>
<td>GDSS</td>
<td>GDSS teams perform increasingly better in conflict management than manual teams over time, although they have inferior performance at the beginning.</td>
<td>Preventing coalition (+) Anonymity (+) Procedural support (+) Equality (+)</td>
</tr>
<tr>
<td>Sambamurthy &amp; Poole</td>
<td>1992</td>
<td>GDSS</td>
<td>GDSS heightens a group’s capacity to deal with conflict constructively and productively.</td>
<td>Anonymity (+) Visualization of ideas (+)</td>
</tr>
<tr>
<td>Miranda, Bostrom, &amp; Robert</td>
<td>1994</td>
<td>GDSS</td>
<td>Use of GDSS reduces both task and interpersonal conflict.</td>
<td>Task-focused team structure (+)</td>
</tr>
<tr>
<td>Delaney, Foroughi, &amp; Perkins</td>
<td>1997</td>
<td>NSS</td>
<td>NSS leads to negotiator satisfaction with achieved solutions, reducing the chance of post-negotiation conflict.</td>
<td></td>
</tr>
<tr>
<td>El-Shinnawy &amp; Vinze</td>
<td>1998</td>
<td>GDSS</td>
<td>GDSS prevents group polarization and induces beneficial conflicts.</td>
<td>Equality of participation (+) Anonymity (+)</td>
</tr>
<tr>
<td>Wolfe &amp; Murthy</td>
<td>2006</td>
<td>NSS</td>
<td>NSS leads to more but positive conflict after negotiation.</td>
<td></td>
</tr>
<tr>
<td>Johnson &amp; Cooper</td>
<td>2009</td>
<td>NSS</td>
<td>NSS impedes conflict resolution.</td>
<td>Violation of social norms (-)</td>
</tr>
<tr>
<td>Chiravuri, Nazareth, &amp; Ramamurthy</td>
<td>2011</td>
<td>GDSS</td>
<td>GDSS helps to reduce conflict and generate consensus in short or long term depending on the features of technologies.</td>
<td>Visualization (+) Surfacing differences (+) Burden of learning (-) Prolongs discussion time (-)</td>
</tr>
</tbody>
</table>
Communication-Driven Systems have been empirically proven to contribute to conflict reduction – especially task conflict – in projects. Technologies do not eliminate the differences between project members; rather, they facilitate the resolution of differences and the reaching of consensus before conflicts arise or escalate. For example, Chiravuri et al. (2011) find that visualization functions included in GDSS help to reduce conflict through surfacing and resolving differences. Another study by Miranda, Bostrom, and Robert (1994) shows that GDSS helps to reduce both task and relationship conflict because it supports the creation and maintenance of a task-focused structure within the team. These systems might also bring overflow effects with regard to reducing conflict. In a study of NSS, Delaney et al. (1997) found that the use of NSS lead to negotiator satisfaction with the solution which in turn reduced the chance of post-negotiation conflict.

Technologies may not only reduce conflict within teams, but also contribute to the heightening of a team’s capacity to cope with conflict in a constructive and productive way. The features of GDSS that serve this purpose include preventing unproductive coalitions, supporting anonymous opinion sharing, promoting equal participation, and providing procedural support (Chidambaram et al., 1990; Sambamurthy & Poole, 1992). Specifically, preventing coalitions refers to preventing the formation of cliques and subgroups that would impede agreement within teams. Anonymity encourages task-oriented communication and reduces the influence of personal factors. With equal participation structures and procedural support, discussion is more likely to be strategically structured and less likely to be sidetracked by dominant team members.

The benefits of using these systems can also be achieved through inducing more, but functional conflicts. Conflict can be either detrimental or beneficial to a team, and a moderate level of conflict can improve team performance, advance team cohesion, and even prevent stagnancy and destructive conflicts (De Dreu & Gelfand, 2008). An experimental study conducted by El-Shinnawy and Vinze (1998) shows that GDSS is worthy of consideration for application because it encourages beneficial conflict within teams. The two features of GDSS –
equal participation and anonymity – reduce normative influences, and increase team members’ independence and the deep deliberation that counters the pull toward a single dominate idea. In another experimental laboratory study, Wolfe and Murthy (2006) find that NSS contributed to the prevention of impasse during a budget negotiation between supervisors and subordinates. Although subordinates perceived more task conflicts during the negotiation, the results showed that these conflicts produced positive influences on the subordinates’ post-negotiation performance. One plausible explanation is that with NSS, subordinates are more likely to focus specifically on the budget issue, resulting in attitudes open to differences and discussion.

Like other technologies, these systems are not free from negative influences. For example, issues like the burden of learning and prolonged communication time have been identified as drawbacks of GDSS – though they do not offset the benefits brought in by the technologies (Chiravuri et al., 2011). With regards to NSS, an experimental laboratory study finds that NSS can impede conflict resolution during the negotiation process (Johnson & Cooper, 2009). This study shows that in a NSS setting, conflict resolution is hard to achieve because negotiators tend to violate social norms, specifically in that second movers tend to take advantage of first movers’ concessions.

**Decision-Driven Systems**

Different from communication-driven systems which only transmit information but do not generate new information and insights, decision-driven systems generate new information from input to support uses to understand conflict and make management decisions. Very few systems have been specifically designed for this purpose. Table 2-6 summarizes the relevant studies identified in the literature. The functions provided in these systems include conflict detection and visualization, and support for identifying optimal resolutions. In this section, I will
review and compare the designs of these systems in terms of input data, analytical technique, and function.

<table>
<thead>
<tr>
<th>Studies</th>
<th>Proposed System</th>
<th>Input Data</th>
<th>Analytical Technique</th>
<th>Decision Support Functions</th>
</tr>
</thead>
</table>
| (Flack & Summers, 1971)       | Computer system that resolves conflicts in water resource planning.              | CPs’ judgments of projects (Num)                    | Correlational and multiple regression statistics | - Detect differences in judgments  
                                |                                                                                  |                                                      |                                               | - Visualize differences                       |
| (Fraser & Hipel, 1981)        | Program that identifies potential conflict outcomes.                            | CPs’ potential actions (Num)                        | Metagame analysis                             | - Facilitate users to identify equilibrium between CPs                                    |
| (Klein, 1991)                 | IT system identifies conflict in cooperative design and provides conflict resolution suggestions. | Design parameters (Num) and CPs’ answers to query language questions (Ver) | Heuristics                                    | - Detect conflicts between parties  
                                |                                                                                  |                                                      |                                               | - Classify conflict                           
                                |                                                                                  |                                                      |                                               | - Propose resolution based on classification   |
| (Hine & Goul, 1998)           | Organizational learning support system that detects conflict in member interpretation of the environment. | CPs’ answers to query questions (Ver)               | Heuristics                                    | - Detect differences in CPs interpretation of the competition environment                 |
| (Taratoukhine, 2002)          | System to detect mismatches between assembly parts of a design developed by a distributed team. | Design parameters in CAD system (Num)              | Not specified                                 | - Detect mismatches between components  
                                |                                                                                  |                                                      |                                               | - Visualize mismatches                         |
| (Hamouda et al., 2006)        | Program used to identify stable state between conflict parties.                  | CPs potential actions (Num)                         | Metagame analysis                             | - Facilitate users to identify equilibrium between CPs  
                                |                                                                                  |                                                      |                                               | - Visualize reasoning process and outcome     |
| (Madani et al., 2014)         | System to provide optimal solutions for international boundary division.         | CPs claims for territory (Num)                      | Heuristics                                    | - Identify optimized division solution                                                    |

Notes: CPs = Conflict Parties, Num = Numerical data, Ver = Verbal data
Conflict detection and visualization are two of the most popular functions realized in decision driven systems. In the earliest work, Flack and Summers (1971) design a software program that could detect differences between the judgments of various experts pertaining to water resource planning projects. The system accepts input data such as the allocation of funds between several sub-goals of the project and then runs correlational and multiple regression statistics to determine differences between the experts’ proposals. Differences in judgment are detected and visualized to support follow-up discussions. Taratoukhine (2002) proposed a similar system that could detect and visualize conflicts in product design, and the main difference between the two is the source of input data. While the former system requires manual input, the latter system accepts data directly imported from the computer-aided design (CAD) tools. With access to a much a larger data resource, the system proposed by Taratoukhine can manage design conflicts in large, complex projects such as aerospace consortium projects.

Recognizing that not all conflicts can be quantified, Hine and Goul (1998) propose a learning support system that could detect conflicts between different manager’s interpretations of a competition environment. The system works on categorical data generated from users’ answers to a set of query questions. The data is analyzed through heuristics to understand user interpretations and detect potential conflict.

Identifying potential or optimal resolutions to conflict is another function that has been realized in decision-driven systems. Based on the game theory, Fraser & Hipel (1981) depict conflict resolutions as a combination of the preferred actions of the conflict parties. Potential resolutions can be identified by searching for states of equilibrium at which no conflict party is likely to change their preferred action. In line with this thinking, they propose a computer program that could help users identify potential equilibrium points in labor-management negotiations. The system works on manually-input binary data that reflects the potential actions of conflict parties. The program was later upgraded with graphic models to visualize the reasoning process from status quo to final equilibrium (Hamouda et al., 2006).
While not all conflicts have rational solutions, the process of identifying optimal solutions is still helpful for conflict parties because it assists in developing a clear vision of the issues and options surrounding a specific conflict. An example of this type of system is proposed by Madani et al. (2014) which serves negotiations surrounding the division of territory. The system can offer multiple optimal division solutions based on an analysis of all participant claims to a shared territory, and each of the solutions is based on a unique heuristic approach.

Both of these systems (Hipel et al. and Madani et al.) offer multilateral solutions in situations where a final solution cannot be achieved solely through the actions of one or a coalition of the conflict parties. And, there are further variations on this theme. Klein (1991) proposed a system that provides conflict resolution suggestions to individual users in cooperative design. The system is fed with both numerical data (design parameters) and verbal data (user answers to questions proposed by the system) and uses heuristics to detect incompatibilities between individual parts and the overall design; to classify detected conflicts; and, to propose resolutions based on that classification.

**Summary**

Existing communication-driven and decision-driven systems represent two directions for leveraging IT toward conflict management. However, both types of systems suffer from drawbacks that prevent them from being effective. Communication-driven conflict management systems provide platforms for mediating interactions between conflict parties, but they provide few – if any – intelligent supports to generate more useful information directly pertaining to conflict. Although dialogs between conflict parties can be captured and recorded in these systems, un-processed raw data provides little useful information to help users understand the conflict process.
Decision-driven systems are able to provide valuable information to support conflict parties in understanding conflict and identifying possible resolutions, but they work only when the conflict issue can be codified. These systems are task and context dependent because their functions are designed around a particular set of numerical data representing specific conflict issues that must be resolved. Thus, these systems can only be applied or adapted to specific scenarios. Furthermore, these systems offer only a simplified view of conflict. A lot of useful information – such as the emotions felt by conflict parties, and their intentions and interactions – is not accounted for in the coding of conflict issues into numerical representation.

Although many functions in communication-driven systems are the technological enablers of distributed teams, advanced applications, such as analytical tools included in decision-driven systems, have rarely been developed and tested in distributed teams. In the absence of such applications, the difficulty in tracking and understanding conflict increases exponentially as electronic communication data is over-loaded. To provide better support, IT systems should be designed in such a way that they can take advantage of rich conflict information conveyed in CMC data, and leverage it to support conflict parties in understanding the conflict processes. Because no system has yet been developed to include these parameters and goals, I believe that this is a rich terrain of study that further study will contribute to the literature.
Chapter 3

Research Method

The present research sets out to explore novel approaches for making sense of conflict management in distributed teams. Instead of developing and justifying theories that explain or predict the conflict phenomenon, this study seeks to improve the practice by creating new artifacts. The new artifacts instantiate a novel way to approach the conflict phenomenon and points the way toward a new design theory (Gregor, 2006). The nature of this study – research through design – indicates that it falls within the scope of design science research (Gregor & Hevner, 2013; Hevner, March, Park, & Ram, 2004). In this chapter, I first discuss the philosophical foundations of design science research, and then analyze existing design research methods to abstract a generic research model. After that, I describe the overall research process of this study.

Design Science Research

In spite of some degree of variance (Sein, Henfridsson, Purao, Rossi, & Lindgren, 2011), design science research (DSR) has gained legitimacy in the field of Information Systems (IS) as a valid research paradigm (Hevner et al., 2004; Kuechler & Vaishnavi, 2008b; Purao et al., 2008). It draws attention back to the “core subject matter” of IS research – the IT artifact (Orlikowski & Iacono, 2001). A DSR study is concerned with the creation of new knowledge about a class of problems and finding solutions through building and evaluating novel IT artifacts (Hevner & Chatterjee, 2010).

Unlike the two primary research traditions that have been dominant in IS research – positivism and interpretivism – the philosophical foundations of design science research have only recently been clarified (Iivari, 2007; Purao, 2011). Understanding these foundations is
crucial for understanding the nature of design science research and deploying it correctly (Nunamaker Jr, Chen, & Purdin, 1991).

Ontology refers to the fundamental assumption of the researcher about the phenomenon of interest. Rather than taking only one phenomenon into consideration, a design science researcher looks at two conjoined phenomena – the problem of interest and the artifact designed to solve that problem. Design science researchers embrace an evolutionary ontology because their assumptions about both the problem and the artifact undergo a gradual transformation as the research process unfolds (Purao, 2011). In the process, the artifact transforms from an abstract idea (object) in the researcher’s mind to an empirical entity that exists in the real world. This process is accompanied by a parallel process in which the problem of interest transforms from a real-world perception into an abstract issue interpreted in relationship to the artifact (Purao, 2011; Simon, 1996).

Epistemology refers to the manner in which the research generates knowledge about the phenomenon of interest. Design science researchers possess a unique epistemology – ‘knowing via design’ – that is different from the positivist ‘knowing-via-observation’ and the interpretivist ‘knowing-via-participation’ (Purao, 2011). As a compliment to their ontological stance, design science researchers gain knowledge through building innovative artifacts that provide solutions to perceived problems. This epistemology represents a convergence between the problem and the artifact that the researcher achieves through dialogical exchange between the two pillars of the research. Through these iterative exchanges, design science researchers generate new interpretations of the problem in the process of developing the IT artifact.

A Design Science Research Approach

Multiple studies have been published to clarify and unify a methodology specific for design science research. These efforts explain the procedures (Peffers, Tuunanen, Rothenberger,
& Chatterjee, 2007), guidelines and principles (Hevner et al., 2004; Sein et al., 2011), and presentation of design science research (Gregor & Jones, 2007; Walls, Widmeyer, & El Sawy, 1992). As shown in Table 3-1, there is an observable trend showing that elements of design science research are becoming more fully defined over time. Some elements have been repetitively stressed across studies. Taking these elements as the essences of design science research, I synthesize them into a research model that is employed in this study (see Figure 3-1).

![Figure 3-1 Methodological Overview of the Proposed Research](image)

In line with the research model, my design and research efforts will be organized into three phases. Phase 1 is dedicated to understanding the problem of conflict and clarifying design requirements based on selected kernel theories. At Phase 2, an iterative process is employed to design a novel conflict detection and analysis approach that meet the requirements specified at Phase 1 and implement it in multiple studies. Phase 3 seeks to formalize the knowledge and the design theory archived at the first two phases, and to externalize the knowledge into three essays. Each stage involves several rounds of iterations during the study. In the rest of this section, I lay out the plan of each phase.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Problem Definition</td>
<td>Driving problem</td>
<td>G2: Problem relevance</td>
<td>Problem identification and motivation</td>
<td>Purpose and scope</td>
<td>Justificatory knowledge</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kernel theories</td>
<td>G5: Research rigor</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Meta-requirements</td>
<td>Meta-requirements</td>
<td>Definition of the objectives</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solution Design</td>
<td>Artifact design</td>
<td>G6: Design as a search process</td>
<td>Design and develop</td>
<td>Constructs</td>
<td>P3: Reciprocal shaping</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Evaluation</td>
<td>G3: Design evaluation</td>
<td>Demonstration Evaluation</td>
<td>Principles of form and function</td>
<td>P4: Mutually influential roles</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>G7: Communication of research</td>
<td>Communication</td>
<td>Principles of implementation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knowledge Formulation and Communication</td>
<td>Design theory</td>
<td>G4: Research contribution</td>
<td>Artifact mutability</td>
<td>P6: Guided emergence</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Communication</td>
<td>G7: Communication of research</td>
<td>Communication</td>
<td>Principles of implementation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3-1 Phases and Elements Suggested in Design Science Research Methods
Phase 1 - Problem Definition

Phase 1, problem definition, sets the foundation of the present research. At this phase, I seek to identify *a class of problems* to be solved and select theoretical premises that can be adopted to conceptualize the problems and guide the design. By doing so, I formulate a set of meta-requirements for the solution design.

Identify the Problem

The necessity of leveraging IT for conflict management in distributed teams has been recognized in the literature (rf. Chapter 1&2). However, in order to warrant the problem relevance (Hevner et al., 2004; Sein et al., 2011), I conducted a pilot study to gain direct understanding about the conflict management practice in these settings (G. Zhang & Purao, 2013). In the study, I conducted semi-structured interviews (Myers & Newman, 2007) to collect enterprise architects’ experiences of conflict in their work. Enterprise architects were selected because they work in enterprise architecture (EA) projects, which usually include a variety of participants coming from different departments and across sites (Lankhorst, 2009). Conflict is pervasive in these projects due to incompatible interests, differing understandings of technology, and structure and power changes caused by the technology (Meissonier & Houzé, 2010). The analysis shows that managing conflict is a crucial but challenging task that enterprise architects must cope with throughout the projects.

Together with my personal experience of working in a global team in a multinational IT company and investigation into online communication data collected from Bugzilla.org, I identify *a class of design problems* (Markus, Majchrzak, & Gasser, 2002; Walls et al., 1992) involved in distributed teams. The problems are associated with supporting distributed teams to make sense
of conflicts and make decisions for conflict management. Although the problem varies in different settings (e.g. distributed teams within an organization, across organizations, and without organization), they share some common issues so that they can be solved by a same class of artifacts/solutions (Gregor, Müller, & Seidel, 2013; Iivari, 2007). Table 3-2 describes the problems and the issues shared by the problems.

### Table 3-2 A Class of Design Problems

<table>
<thead>
<tr>
<th>The Problems</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>To support sense making and decision making for conflict management in distributed teams.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>• <strong>Information overload</strong> – Conflict is hidden in CMC, which is asynchronous, and of multiple threads (Hinds &amp; Mortensen, 2005).</td>
</tr>
<tr>
<td>• <strong>Lack of experience with remote partners</strong> – One may have limited knowledge about remote partners due to few or no previous collaboration experience (Majchrzak et al., 2000).</td>
</tr>
<tr>
<td>• <strong>Unstructured reasoning</strong> – Participants rely on their individual experience and judgment to explain conflict situations and make decisions (Cramton, 2001).</td>
</tr>
<tr>
<td>• <strong>Time and cognitive limitations</strong> – Participants may overlook or lose track of a conflict due to insufficient time and effort spent with each topic (Observation).</td>
</tr>
</tbody>
</table>

### Select Kernel Theories

Prior theories, such as conflict theories, case-based reasoning, san and speech acts are selected as the *kernel theories* to explain the conflict problem and guide the solution design (Gregor & Hevner, 2013; Gregor et al., 2013; Kuechler & Vaishnavi, 2008a). In particular, theories about conflict and conflict management are applied as “descriptive knowledge,” which explains the conflict phenomenon and the law, norms, and regularities that guide the progress of conflict. These theories are applied across all three essays. From this knowledge base, I draw descriptive and propositional models that explain the conflict phenomenon in distributed teams. The literature about case-based reasoning and argument detection, which will be reviewed in
Essay 1 and 2, presents the “prescriptive knowledge,” the ‘how’ knowledge of building the new artifact. From this knowledge base, I detail the system architecture and components that meet the design requirements.

Table 3-3 lists the kernel theories applied in this study. These theories compose the foundations of the proposed design.

Table 3-3 Selected Kernel Theories

<table>
<thead>
<tr>
<th>Theory</th>
<th>Description</th>
<th>Nature of Theory</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process View</td>
<td>A conflict unfolds with a series of events as conflict parties interact with one another.</td>
<td>Descriptive</td>
<td>Thomas, 1992</td>
</tr>
<tr>
<td>Punctuated Process Model</td>
<td>A conflict may undergo several turning points (transitions) before it reaches resolution.</td>
<td>Descriptive</td>
<td>Putnam, 2004</td>
</tr>
<tr>
<td>Conflict Escalation Theory</td>
<td>A conflict transition is associated with either conflict escalation or de-escalation.</td>
<td>Descriptive</td>
<td>Pruitt &amp; Kim, 2004</td>
</tr>
<tr>
<td>Conflict Management Strategies</td>
<td>5 types of strategies: collaborating, compromising, contending, yielding, and avoiding.</td>
<td>Descriptive</td>
<td>Olekalns et al., 2008</td>
</tr>
<tr>
<td>Case-Based Reasoning (CBR)</td>
<td>A new problem can be resolved by referring to previous problems and their solutions</td>
<td>Prescriptive</td>
<td>Ross et al., 2002</td>
</tr>
<tr>
<td>CBR System</td>
<td>A CBR System includes three components: case base, case generation, and case retrieval and representation</td>
<td>Prescriptive</td>
<td>Kolodner, 1992</td>
</tr>
<tr>
<td>Argumentation Theory</td>
<td>Argumentation is composed of three elements: proposition, standpoint, and reasoning.</td>
<td>Descriptive</td>
<td>Eemeren et al., 2002</td>
</tr>
<tr>
<td>Argument Detection</td>
<td>A set of text mining and machine learning techniques</td>
<td>Prescriptive</td>
<td>see Table 5-1</td>
</tr>
</tbody>
</table>

**Define Meta-requirements**

Meta-requirements are a set of goals that a design must achieve. They are “meta” because these requirements represents the goals that a class of artifacts need to meet to solve a class of problems (Walls et al., 1992). In line with the identified kernel theories, I formulate the meta-requirements for the proposed conflict management approach (see Table 3-4).
Table 3-4 Meta-requirements

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable case generation and representation</td>
<td>Define constructs for the rebuild of conflict episodes</td>
</tr>
<tr>
<td>Provide conflict repository</td>
<td>Provide a repository of past conflict situations in order to support analytical thinking and enable learning over time.</td>
</tr>
<tr>
<td>Enable automatic conflict detection</td>
<td>Extract relevant information to detect conflicts from computer-mediated communication</td>
</tr>
<tr>
<td>Measure conflict elements</td>
<td>Define instruments to reflect essential elements of conflict, such as the process and conflict behaviors</td>
</tr>
</tbody>
</table>

This phase, problem definition, is not completed with one stroke. Rather, it is continuously refined through the iteration with solution design efforts. While problem formulation clarifies design goals for the solution design, the design effort in turn provides feedbacks and new design problems to help adjust and elicit new kernel theories and meta-requirements. For example, theories about case-based reasoning and CBR system were not recognized and selected until I started to structure the proposed conflict management approach.

**Phase 2 – Solution Design**

Phase 2 – solution design – includes an iterative design/build/test cycle through which I design a novel conflict detection and analysis approach that meet the meta-requirements, implement it into a set of design artifacts, and continuously evaluate them. The iterative process is conducted until effective and acceptable solutions are achieved (March & Smith, 1995; Simon, 1996). Specifically, this phase is composed of three rounds of design efforts. Each round targets at one of research questions and a particular set of design issues (see Table 3-5). The organization of the three stages is shown in Figure 3-2.
Table 3-5 Design and Research Plan

<table>
<thead>
<tr>
<th>Round</th>
<th>Focus</th>
<th>Design Efforts</th>
<th>Evaluation</th>
<th>Deliverable</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RQ1 - Conceptual modeling and architecting</td>
<td>Conflict episode – Vignette, CBR architecture and prototype</td>
<td>Descriptive illustration</td>
<td>Essay 1</td>
</tr>
<tr>
<td>2</td>
<td>RQ2 - Argument detection</td>
<td>Text mining and machine learning</td>
<td>Performance test &amp; Case studies</td>
<td>Essay 2</td>
</tr>
<tr>
<td>3</td>
<td>RQ3 - Conflict measurements</td>
<td>Instruments design and application</td>
<td>Formative assessment</td>
<td>Essay 3</td>
</tr>
</tbody>
</table>

Figure 3-2 Organization of the Design Efforts

Phase 3 – Theory Formulation and Communication

This phase – Theory Formulation and Communication – is a continuous effort that parallels the other two phases. The phase recognizes that design science research involves a recursive process of casting the perceived problem into a class of problems and discovering a general solution by creating an artifact (Hevner et al., 2004; Sein et al., 2011). By consciously reflecting on the problem definition and solution design, I seek to reach a mature design theory about the conflict management approach proposed, including an expository instantiation. A
design theory is about “how to do something” (Gregor, 2006). A mature design theory provides a holistic description about the new artifact in terms of the target problems, theoretical premises, principles of functions and implementation, testable propositions, and potential mutability (Gregor & Jones, 2007; Walls et al., 1992). The final research outcomes of the study will be presented as an initial formulation of design theory that gives explicit prescriptions about how to develop IT-supported conflict management approaches to improve conflict management in distributed teams. Partial theories (Gregor & Hevner, 2013) that reflect the design progress towards the mature theory are also formulated for the purpose of publication and communication.

Design science research needs to be presented to both potential users and academic researchers (Hevner et al., 2004; Peffers et al., 2007; Sein et al., 2011). I present this study and its outcomes to both audiences. The three essays cover the work from three rounds of research efforts and answer the three research questions, respectively. Details of the plans of the three essays are included in the following chapters.
Chapter 4

Essay 1 – CM2: A Conflict Management System

Chapter Summary

This essay introduces a novel conflict management system along with an instantiation called CM2. The system is designed as an aid to understand and manage conflict situations by referring to past situations, which are stored as Vignettes – vivid yet analytical, theory-laden descriptions of past experiences. This essay discusses the theory background of Vignettes and their usage. A meta-model, developed on basis of kernel theories, is created to structure and represent the constructs and progression of a conflict vignette. In line with the idea of case-based reasoning, we design the structure of the proposed system. CM2 implements the storage and case-based reasoning mechanisms. This essay illustrates CM2 with a use scenario, and concludes with discussion about implications, and future work.

This chapter is adapted, updated, and rewritten based on two publications at Conferences (G. Zhang & Purao, 2013) and DESRIST (G. Zhang & Purao, 2014). All uses of ‘we’, ‘our’, and ‘us’ in this chapter refer to coauthors of the aforementioned papers.

Background and Motivation

Conflicts are pervasive in organizational settings (De Dreu & Gelfand, 2008). A conflict refers to an interactive process that begins with incompatibility between interdependent parties (Putnam & Poole, 1987; Thomas, 1992). Prior research points out that conflicts may arise because of differences in goals, differences in ways of working, or interpersonal dissonance (Jehn & Mannix, 2001; Rahim, 2010). Managing conflicts is becoming increasingly important as modern teams become geographically distributed (Hinds & Mortensen, 2005; Mannix et al.,
and projects become large and complex (Alderma, Ivory, McLoughlin, Thwaites, & Vaughan, 2013). Further, contemporary teams often involve participants from different knowledge backgrounds (Geraldi, Maylor, & Williams, 2011), aiming to unleashing creativity. These differences can also sow the seeds for conflicts.

Managing concurrent, prolonged or diverse conflict is challenging for human beings because of inherent limits on our cognitive ability. Our actions are often subject to recency bias and affect, which can lead to undesirable consequences in conflict situations (Kahneman, 2011). The recency bias indicates that people involved in a conflict will more be influenced by recent experiences. The influence of affect suggests that people tend to make irrational decisions with little deliberation when they are emotionally involved in a situation. These and other limitations can lead to inappropriate or even counterproductive decisions and behaviors.

A few tools (in the form of software support) have been proposed for conflict management (Madani et al., 2014; S. Ross, Fang, & Hipel, 2002). These tools either provide communication support (e.g. anonymous messages, procedural support, and voting) (Chiravuri et al., 2011; El-Shinnawy & Vinze, 1998) or offer rational solutions to conflicts (e.g. based on quantitative codification) (Fraser & Hipel, 1981; Madani et al., 2014). Although these types of systems provide a suite of tools to manage the underlying data, they do not address the basic challenge: providing managers the ability to understand a wide variety of conflict situations, and make effective conflict management choices.

To address this challenge, I design and demonstrate a novel artifact – a conflict management system – that can help conflict parties make sense of conflict situations and become aware of conflict management behaviors in a systematic manner. To build the system, we follow a design science research approach (Gregor & Hevner, 2013; Gregor & Jones, 2007; Walls et al., 1992). The system is designed based on a novel approach of capturing and presenting conflict episodes – vignettes, each of which conveys key information about a conflict (e.g. claims, actions and strategies). I identify both descriptive knowledge required to understand conflict phenomenon
and prescriptive knowledge that guides the system architecture, and apply them as the kernel theories of the design. I present the architecture of the new conflict management system, as well as an instantiation of the system, called CM2. The use of the system is illustrated by outlining a use scenario.

**Prior Work and Kernel Theories**

**Conflict and Conflict Management**

Organizational conflict has been intensively studied in the fields of organization, psychology, and communication (De Dreu & Gelfand, 2008; Putnam & Poole, 1987; Thomas, 1992) to investigate a number of components of conflict. The selective review we present below emphasizes findings from prior research; and provides a structure that we can use to understand conflict.

Understanding a conflict starts with examining its *cause*. Prior work suggests three broad types of conflicts: task, process, and relationship (Jehn & Mannix, 2001). *Task* conflicts arise because of incompatible viewpoints and opinions pertaining to essential ideas of the task, such as its goal, logic, and value. *Process* conflicts are triggered between people who have different viewpoints and opinions pertaining to duty, resource delegation, and schedule. *Relationship* conflicts can be traced to adverse emotions and feelings between people. Understanding the cause is important because it can both predict the outcome as well as suggest strategies for mitigation of consequences. For example, prior studies show that task conflicts have the potential to lead to beneficial outcomes while process and relationship conflicts often lead to dysfunctional consequences (Jehn, 1995).
Understanding a conflict situation also means understanding it as a process. Conflicts may be conceptualized following a sequential model, or a more nuanced, punctuated model. The punctuated process model emphasizes events that occur as a conflict unfolds. In line with the punctuated equilibrium theory, the punctuated process model describes the conflict process as alternating between stasis and transition (Gersick, 1988; Putnam, 2004). Here, a transition refers to a short period of time when a conflict shifts substantially in terms of its intensity and its progress toward resolution. A transition is characterized by behaviors and emotions of parties to a conflict. The conflict escalation theory suggests that a conflict may undergo a combination of two types of transitions – conflict escalation and conflict de-escalation (Pruitt, 2008). Conflict escalation occurs when a conflict is intensified with mechanisms such as more extreme strategies, emotional involvement, issue generalization, or increased number of participants. Conflict de-escalation occurs when conflict parties re-assess the conflict situation and change their language and behaviors to more moderate, collaborative ones.

Knowing the roles played by conflict parties is a precondition for understanding their behaviors. The parties involved in a conflict can be broadly classified into two categories: principal conflict parties and third parties. The term ‘principal conflict parties’ refers to individuals or groups who are interdependent with one another in the pursuit of their interests in spite of the incompatibilities between them (Putnam & Poole, 1987). The term ‘third parties’ refers to individuals or groups who intervene into the conflict either to fulfill their duties or because they have a stake in the conflict (Pruitt & Kim, 2004). Understanding a conflict situation, thus, requires recognizing the perspectives and actions of both principal conflict parties as well as third parties who are affected by the conflict.

Prior work also outlines conflict management strategies to characterize the general intention of the principal conflict parties toward a conflict and the approaches they use for coping with the conflict (Thomas, 1992). Scholars have conceptualized conflict management strategies in a space defined in two dimensions: concern for self and concern for others. Combining these two
dimensions results in five styles or strategies that the principal conflict parties can use: collaborating (H-H), compromising (M-M), yielding (L-H), contending (H-L), and avoiding (L-L) (Olekalns et al., 2008). A conflict management strategy, thus, represents a composite construct that characterizes the general orientation of a conflict party, not a specific action. This review of prior work (briefly described above) provides important constructs and kernel theories that guide this research.

**Case-based Reasoning for Managing Conflicts**

Conflict management is a task full of uncertainty. Although conflict management strategies provide high-level explanation of conflict behaviors, knowledge about how to apply these strategies is still scarce. One important reason is that conflict parties’ behaviors are influenced by a very large number of external and internal factors. The relationship between these factors and the behaviors cannot be captured by structured and un-ambiguous rules. A case-based reasoning strategy provides a way to overcome this challenge.

Case-based reasoning (CBR) is different from rule-based reasoning. It is a problem-solving strategy based on previous experiences (Kolodner, 1992; S. Ross et al., 2002). A CBR system refers to a type of expert system or decision support system that applies reasoning based on matching of prior cases to the current situation. The intent is to assist users to understand and solve current problems by referring to similar problems that may have occurred in the past. A CBR system achieves this by storing and indexing a sufficient amount of previous experiences as cases. When a new problem arises, the user can identify past cases and adapt them to address the current problem.

CBR systems can be designed for two potentially overlapping concerns: problem solving and/or problem interpretation (Kolodner, 1992). In the first instance, a CBR system is designed to propose almost-right solutions to new problems on basis of previous problem-solution pairs. In
the second instance, a CBR system assists users to identify previous cases similar to current problem or situation for the purpose of generating deeper understanding and assessment of the current problem or situation. It is possible to analyze the potential usefulness of CBR for conflict management by comparing the problems related to conflict management elaborated in prior research, and the professed benefits of CBR (Kolodner & Simpson, 1989; Main, Dillon, & Shiu, 2001). Table 4-1 summarizes these as the potential for application of CBR to assist users with conflict situations.

To the best of our knowledge, only two CBR systems have been reported in the context of conflict management situations (Kolodner & Simpson, 1989; S. Ross et al., 2002). Both have been designed to identify conflict resolutions within a limited problem domain (Nature Resource Dispute) and therefore, offer little direct help for understanding conflict situations and conflict management behaviors in other domains. Additionally, neither system has incorporated prior scholarly research about conflict and conflict management as kernel theories into the systems. The work we report breaks from these prior efforts because it represents a conscious application of the design science research paradigm and recognition of prior scholarly work as kernel theories that contribute to the design effort.

Table 4-1 Applicability of CBR for Conflict Management

<table>
<thead>
<tr>
<th>Factor</th>
<th>Elaboration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowering knowledge acquisition costs</td>
<td>Reduce cognitive efforts used for maintaining a collecting of conflict management experience/knowledge</td>
</tr>
<tr>
<td>Reasoning shortcut</td>
<td>Avoid repeating all reasoning steps in situations where a repetitive conflict occurs or a long chain of reasoning is required to understand and make decision for the conflict.</td>
</tr>
<tr>
<td>Reasoning focus</td>
<td>Indicate what are essential for understanding and solving conflict and reduce emotional impacts.</td>
</tr>
<tr>
<td>Error prevention</td>
<td>Avoid repeating mistakes made in previous conflict situations by indicating what cause failure in the past.</td>
</tr>
<tr>
<td>Learn over time</td>
<td>Accumulate knowledge about conflict and conflict management over time.</td>
</tr>
</tbody>
</table>
Kernel Theories

Prior literature about conflict management and CBR plays an important role in our system design. The former set of literature provides “descriptive knowledge” that explains the social phenomenon – conflict – and the law, norms, and regularities that guide the development of a conflict. From this knowledge base, we draw descriptive and propositional knowledge that inform us to understand the conflict problem in organizations and clarify design requirements. The latter set of literature presents “prescriptive knowledge,” the ‘how’ knowledge of building the conflict management system. From this knowledge base, we investigate known artifacts and identify the CBR system as the prototype that will be re-designed for conflict management. Both sets of knowledge constitutes the “kernel theories” that conceptualize the design problem – conflict management – and reveal meta-requirements of the design (Gregor & Hevner, 2013). The relationship between the problem, kernel theories, and meta-requirements is shown in Figure 4-2.

Design Problem

- Concurrent, prolonged, and diverse conflicts
- Information overload
- Unstructured reasoning
- Time and cognitive limitations

Kernel Theories

- Process View
- Punctuated Process Model
- Conflict Escalation
- Conflict Management Strategies
- Case-Based Reasoning
- CBR System

Meta-Requirements

- Provide process view
- Pinpoint transitions
- Reveal conflict management strategies
- Capture speech acts
- Provide conflict repository
- Enable case generation and retrieval

Figure 4-1 Design Problem, Kernel Theories, and Meta-Requirements
Defining Vignettes

The foundation for our design science effort is a Vignette, analogous to but different in important ways from a Case. We make this choice because Cases form the basis of CBR systems. In CBR design, a case refers to a set of information that records previous experience or knowledge. The design of a CBR system involves three considerations: case representation, case retrieval and adaption, and case entry and maintenance (Main et al., 2001). Among these, case representation is crucial because it reflects the informational components and structure that lay the foundation of the CBR system.

Case representation is about structuring and formatting previous experience and knowledge in a well-packaged form (Main et al., 2001). Each case can be conceptualized in terms of a problem and a solution. In a CBR system, this problem description and circumstance is codified for the purpose of case indexing and retrieval. Such a design fits situations in which previous solutions can be readily applied to new problems (S. Ross et al., 2002). However, this design has limited capacity to assist users to understand new, hitherto not encountered, problems and develop new solutions. To overcome this limitation, we draw upon and extend the notion of Case, to a new structure we call Vignette.

We define a Vignette as an analytical narrative of practice. It is constructed to convey information about a series of events taken to be representative, typical, or emblematic (Erickson, 1986). As shown in Table 4-2, vignettes are created to inspire thinking and reflection on how and why conflicts occur drawing on theoretical precursors, different from cases, which are constructed to explain what it is. Previous studies point out two features of vignettes – richness and abstraction – that qualify it for analytical purpose (G. Zhang & Purao, 2013).
Table 4-2 Comparison between Case and Vignette

<table>
<thead>
<tr>
<th></th>
<th>Case</th>
<th>Vignette</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purpose</td>
<td>Informative</td>
<td>Analytical</td>
</tr>
<tr>
<td>Description Focus</td>
<td>Solution</td>
<td>Process</td>
</tr>
<tr>
<td>Format</td>
<td>Experience-based</td>
<td>Theory-based</td>
</tr>
</tbody>
</table>

Vignettes are rich because they contain the information necessary for readers to build a picture of a set of events by activating their imagination and interest (Huebner, 1991; Poulou, 2001). Although Vignettes are usually short, they provide adequate information about an experience that is limited in scope. By reading a Vignette, a reader is able to capture the essence of a past experience and develop a sense of “being there” (Erickson, 1986). A Vignette can achieve this by providing moment-to-moment description.

However, even a richly detailed Vignette is a reduced account and an abstraction of the original set of events (Erickson, 1986). It conveys information that reflects and sharpens only some details of the original events and leaves out others. The selection is conducted to heighten some analytical concepts (Erickson, 1986) to stimulate reflection and thinking about these aspects (Angelides & Gibbs, 2006). The construction of vignette heavily, thus, relies on both the experience of creator or experts as well as prior theoretical constructs.

A Meta-Model of Conflict Vignette

To leverage vignettes to our system design, we construct a meta-model of conflict vignettes on basis of constructs and relationships identified in prior research. Constructs represent the information sets that are important for understanding conflict and making conflict management decisions. Incorporating these in the meta-model of conflict vignettes, in essence, has the effect of treating these prior findings as kernel theories (Gregor & Jones, 2007). Table 4-3 shows the set of constructs used in the meta-model.
Table 4-3 Constructs in Conflict Vignettes

<table>
<thead>
<tr>
<th>Construct</th>
<th>Description</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conflict</td>
<td>Conflict is a process in which incompatibility between Claims from Conflict Parties surfaces, and may be resolved.</td>
<td>(Thomas, 1992)</td>
</tr>
<tr>
<td>Cause</td>
<td>Cause refers to the underlying reasons for the incompatibility among Claims.</td>
<td>(Jehn and Mannix, 2001)</td>
</tr>
<tr>
<td>Claim</td>
<td>Claim represents an intrinsic interest, goal, or opinion of a Conflict Party.</td>
<td>(Thomas, 1992)</td>
</tr>
<tr>
<td>Transition</td>
<td>Transition refers to a moment when the nature of the Conflict shifts substantially.</td>
<td>(Gersick, 1988) (Putnam, 2004)</td>
</tr>
<tr>
<td>Conflict Party</td>
<td>Conflict party refers to individuals or groups who are engaged in a Conflict.</td>
<td>(Thomas, 1992)</td>
</tr>
<tr>
<td>Role</td>
<td>Role reflects the perspective of a Conflict Party involved in a Conflict.</td>
<td>(Putnam and Poole, 1987)</td>
</tr>
<tr>
<td>Strategy</td>
<td>Strategy represents the plan that Conflict Parties apply for coping with Conflict.</td>
<td>(Olekalns et al., 2008)</td>
</tr>
<tr>
<td>Action</td>
<td>Action is the behavior enacted by a Conflict Party following a Strategy.</td>
<td>(Rahim, 2010)</td>
</tr>
</tbody>
</table>

The constructs capture the lifecycle of a conflict, i.e., how the conflict may undergo transitions marked as key moments. The model recognizes that a conflict can arise if there are multiple conflict parties, each with a specific role. A conflict may arise, escalate and may be resolved based on the claims that each conflict party holds. To resolve a conflict, the conflict parties may employ various strategies based on their role and their interpretation of the conflict. These strategies are then enacted through actions, as the conflict unfolds. These actions may include directly arguing for or against specific claims, querying, informing and proposing solutions. As conflict parties interact with one another through a sequence of actions, the accumulated sequence of actions can sometimes lead to substantial shifts, i.e. transitions in the conflict. Figure 4-3 shows the meta-model we use to structure the vignettes, which includes constructs as well as relationships among the constructs.
CM2: A System for Reasoning with Conflict Vignettes

The conflict management system we designed is called CM2, i.e., Conflict Management System based on Computer-Mediated Communication. CM2 represents a class of conflict management systems that leverage case-based reasoning for conflict management in distributed teams. Conflict vignettes provide the foundation of the system. Our design is also inspired by the fact that computer-aided communication (CMC) has been frequently used in distributed teams (Gibson & Gibbs, 2006). The increasing use of CMC produces a large amount of communication data that is rich in interaction, opinion, and emotional information. This provides an opportunity of overcoming the constraint in terms of data availability that has limited prior work on conflict management system development.

CM2 is designed in accordance with the guidelines for CBR system design (Main et al., 2001). An overview of the system architecture of the CM2 system is shown in Figure 4-3.
In the core of the system is the vignette base, a repository for storing and organizing conflict vignettes. The vignette base is the memory of previous conflict episodes. It is designed as a relational database. Conflict vignettes are stored in a way that they are decomposed into a set of information and loaded into multiple tables. The vignette base is not static, rather evolving. While the base provides ingredients for making sense of new conflict situation and formulating “knowing” to solve the conflict (G. Zhang & Purao, 2015), the new conflict is captured and added back into the base right after it is resolved or terminated, which in turn serves to cope with later conflict situations.

Indexing conflict vignettes is necessary for searching and accessing relevant conflict situations. A conflict vignette is an analytical account of a conflict situation. Instead of relying on problem descriptions only, we apply multiple features, varying from conflict parties’ behaviors to conflict transitions, to index conflict vignettes. The goal is support users to analyze conflict vignettes from different perspectives.

Figure 4-3 System Architecture – CM2
Vignette Extraction

The more conflict vignettes it contains, the more likely it can provide relevant references to users. While electronic communication produces a large amount of raw data, such data is not directly ready for use. To implement the designed system, 30 conflict vignettes have been created by manually examining, analyzing and coding a set of CMC data collected from an online community. While the creation of these vignettes validates the design of conflict vignettes and the feasibility of utilizing CMC data, it also shows that manual processing is a daunting task for end users. Automatic or semi-automatic approaches are required to help users to generate more conflict vignettes through parsing and analyzing CMC data.

Based on our manual processing, we identify two essential steps for generating conflict vignettes from CMC data: text annotation and construct generation. Accordingly, design focus at the back end side is to work out a CMC text analysis solution and a set of heuristics and machine learning mechanisms to automatize the two steps, respectively. With regard to text annotation, prior studies show that CMC text not only contains rich information but also is analyzable. Techniques are available for identifying and visualizing interaction patterns between participants, analyzing the affect content in free text, and determining message authors’ attitude and opinion (Abbasi & Chen, 2008). They have the potential replace part of CMC text annotation effort that we applied in the manual processing.

Heuristics are needed to connect annotated text to conflict vignettes constructs. Annotated text conveys atomic information expressed in each message, such as action and interaction information. Such information needs to be aggregated before generating conflict vignettes. For example, we need review all actions performed by a participant to identify his/her role in a conflict and his/her strategy toward the conflict. Such heuristics have been generated and applied in our manual processing. The following design efforts include revising and formalizing
these heuristics and adding machine learning mechanisms to address uncertainties and ambiguities. Details about the design will be discussed in Essay 2.

Vignette Retrieval and Representation

At the front end are the modules that interact with users to retrieve and display conflict vignettes. The direct manipulation user interface is linked to modules for case retrieval and case display. The case retrieval module provides search options, such as keyword search and condition-based search. The retrieved conflict vignettes can be presented in either textual or graphic format. For instance, the progress of a conflict and its transitions are presented via a “timeline” in CM2.

An Instantiation of the CM2 System

In order to show case the design of the proposed conflict management, we develop a prototype of the CM2 system. Currently, the prototype includes modules that interact with users to retrieve and display conflict vignettes. As shown in Figure 4-4, the vignette retrieval module provides two search options, keyword search and condition-based search. The retrieved conflict vignettes are presented in plain text format, along a visualization that depicts conflict process via timeline. The appearance of the timeline draws upon Timeline and TimePlot, two widgets proposed by CSAIL (Karger, 2008). The scale and interval used in our timeline is calculated based on the number of total messages exchanged for a conflict considering that computer-mediated communication is often asynchronous (Sack, 2000). Following these decisions, information about the conflict vignettes was structured and stored in the form of vignette base that was constructed on basis of a temporal meta-model of conflict. The base is now populated
with the 30 manually generated vignettes. Vignette generation functions that simulate the manual entry process are being designed.

![Screenshot of CM2 – Main Window with Search Results](image)

**Illustrative Example**

To demonstrate the application of the CM2 system, we follow a use scenario that describes how conflict is managed without and with the proposed system.

**Scenario**

Grant is the leader of an online community aimed at developing and maintaining open source software tools. Members of the community come from a number of backgrounds. These members, developers and users, use a mailing list to exchange ideas and make decisions about software design. Conflicts arise when, for instance, developers propose incompatible software designs or users express discontent with software changes made by developers. After a new version of a software is released, Grant observes multiple discussions in the mailing list that provide pointers to such conflicts. Grant realizes that he needs to better understand and manage these conflict situations. He senses that this
is not an easy task not only because multiple conflicts are going on in parallel but also because of the lack of clues about how these conflicts would evolve, and how different participants would react.

The Scenario without CM2

Without any external assistance, Grant can only draw on his experiences to envision these conflicts, anticipate participants’ reactions, and choose strategies for his actions. It is challenging to develop a reasonable vision of each conflict partly because of the sheer number of conflicts, and partly because he cannot always recall similar conflict situations and other participants. As a result, his understanding and decision making is largely influenced by his recent experiences, which may be misleading. For example, a recent conflict that resulted in a favorable experience may cause Grant to develop an optimistic attitude to all of the conflicts he is now facing. This may then cause him to neglect or pay insufficient attention to the conflicts. Another potential mistake may be to misjudge the orientation of a conflict party. Unfavorable experience with an opponent in a recent conflict may lead Grant to be more aggressive when he interacts with the conflict party, even if the other party tries to adopt a collaborative strategy.

Re-visioning the Scenario with CM2

Using CM2 can help Grant overcome challenges mentioned above. CM2 not only provides a repository of previous experience but also helps to organize the reasoning process. Without knowing what kind of conflicts he wants to review, Grant can start his reasoning by looking at what occurred in the past. As shown in Figure 4-4, the four list panels show the available selection criteria that can be used to search conflict vignettes. Each of the criteria has at least one conflict vignette that matches it.
With the assistance of these criteria, Grant can start his reasoning by categorizing the new conflicts he faces and prioritizing them. Let’s assume that Grant decides to start with handling task-related conflicts first, considering that they are more relevant to their development effort. He is interested to know how a typical task conflict evolves, how it can be solved in a collaborative way, and how third parties contribute to the resolution of a conflict. With these clues, Grant specifies the search conditions and searches for conflict vignettes. Figure 4-4 also shows the search results.

For each conflict, a Vignette is generated by using key information about the conflict narratives stored in the repository. After considering the retrieved vignettes, Grand may identify a few conflict Vignettes that are most appropriate to the ongoing conflict he is seeking to resolve. He can investigate the selected, most relevant Vignettes by examining them more closely. The CM2 system will display all details about the conflict (see Figure 4-5) including parties, claims and transitions. Grant can drill down into each transition and examine how the conflict progressed to gain a better understanding and to assess whether the lessons found in the Vignette may be applicable to the ongoing conflicts.

With the CM2 system, we envision that Grant would be able to manage conflicts in a more organized way. The vignettes would provide Grant a place to start his investigation when new conflict arises. The kernel theories behind the system would provide a set of criteria that Grant can use to categorize and understand new conflicts. Biases such as recency (and others) will be minimized because Grant would be able to access to a large base of vignettes along with pointers to those that might potentially be useful. With the help of CM2, Grant would be able to develop a more realistic vision of new conflicts by referring to relevant conflict situations from the past. Grant would also have a better understanding of other conflict parties, especially those he may not be familiar with by looking at their pattern of actions in the Vignettes. This insight would allow Grant an opportunity to adjust his strategy and tactics when he approaches them.
Conclusion

Conflict is a part of the modern, distributed workplace that relies on a geographically distributed and diverse set of knowledge workers. However, conflict does not need not be the “dark side” of the workplace. Instead, it can be beneficial when appropriately managed (Jehn, 1995). To assist managers to limit the negative aspects of conflict while promoting its positive aspects, we have proposed the design of a case-based conflict management system. The system, called CM2, can assist conflict parties to understand conflict situations and make better decisions by referring to experiences gathered from past conflict situations. The foundation for the design we have proposed is the notion of a conflict Vignette. We have defined a Vignette as similar to, yet distinct in important ways from a Case in that it is analytical and theory-laden. It represents key information pertaining to a conflict, such as cause, claims, transitions, actions, strategies, and
other relevant information. These constructs are identified following kernel theories about conflict management in prior research.

The design we have outlined has the potential to make several contributions to theory and practice. First, the CM2 system, to our best knowledge, is the first system that tries to structure how assistance may be provided to managers for more effective conflict management behaviors. Second, the design we propose builds on a meta-model of conflict Vignettes, which can be applied for capturing and representing vivid yet analytical descriptions of conflict episodes. The notion of a Vignette is, in itself, an important contribution because it suggests a new structure that provides a foundation for case-based conflict management system. Further, it also provides an approach that can be leveraged for capturing and representing theory-laden episodes in other domains beyond conflict management (G. Zhang & Purao, 2015). Third, we claim that practitioners may benefit from CM2 in that it can improve their conflict management strategies, and may eventually develop more constructive attitudes toward conflict.

We acknowledge that there are some limitations to our design. First, the design effort and outcome reported in this paper can contribute only a “partial theory,” that consists of constructs and a model along with an early instantiation. We describe this as the initial step in a journey toward a mature design theory (Gregor & Hevner, 2013). Evaluation of the proposed system is required to validate the design and understand the phenomenon in which the design is embedded. Second, as mentioned early, automatic or semi-automatic approaches are required to help users to generate key information as specified in this paper through parsing and analyzing the raw communication data. A formalized methodology for this process along with supportive NLP techniques needs to be designed and implemented. Third, as more conflict vignettes are populated into the system, users will also face the new challenge of locating the vignettes that are most appropriate and informative for current conflict situation. Some matching functions are necessary for enhancing performance of the system. Candidate approaches, such as nearest neighbor
retrieval, inductive approaches, and knowledge guided approaches can be analyzed and added to enhance the system further. These are part of our future research agenda.
Chapter 5

Essay 2 – Argument Detection in CMC: A Theory Based Approach

Chapter Summary

The rapid growth of online communication has dramatically changed the manner in which arguments take place. Participants in virtual teams and communities continue to face challenges to detect and make sense of arguments when the constituent elements of an argument are scattered in prolonged online discussions. Few methods or tools are available for the detection of arguments from available sources. This chapter develops a theory-based argument detection model. Drawing on the argumentation theory, I propose a model for argument detection. It is composed of features that reflect five categories of argumentation functions, including: announcement; reasoning; modality; transition; and, affect, with the addition of a language feature that is informative for recognizing arguments. The evaluation results show that the model achieves higher accuracy and recall in detecting arguments in message sets compared to baseline models. The paper also presents an illustrative example to show the practical application of the model.

This chapter is adapted, updated, and rewritten based on a publication at AMCIS (G. Zhang, Zhou, Purao, & Xu, 2016). All uses of ‘we,’ ‘our,’ and ‘us’ in this chapter refer to coauthors of the aforementioned paper.

Background and Motivation

Arguments are part of all collaborative activity. With the pervasiveness of online communication tools and changes in work practices, much collaborative activity is now conducted online. Within and across organizations, geographically dispersed teams rely on email,
instant messaging, and social media to support collaboration, which includes engaging in arguments, handling conflicts, and making decisions (Hinds & Mortensen, 2005). Beyond organizational work, online communities enable mass debate (Somasundaran & Wiebe, 2010) and mass collaboration in domains such as open source software development (J. Wang et al., 2015) and product development (Brabham, 2008).

The use of online discussion platforms presents new challenges for participants to make sense of arguments because the nature of the dialog in these forums is different. For example, talking allows interaction where participants can develop their arguments in an ongoing manner. Written arguments via formal communication such as articles or essays allow asynchronous interaction, where participants deliberate to develop arguments and deliver them at once. Online discussion has been compared with both. It is similar to talking with regard to the length and information content of each message. It is similar to writing with regard to pace (Yates, 1996). As an online discussion progresses with multiple participants and several threads, it can generate a large amount of messages. Within these threads, arguments can be obscured – they may be interspersed with other conversation or they may simply be abandoned. This can prevent participants from tracking arguments as they develop, and from returning to prior conversations to understand how arguments develop and can be managed (Barcellini, Détienne, Burkhardt, & Sack, 2005; Bex, Lawrence, Snaith, & Reed, 2013).

Although the challenge has drawn more attention in recent studies (e.g. Bex et al. 2013; Lippi and Torroni 2015; Schneider 2014), few methods and tools are directly aimed at extracting arguments from the data available via online discussions. One reason for this may be that online communication tools – including group decision support systems – are rarely equipped with the capabilities necessary for processing argumentative content (G. Zhang & Purao, 2014). Another is that the existing argument processing tools are focused on analyzing the argument structure in articles or essays (Moens, Boiy, Palau, & Reed, 2007; Ozyurt, 2012) which tend to be different from online discussions in both vocabulary and structure (Yates, 1996). Further, these models
tend to be inconsistent in feature selection and lack robustness due to the lack of strong theoretical foundations. To address these shortcomings in prior work, and to address the contemporary set of problems outlined above, the research objective of this study is to design a model for detecting argument in online discussion based on argumentation theories.

To achieve the objective, we propose an automatic argument detection solution, which serves as the foundation for making sense of arguments in online discussion. Specifically, we propose a Theory-based Argument Detection (TAD) model to effectively detect argumentative messages in online discussion. Drawing upon the argumentation theory (Eemeren, Grootendorst, & Henkemans, 2002), TAD is built upon features that reflect five categories of argumentation functions: announcement, reasoning, modality, transition, and affect, as well as another feature that is informative for recognizing argumentative language. Machine learning techniques are applied for classification. To evaluate TAD, we create an annotated argumentation corpus using data collected from Bugzilla. The evaluation results show that TAD achieves higher accuracy and recall rates in detecting argumentative messages, as compared to baseline models. The paper also presents an illustrative example to show how the model detects arguments in online discussions.

The rest of this paper is organized as follows. The next section reviews the argumentation theory and relevant prior works in argument detection. In section three, we describe the design framework of TAD. We then describe the corpus and experiment used to evaluate the performance of TAD, followed by the example. We conclude the paper with a discussion of the implications and limitations.
Related Work

Argumentation Theory

Argument and argumentation are two concepts that have been used interchangeably in previous work (Lippi & Torroni, 2015; Mochales & Moens, 2011). In this study, we distinguish argumentation as the action or process of reasoning that is systematically organized in support of an opinion. In contrast, an argument is a piece of text or utterance that is made in the process of argumentation. Unlike in articles and essays (where all of the arguments in an argumentation are included in a single document), the differentiation between argument and argumentation is critical in the context of online discussions because the arguments of an argumentation are often dispersed across a conversation, i.e. across several distinct messages.

Eemeren et al. (2002), describe argumentation as the means of resolving a difference of opinion. It is performed when one party’s opinion meets with doubts from the other party and both parties have the intention to discuss the matter and address the difference (Thomas, 1992). Argumentation theory arose in response to the need of understanding the actual argumentation practice in conversation (Eemeren et al., 2002). In contrast to the logician’s view, which focuses on abstract forms and patterns of argumentation (Toulmin, 2003; D. Walton, Reed, & Macagno, 2008), argumentation theory concentrates on rules and guidelines in constructing, detecting, and analyzing arguments. It has been employed to promote critical thinking and scientific argument (Bricker & Bell, 2008), and recognize effective communication pattern in online environments (Clark & Sampson, 2008).

A complete argumentation is composed of three elements: proposition, standpoint, and reasoning. A proposition is the matter of interest, which can be a description, a prediction, a judgment, or a suggestion. With respect to a proposition, one can take a positive standpoint to justify the proposition, or take a negative standpoint to refute the proposition, or take a neutral
standpoint. An argumentation is strengthened when some kinds of reasoning are provided to justify the standpoint (Eemeren et al., 2002). In a conversation, the three components need not all be included in a single statement. For example, reasoning might not be included at the beginning of a discussion, while the proposition and the standpoint of the parties are often left out as the conversation moves forward. Eemeren et al. (2002) stress that verbal expressions provide strong indicators of argumentation and its elements. They provide an initial list of vocabularies that signal standpoint and reasoning, which however, need to be enriched and refined for the purpose of automatic argument detection and processing.

**Argument Detection Models**

Argument detection in online discussion is still understudied. Relevant studies are mostly found in the field of argumentation mining, aimed at detecting, decomposing, and reconstructing argumentation in textual materials (Lippi & Torroni, 2015; Palau & Moens, 2009; Schneider, 2014). With respect to argument/argumentation detection, few models have been proposed (see Table 5-1). A quick review shows that most of these models are developed based on and for documents like articles and essays. Their applicability for online discussion is questionable considering the differences in vocabulary and sentence structure (Yates, 1996). One exception is the work by Levy et al (2014). Their model aims to detecting claims in online debates. Since all messages are argumentative by their nature, the model might not be applicable for general online discussion. Another important aspect of existing models is that they rely heavily on classic features for text representation, such as bag-of-words, word bigrams/trigrams, part-of-speech information, and text statistics (A. Conrad, Wiebe, Hwa, & Rebecca, 2012; Palau & Moens, 2009; Stab & Gurevych, 2014). A few exceptions incorporate features such as sentiment analysis indicators (A. Conrad et al., 2012) and subjectivity scores of sentences (Levy et al., 2014). A third observation is that while word indicators have been recognized and applied in these models, there
is little to no consistency in regard to what features should be included for analyses. Lexicons in these models vary with regard to size, part of speech, and composition. A final observation is that features included in these models mainly represent researcher experience and peculiarities of the corpora, instead of incorporating broader theories about arguments and argumentation, which may improve and support the design of argument detection models.

Table 5-1. Argument Detection Models in Prior Work

<table>
<thead>
<tr>
<th>Works</th>
<th>Corpus</th>
<th>Machine Learning Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moens et al. 2007</td>
<td>News Articles, Legal Text, Reports</td>
<td>Generic Features: punctuation, text statistics, n-grams</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Special Lexicons: verbs, adverbs, modal auxiliary, rhetorical relations</td>
</tr>
<tr>
<td>Palau and Moens 2009</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mochales and Moens 2011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ozyurt 2012</td>
<td>Biomedical Literature</td>
<td>Generic Features: POS, position of sentence, semantic network</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Special Lexicons: predict noun, head word</td>
</tr>
<tr>
<td>Conrad et al. 2012</td>
<td>Blogs and Editorials</td>
<td>Generic Features: sentiment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Special Lexicons: relation phrases, noun, verb</td>
</tr>
<tr>
<td>Levy et al. 2014</td>
<td>Debate Forum</td>
<td>Generic Features: cosine similarity between sentences, subjectivity score, sentiment, name entity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Special Lexicons: claim words</td>
</tr>
<tr>
<td>Stab and Gurevych 2014</td>
<td>Persuasive Essays</td>
<td>Generic Features: text statistics, location of sentences, punctuation marks, n-grams, POS n-grams,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Special Lexicons: verbs, adverbs, pronoun, reasoning markers</td>
</tr>
</tbody>
</table>

**Design Framework**

To address these challenges, this study proposes a Theory-based Argument Detection (TAD) model. Figure 5-1 show the framework of TAD. TAD model takes discussion text as input. Standard text pre-processing steps including tokenization and stop word removal are adopted. The core component of TAD is feature extraction. Specifically, TAD relies on two sets of features to classify argumentative messages in online discussion. One set of features is built upon the argumentation theory. It includes vocabularies that reflect the five categories of
argumentation functions. The other set include language features that are informative for recognizing argumentative language. These extracted features are then fed to a number of classification algorithms to train and tune the final model. Popular text classification algorithms, including Support Vector Machines (SVM), Decision Tree (J48), Naïve Bayes, and Logistic Regression, were evaluated. We focus our discussion on the core component: the construction of argumentation theory based features and language features.

![Diagram of Theory based Argument Detection Model](image)

Figure 5-1. Framework of Theory based Argument Detection Model

**Argumentation Theory Based Features**

A comprehensive set of text indicators is critical for detecting argumentative messages in online discussion. Drawing upon and extending vocabularies identified in (Eemeren et al., 2002), we develop a fine-grain argumentation lexicon that reflects five categories of argumentation functions.

**Announcement.** Typing to speak is constrained by space, time pressure, and available attention. People, thus, tend to express their opinions in a clear, straightforward language when they involve in online discussions (Yates, 1996). To join a discussion, one often starts with an announcement of his/her standpoint. Generic phrases that can be used for announcing standpoints
include *in my opinion, I think, I assert, I vote*, and so on. While there are comparatively few
indicators of positive standpoint, negative standpoints can also be announced with phrases like *I
debate, I doubt, and I disagree*. Verb indicators might also appear as nouns, e.g., *My assertion is*
and *The suggestion is*.

**Reasoning.** To make argumentation sound and cogent, announcements of standpoint
need to be supported with reasoning, such as providing evidence, making cause-effect analysis,
delimiting the scope of the problem, and making comparison/evaluation. They provide reasons
and premises that can be used independently or combined to support the standpoint (Toulmin,
2003). Indicators of reasoning includes phrases such as *for example, demonstrate, because, so
that, respect, circumstance, more, than, and superior*.

**Modal Qualifier.** An argumentation can be made with total conviction or, at the other
extreme, it could be vague and uncertain in terms of either agreement or doubt. An experienced
debater or negotiator is often good at presenting a mix of arguments that vary in force to highlight
his or her firmness on some points (Joni and Beyer 2009). Words that indicate the strength of an
argument include, *such as, absolute/absolutely, always, maybe, must, possible/possibly, perhaps,
sometimes*, and so on. These words reflect one’s assertion as to standpoint and confidence with
the stated evidence and reasons.

**Transition.** Transition words are essential for organizing the elements of argumentation.
They serve to connect parts of the argument and introduce shifts. One important technique of
argumentation – concession – also relies on transition words. By admitting some of the points
claimed by one’s opponents, it is possible to build a cooperative relationship with the opponents
and promote reciprocity (Johnson & Cooper, 2009). Verbal indicators of transition include
coordinators and conjunct adverbs, such as *and, or, first, and finally*, establish links between
clauses and argument components. Subordinators, such as *although and despite*, are strong
indicators of concession.
**Affect.** Heated debate can lead to personal attack with the use of strong affective language. This is especially the case in virtual contexts, where participants are less confined by the social norms well-established in face-to-face contexts (Johnson & Cooper, 2009). Offensive and abusive languages might emerge when a debate escalates. Obscene language with words such as *f***, *damn, stupid*, and *hell* often appear in messages sent by “flamed” parties. Similarly, words such as *annoying, upset*, and relevant emoticons are often used in messages to convey angriness, disappointment, or other negative attitudes toward the argument or the opponent.

Following the categories described above, an initial argumentation lexicon was created with words identified in previous studies and web resources. Particularly, announcement and reasoning indicators were extracted from the seminal work by Eemeren and his colleagues (2002) and verb/nouns lexicons used in previous models. Modal qualifiers and transition words are developed based on the lexicon of relational phrases (Knott & Dale, 1993). Offense and affective words are extracted from a lexicon for offensive language detection (Chen, Zhou, Zhu, & Xu, 2012) and other online resources.

**Language Features**

Another set of five features was incorporated into TAD based on the linguistic features identified in previous studies (Chen et al., 2012; Mochales & Moens, 2011; Stab & Gurevych, 2014).

**Contextual Lexicon.** The Argumentation Lexicon created above is context-independent. It includes vocabularies that represent basic argumentation functions. Similar to other text mining tasks (Chen et al., 2012), understanding the conversation context is also important for argument detection. In this study, TAD incorporates a set of vocabularies specific for the dataset collection.

---

from Bugzilla. These vocabularies relate to controversial topics (e.g. usability, performance, and easy use of Firefox), competitors (e.g., Chrome, Google, and IE), and abbreviations of modal qualifiers, such as IMHO (in my humble opinion) and FWIW (for what it's worth).

**Sentence Style.** In online discourse, it is a common practice that speakers use special punctuation and word formats to indicate feelings or speaking volume. Punctuations, such as repeated exclamation marks, or questions marks (or a combination of both) and words with all uppercased letter often appear in arguments of strong tone. Occurrences of these patterns are captured in TAD to represent sentence style.

**Sentence Structure.** Sentence function indicates a speaker’s purpose in uttering a specific sentence, phrase, or clause. In arguments, combinations of various sentence functions yield different intentions or different argumentation levels. For example, intensive use of interrogative sentences is observed when one refutes another’s argument. Explanative and imperative sentences often appear in strong persuasive arguments. Ratios of the four types of sentence are included in TAD.

**Sentiment.** Sentiment analysis has been widely used to discover a person’s attitude or opinion (negative, neutral, and positive) toward a particular topic. In online discussion, the sentiment of a message can be interpreted as one’ attitude toward the messages before the current one. To incorporate sentiment information, we incorporate both positive and negative scores for message sentiment. The rationale is that an argument can be made either in favor of or in conflict with statements made in previous messages.

**System Information.** To facilitate communication and reduce confusion, communication platforms provide functions specifying reply-to messages, quoting previous messages, adding attachments, and so on. These functions have been intuitively employed in online discussions. For example, by explicitly indicating the reply-to messages, one can deliver a targeted argument refuting a previous argument or providing supplementary argument. Therefore, automatically
generated system information on Bugzilla including reply-to indicator, quotation, create attachment, and comment on attachment, are included in TAD.

Experiments and Evaluation

This section describes the results of experiments conducted to evaluate the performance of the TAD model in argument detection.

Dataset

Annotated corpora for argument detection are still in short supply (Lippi & Torroni, 2015). As shown in Table 1, the few existing corpora are mainly based on articles and essays. To our knowledge, there is no argumentation corpus for online discussion. In this study, we created a corpus using email list data collected from Bugzilla. We selected this data source for three reasons. First, arguments are a common phenomenon in its email lists (Wang et al. 2015). Second, it represents a sample from a much larger data source that can be further employed to create a larger argumentation corpus. And third, it represents a corpus that is relatively easy to access. Based on email threads identified in Wang et al.’s study (Wang et al. 2015), we selected 7 email threads, consisting of 671 messages that constitute the corpus. The length of one message ranges from 1 to 734 words, with the median of 50 words. The time frame of each thread ranges from 1 month to 4 years, with the median of 10 months.

Two undergraduate students were hired to label messages in the corpus. Both annotators are native English speakers with a background in information systems. The annotators first underwent several rounds of training to understand the argumentation lexicons and features and to become familiar with topics on Bugzilla. They then each independently annotated the messages with regard to whether it was augmentative. Disagreements were resolved via discussions. The
annotation results showed that out of 671 messages, 396 were argumentative and 275 were non-argumentative.

**Feature Extraction and Adopting TAD Model**

During text pre-processing, we parsed all discussion text, removed stop words, and noise (such as unrecognized symbols and wrongly encoded characters). Stanford CoreNLP (Manning et al., 2014) was then used for tokenization. Following our feature extraction model, we extracted all features that appeared in our argumentation lexicon. The same practice was performed to extract features that appeared in contextual lexicon. Furthermore, sentence structure and style features were extracted and calculated during processing to support language features. To extract sentiment features, we adopted the opinion lexicon provided by Hu and Liu (2004) and generated a positive and a negative sentiment score for each message. All these features were then used in text classification algorithms. A 10-fold validation was used across all models to select the best parameters for each classification algorithm.

**Experiment Setting**

To evaluate its performance, we compared TAD against two baseline models: Bag-of-Words (BoW) and 2-gram. Both baseline models are popular in text classification. The purpose of the experiment was to assess whether TAD was able to identify and extract arguments from the corpus more effectively compared to baseline models. Before feeding it to the baseline models, the dataset was processed through the same procedure described above. Four text mining algorithms were tested with baseline models and TAD, including Decision Tree (J48), Naïve Bayes, Logistic, and Support Vector Machine.
**Baseline 1.** Our Baseline 1 is a most popular text representation method, Bag-of-Words approach. It uses each appearing word as features to represent textual data, disregarding grammar and order. Word frequency, excluding stop words, is then used for classification. Special content such as quotes and URL links are replaced with special markers, e.g. “#quote#” and “#url#”.

**Baseline 2.** Baseline 2 uses a 2-gram approach, which is an improvement on Baseline 1. The 2-gram approach captures all sequences of 2 words in a text and is thus able to retain certain contextual information. Similar to Bag-of-Words, it does not concern grammar or order between the tokens. The frequency of each 2-gram token is used as a feature value in classification.

**Evaluation Metrics**

In line with previous studies in text mining (Abbasi, Albrecht, Vance, & Hansen, 2012; Chen et al., 2012), standards evaluation metrics were used to evaluate the performance of the tested models. Specifically, overall accuracy is represented in the percentage of messages (both argumentative and non-argumentative) that are correctly classified. Precision represents the percentage of identified messages that are, in fact, argumentative. Recall represents the percentage of truly argumentative messages that are correctly identified as such. F-measure (Chen et al., 2012) represents the weighted harmonic mean of precision and recall, which is defined as: $F = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$. ROC area (Abbasi et al., 2012) represents the ability of the model to correctly classify argumentative and non-argumentative messages.

**Results**

The experiment results are summarized in Table 5-2. We first tested baseline models with different classification algorithms and both of them achieved the best performance with Support Vector Machine (SVM). Thus, we present the baseline models with SVM. Bag-of-Words
(baseline 1) performed better than the 2-gram model (baseline 2) on all measures, achieving a 0.77 F-Measure. The TAD model also achieved the best performance with SVM. It achieved 82.7% accuracy, 87.9% recall and 0.86 F-measure. The F-measure is an 11.6% improvement from the best performing baseline model (0.77).

The improvements in these four metrics are tempered by different learning mechanisms. The Accuracy scores improve from 76.3% (baseline 1) to 82.7% attributable to TAD+SVM. Corresponding improvements for Recall scores show a move from 66.9% (baseline 1) to 87.9%, also attributable to TAD+SVM; for the F-measure score, a move from 0.77 (baseline 2) to 0.86, again attributable to TAD+SVM; and for the ROC area measure, a move from 0.78 (baseline 1) to 0.88, attributable to TAD+Bayes.

Table 5-2. Comparing the Performance of TAD Models against Baselines

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>ROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 1</td>
<td>76.3%</td>
<td>90.4%</td>
<td>66.9%</td>
<td>0.77</td>
<td>0.78</td>
</tr>
<tr>
<td>Baseline 2</td>
<td>69.4%</td>
<td>89.3%</td>
<td>54.8%</td>
<td>0.68</td>
<td>0.73</td>
</tr>
<tr>
<td>TAD +Bayes</td>
<td>74.2%</td>
<td>90.0%</td>
<td>63.4%</td>
<td>0.74</td>
<td>0.88</td>
</tr>
<tr>
<td>TAD +Logistic</td>
<td>73.5%</td>
<td>81.5%</td>
<td>71.2%</td>
<td>0.76</td>
<td>0.76</td>
</tr>
<tr>
<td>TAD +J48</td>
<td>80.9%</td>
<td>81.6%</td>
<td>87.4%</td>
<td>0.84</td>
<td>0.85</td>
</tr>
<tr>
<td>TAD+SVM</td>
<td>82.7%</td>
<td>83.7%</td>
<td>87.9%</td>
<td>0.86</td>
<td>0.82</td>
</tr>
</tbody>
</table>

In general, all models built using TAD achieved similar or better performance than the two baseline models. We also notice that the two baseline models obtained better precision rates. However, they suffered significantly from low recall rates. The TAD model improved the two baseline models by 31.3% and 60.4% on recall. In another words, the baseline models missed a number of argumentative messages which increased false positive. Consistent with the strategy of maximally argumentative interpretation (Eemeren et al. 2002), we value recall rate over precision rate. Since the model potentially provides decision and analysis support for humans, we are more interested in detecting arguments. The cost of missing an argumentative message is higher than classifying a normal message as argumentative.
An Illustrative Example

To illustrate the application of TAD, we show the results as a visualized argument following the manual processing procedure described in (Barcellini et al., 2005). One email thread ("bug thread") in our corpus was selected for the illustration. Figure 5-2 provides a visual representation of the discussion. Each node represents a message, red for argumentative and while for non-argumentative. A directed line shows a reply-to relationship from start node to the end node. It should be noted that the argument feature of a message is predicted by TAD, which obtains accuracy of 89% on this bug thread.

With a quick glance, we can find that the email thread involves several sub-threads. Based on the premise that threads that attract more discussion carry more information, we investigated in eight sub-threads. Table 5-3 provides a summary of each.

Based on an examination of the figure, the first three sub-threads seem to be argumentative. Upon investigation of the data, we find that they were normal discussions with few arguments. The false alarms were caused by the fact that TAD does not work well with messages at the beginning of a discussion. The fifth sub-thread, as the application predicts, is not an argumentative one. Sub-threads 4, 6, 7, and 8 uncover a conflicted issue under argument. The conflict is about the proposal to unify browser behavior so that Firefox always opens web pages in either new tabs or new windows regardless of where the request is initiated. An add-on solution was suggested to those who were against the proposal but was not accepted. The proposal received continuous criticism. Issues like breaking current use patterns (#7) and losing flexibility in the control of browser behavior (#8) were debated. By going through all of the messages in the bug, we also noticed that one argument thread, messages 60 to 63, was not captured by the application because those messages were not connected to the main thread.

---

2 The source data can be retrieved at: https://bugzilla.mozilla.org/show_bug.cgi?id=324164
Figure 5-2. Visualizing an Argument, A Bug Discussion on Bugzilla

Table 5-3. Summary of Sub-threads in a Bug Discussion on Bugzilla

<table>
<thead>
<tr>
<th>#</th>
<th>Sub-Threads</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5-6*-15*</td>
<td>Clarify what to do to unify the browser behaviors.</td>
</tr>
<tr>
<td>2</td>
<td>18-21*</td>
<td>Propose a patch code</td>
</tr>
<tr>
<td>3</td>
<td>24*-25-26</td>
<td>Discuss a proposed patch code</td>
</tr>
<tr>
<td>4</td>
<td>28-43-45-48</td>
<td>Debate on the add-on solution to be used by those against</td>
</tr>
<tr>
<td>5</td>
<td>29-33</td>
<td>Remind to update user document</td>
</tr>
<tr>
<td>6</td>
<td>37-38</td>
<td>Debate on the rationale of the change</td>
</tr>
<tr>
<td>7</td>
<td>49-50</td>
<td>Argue against the change – breaking previous use patterns</td>
</tr>
<tr>
<td>8</td>
<td>51-52</td>
<td>Argue against the change – losing flexibility</td>
</tr>
</tbody>
</table>

*Notes: Message 6, 15, 21, and 24 are mistakenly classified as argumentative message.*

The detailed examination of the argument visualization, combined with an investigation of the data suggests that TAD did, in fact, provide the ability to see the structure of an argument with both greater confidence in the way that argumentative messages were classified, and in the ability to distinguish between argumentative and non-argumentative messages (see Table 5-2 above).
Conclusion

In this paper, we proposed a theory based argument detection model, TAD, which can effectively detect argumentative messages in online discussions. TAD is built upon (a) features that reflect five categories of argumentation functions that are useful for recognizing argumentative language, and (b) leveraging language features drawn from classical text mining techniques. The empirical evaluation results show that the use of TAD models can achieve higher Accuracy and Recall rates in detecting argumentative messages compared to baseline models. The example shows the application and usefulness of TAD in making sense of conflicts in online discussions.

The study has the potential to make several contributions to research and practice. To the best of our knowledge, TAD is the first model built upon features of argumentation theories, combined with classical text mining functions. It also includes the first fine-grained lexicon of argumentative language in online discussion. The lexicon can be continuously enriched and applied to other argumentation mining tasks, such as argument decomposition. The model can be directly applied in the development of argumentation theory based learning tools, such as Argumentation Web (Bex et al., 2013), and used to enrich other collaboration tools (Barcellini et al., 2005).

We acknowledge that there are some limitations to our work. First, our corpus is comparatively small and it prevents us from conducting extensive evaluation. As a challenge faced by all argumentation mining studies (Lippi & Torroni, 2015), a long term effort is required to create an annotated corpus for online discussion. Second, TAD can only do binary classification between argument and non-argument. Our pilot study shows that it can be extended to extract more detailed argument information. For example, by adding syntactic analysis, TAD has the potential to parse an argument messages further into claims and premises. Third, our example shows that argument information is more informative when it integrated with other
interaction data. Future studies are needed to explore innovative use of argument information with respect to sense making, collaboration, and conflict management.
Chapter 6

Essay 3 – Understanding Conflict in Crowd Collaboration: Conflict Measurements Design and Application

Chapter Summary

Crowd collaboration communities are fertile platforms for conflict. Crowd collaboration involves the integration of diverse opinions and contributions from self-managed participants. In the absence of traditional authority structures, conflicts are an important concern in these settings. Our understanding of how conflicts arise and evolve in these settings remains a puzzle. We examine these questions by utilizing and extending the artifacts designed in prior essays to analyzing the conflict phenomenon in Bugzilla, a crowd collaboration platform. This study follows a grounded theory approach that utilizes prior design science efforts. Based on an analysis of 23,094 conflict situations in Bugzilla, I show how conflict can be measured with extracted conflict information. I also describe patterns associated with the occurrence of conflict, its participants and their behaviors, and the temporal evolution of conflicts. I further interpret these findings with the help of prior work to develop theoretical propositions that explain conflicts in crowd collaboration settings that can provide directions for future research.

This chapter is adapted, updated, and rewritten based on a working paper which has been submitted to CSCW (G. Zhang, Purao, Zhou, & Xu, 2016). All uses of ‘we,’ ‘our,’ and ‘us’ in this chapter refer to the coauthors of the aforementioned papers.

Background and Motivation

Crowd collaboration has been recognized as a promising approach to leverage the “wisdom of crowds” because it allows for the marshaling the outputs of large numbers of contributors and to aggregate them into coherent artifacts or outcomes (Boudreau & Lakhani,
In contrast to other crowd sourcing models organized as contests (e.g. InnoCentive) or microtasks (e.g. Mechanical Turk), crowd collaboration relies on ongoing collaboration. Crowd collaboration is used in several domains such as open source software communities (Barcellini et al., 2005; J. Wang et al., 2015), the encyclopedia Wikipedia (Arazy, Nov, Patterson, & Yeo, 2011), as well as open platforms built by companies such as IBM, Dell, Apple, and Facebook for large-scale collaboration (Boudreau & Lakhani, 2013; Z. Wang & Hahn, 2015).

Crowd collaboration, however, remains difficult to manage due to the diversity among participants. The communities are often composed of participants of different backgrounds, expertise, and interests (Li & Jarvenpaa, 2015). This diversity is encouraged because it can contribute to innovation, but it is also a source of differences among participants that requires norms and practices for handling dissent and building consensus (Ko & Chilana, 2011). Members in these communities often tend to be self-selected and self-managing individuals. Without the benefit of administrative mechanisms such as legitimate authority and peer pressure to coordinate activity (Boudreau & Lakhani, 2013) coordination remains a challenge.

Conflict is, therefore, expected, and a significant concern in crowd collaboration. A moderate amount of conflict can be a positive influence because it encourages a diversity of perspectives, stimulates innovation and creativity, and improves decision making (Jehn, 1994; Simons & Peterson, 2000). In contrast, untamed conflicts can cause tension and dissonance, and may distract participants from performing their tasks, which in turn reduces team performance and individual satisfaction (Rahim, 2010). Conflicts can also break the weak ties among participants and impair the dynamic of a community (Dissanayake, Zhang, & Gu, 2014). Scholars have identified several consequences of conflict in crowd collaboration contexts, such as facilitating situated learning (J. Wang et al., 2015) and externalizing design rationales (Barcellini et al., 2005). The resolution of conflicts is, therefore, a critical step in the process of decision making in crowd collaboration (Ko & Chilana, 2011).
In spite of these acknowledgements, conflict remains understudied. It is often seen only as a smaller element in the collaboration process and rarely analyzed in detail (Gray, Shoaib, Kulkarni, & Suri, 2016; Lykourentzou, Antoniou, Naudet, & Dow, 2016). The few scholars who have examined conflict in crowd collaboration have done so with interviews and surveys (Crowston, Wei, Howison, & Wiggins, 2012; Filippova & Cho, 2015). Their efforts reveal the extent of the problem but only provide a limited window on how conflict occurs in crowd collaboration platforms (Filippova & Cho, 2015). The goal of this paper is to provide a first step towards developing descriptive and explanatory propositions by analyzing data in a crowd collaboration platform. Specifically, it aims to answering three questions: \textit{when and why conflict occurs in crowd collaboration settings, how people engage in conflict situations, and how conflict evolves in crowd collaboration projects}. Our effort is to provide both, analytical results based on data obtained from a crowd collaboration platform, and theoretical propositions derived from these analytical results, informed by prior work on conflicts and conflict management.

The analytical results we report are based on a study of 23,094 conflict situations. They demonstrate that conflict is pervasive in crowd collaboration platforms. We find that these conflicts tend to occur at regular intervals relative to project milestones. We find that the nature of conflicts is dictated by task characteristics, but is accompanied with elements that reflect affective elements. We find that the main participants in a conflict tend to be contenders, with a few collaborators and a significant number of third parties. We find that conflicts affect project performance in that projects take longer time to be completed, and many are closed without fully addressing the source of conflict. Drawing on prior work, we interpret these findings to develop a set of theoretical propositions that can serve as directions for future empirical work.

A secondary contribution of our work is a novel research prototype that includes computational models by extending text-mining algorithms to analyze conflicts in long conversations. To the best of our knowledge, this is the first analytical tool that identifies and represents conflicts based on semantic information in such data. These computational techniques
allow us to analyze the complete data set obtained from the platform that supports crowd collaboration in the Mozilla development community, and bridges the gap in prior work (Crowston et al., 2012).

We describe our work in the paper as follows. In section 2, we review previous conflict studies. Section 3 describes the research method, and the data collection efforts. In Section 4, we describe analytical results from the Mozilla development community, and interpret these to identify constructs and propositions that can be described as exploratory theorizing about conflicts in crowd collaboration. Section 5 provides some concluding remarks and points to future work.

**Related Work**

Conflict, as a substantial activity in collaboration, has been intensively studied in the fields of organization, psychology, and communication (De Dreu & Gelfand, 2008; Putnam & Poole, 1987; Thomas, 1992). In line with the process view of conflict (Thomas, 1992), relevant studies have identified multiple elements that are fundamental to understand the conflict phenomenon, including antecedences, participants, and the process (See Table 6-1). Additionally, a large number of studies have been conducted to evaluate the outcomes of conflicts. These studies show that conflicts don’t need to be negative. The actual impact of a conflict is conditioned by the conflict elements mentioned above (Jehn, 1995).

With the growth of digital collaboration, the same paradigm has been adopted to investigate conflicts in the virtual space. Lack of shared identity, lack of shared context, and information disparity have been consistently described as the fundamental factors that cause more conflicts and make the conflicts hard to cope with (Cramton, 2002; Hinds & Mortensen, 2005). In response to the challenges, several approaches and mechanisms have been proposed to manage conflict in digital collaboration. Examples includes developing swift trust between team members
(Mannix et al., 2002), creating transactive memory within the team (Hinds & Mortensen, 2005), and enforcing coordination mechanisms (Montoya-Weiss et al., 2001).

<table>
<thead>
<tr>
<th>Elements</th>
<th>Topics</th>
<th>Literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antecedences</td>
<td>Task-related vs Affective</td>
<td>Thomas (1992)</td>
</tr>
<tr>
<td></td>
<td>Subject matters</td>
<td>De Dreu &amp; Weingart (2003)</td>
</tr>
<tr>
<td></td>
<td>Social &amp; organizational context</td>
<td>Jehn &amp; Mannix (2001)</td>
</tr>
<tr>
<td></td>
<td>Timing</td>
<td>Gersick (1988)</td>
</tr>
<tr>
<td>Participants</td>
<td>Third party - Intervention strategies</td>
<td>Crowston et al. (2012)</td>
</tr>
<tr>
<td></td>
<td>Conflict party - Attitude, strategy, and action</td>
<td>Rahim (2010)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Olekalns et al. (2008)</td>
</tr>
<tr>
<td>Process</td>
<td>Sequential Model</td>
<td>Thomas (1992)</td>
</tr>
<tr>
<td></td>
<td>Punctuation Model</td>
<td>Pruitt (2008)</td>
</tr>
<tr>
<td></td>
<td>-Conflict transition</td>
<td>Kriesberg (2007)</td>
</tr>
<tr>
<td></td>
<td>-Conflict escalation</td>
<td></td>
</tr>
</tbody>
</table>

Conflict is increasingly recognized as a key managerial factors in recent studies that investigate collaboration in crowd collaboration. For example, a qualitative study by Wang et al. (2015) finds that members in a crowd collaboration communities are diverse in multiple dimensions, e.g. differences in background and differences in role (core vs. periphery members). The differences increase the chance of conflict and change the way that members interact with one another. Another study by Lykourentzou et al. (Lykourentzou et al., 2016) adds that personality differences among members can also cause conflicts in crowd teams which further influence the team performance. When investigating the decision-making process in multiple collaboration communities, Ko and Chilana (2011) identify a variety of means that conflict parties use to make arguments, such as speculation and tradeoff. However, they take these means as “non-evidence” for the task since their focus is on the correctness of technical designs. Focusing on conflict, Filippova and Cho (Filippova & Cho, 2015) take an initial step to re-examine the conflict phenomenon in crowd collaboration. Based on interview data, they
investigate the source and the type of conflict in the new context. Additionally, their analysis reveals transformative patterns among different types of conflict, e.g. from task conflict to affective conflict.

In spite of the maturity of conflict theories in prior work, understanding conflict in crowd collaboration remains fragmented. Questions about whether and how conflict in crowd collaboration occurs in different ways remain unanswered. To inform future research, we take advantage of communication data generated in crowd collaboration to provide a systematic view of conflict in the new context.

**Research approach**

Our research effort primarily involves grounded theory development (Corbin & Strauss, 2007). The data for the effort comes from the crowd collaboration platform used by Mozilla for coordinating development work on the Firefox browser. ([https://bugzilla.mozilla.org](https://bugzilla.mozilla.org)). We analyze this data using a set of computational models and text-mining algorithms aimed at analyzing long conversations, which we developed in chapter 5, following the design science research approach (Gregor & Hevner, 2013; Hevner et al., 2004).

Figure 6-1 outlines this research approach. The initial steps involved selecting a target platform and collecting communication data from the platform. The Bugzilla platform was selected because of its longevity, public access, and large number of participants. Data captured from this platform included project background, communication data, and data about the participants. The data was analyzed with the computational model and text-mining algorithm developed in chapter 5. The complete data set was analyzed, first by identifying messages which indicated arguments, and then to characterize conversations with such messages as conflicts. These message sets (long conversations) were analyzed to characterize how the conflict arose, escalated and was resolved, by examining the semantic content of messages. The complete data
set, annotated in this manner, was stored into a conflict repository, which was further analyzed with a view to answering the research questions, and characterizing the conflict situations. The final step of the research approach required returning to prior work and incorporating the literature (necessary for grounded theory development) to arrive at the theoretical propositions.

Figure 6-1. Research Approach

Site Selection

The site we selected for our analysis was Mozilla, a large and well-recognized open source software community. We focused our analysis on its core project, Firefox, a widely adopted open source browser. Because this is an end-user oriented product, the project involves participants with significant diversity in background, expertise, experience, and interests. The Mozilla development community is composed of two types of members: core developers and
volunteers. Core developers, often employees of Mozilla Cooperation, are those who significantly contribute to the product and take charge of product evolvement for their specific areas. Some enthusiastic contributors, not affiliated with the Mozilla organization, are also given the privilege to conduct super-reviews and make final decisions on the products. Volunteers are composed of a wide range of web-developers, third-party developers, and end-users who occasionally participate in the project based on their own interests. The collaboration platform for this project is Bugzilla, used for reporting bugs and discussing solutions. Each bug is associated with a discussion thread where members append message to discuss the bug. A bug thread is also described as a project where participants investigate the reported bug, make decisions, and fix the problem. Most bug threads are open to all members. Messages sent to a bug thread are archived chronologically in Bugzilla, which is open for review and access.

Data Capture

Bugzilla provides a RESTful API for open access. We developed a tool to retrieve bug reports and discussions. The data collection effort started on March 17, 2016, and lasted for one week. Retrieved data included the background information for all bug reports filed for Firefox since February 9, 2004, when the first version of Firefox was released. For each bug report, messages associated with the bug were also retrieved, including publicly available information about the message sender. This included a total of 1,172,717 messages sent by 86,601 members in 156,765 bug threads over the span of about 12 years. The retrieved data was examined and checked to ensure that it captured all the data available from the Bugzilla platform.
Argument Detection

The collected messages were firstly analyzed using the argument detection model – TAD – as described in Essay 2. This trained argument detection model was run to annotate messages in the data set. The messages in the training corpus were annotated using the same procedure as the 10-fold cross validation. The annotated messages, along with bug and participant information, were then stored in a Conflict Repository.

Conflict Repository

The Conflict Repository, thus, included information about bugs such as creation time, reported by, status, and the software component associated with the bug. Information about messages included creation time, sender, sequence in the bug thread, text body, and argument annotation. The information about participation included participation instances, each of which records a participant and his/her messages in a bug thread.

Since its first release and through a number of versions, 69,464 unique participants have reported 156,765 bugs about Firefox. The number of bugs filed per reporter ranged from 1 to 989, with a majority of them (81.6%) reporting only one bug. An additional 17,147 participants have participated in discussions for fixing the bugs, bringing the number to 86,601 participants, contributing 1,172,717 messages. The number of messages sent per human participant ranged from 1 to 25,482, with a very large number of participants (87.5%) contributing no more than 5 messages. This is consistent with the organizational structure of Mozilla community, where an active core development team is accompanied by a large number of volunteers (J. Wang et al., 2015). When the data was captured, of the 156,765 bugs, 21,766 bugs were still open, while 139,614 were closed.
Conflict Identification

To identify conflicts worthy of further consideration, we identified threads where a number of messages were annotated as arguments. This involved the following steps. First, we discarded threads which had less than 2 participants whose messages were annotated as arguments. Next, we discarded threads which had a preponderance (75%) of messages automatically generated by a robot account. We further removed short threads, using a cut-off of less than 10 messages. Finally, we removed bug threads in which messages annotated as arguments accounted for less than 30% of total messages. The number was determined based on a balance between reducing false positives and ensuring that we examined threads where conflicts occurred for a significant part of the conversation. The resulting set of 23,094 threads represented 14.7% of all bugs. Among the selected threads, 20,318 (88.0%) were closed. The length of these threads ranged from 10 to 538 messages (median 16). The number of participants ranged from 2 to 222 (median of 6). Of the 503,712 messages recorded in these threads, 282,332 (56.1%) were annotated as arguments. As a fraction of total messages, the argumentative messages within a thread ranged from 30% to 100% (median 54.5%), and the number of arguers as a fraction of the total number of participants ranged from 14.3% to 100% (median 71.4%).

Analyses and Interpretations

The set of 23,094 threads represented 23,094 conflict situations. These were investigated to understand three phenomena: the occurrence of conflict, the actions of participants, and the evolution of conflict.
Occurrence of Conflict

When Conflicts Arise

To examine a longitudinal view of when conflicts arise on Bugzilla, we plotted all messages based on the months when they were generated (see Figure 6-2). The diagram can be read as two parts, before April 2011 (the slow release mode), and after April 2011 (the fast release mode) when the Mozilla Community increased the release pace of Firefox. During the slow release mode, we observe that the number of messages, including ones annotated as arguments, surged between release points. Because release dates of Firefox are planned in advance, the pattern indicates that the community experienced concentrated discussions and debates at or after the middle points between these project milestones, and decreased as the next milestone approached. After switching to the fast release mode, the impetus for greater interaction between release points seemed to disappear.

The observation is supported by a boxplot representation (see Figure 6-3), which showed that the community often reached the highest conflict level when it 50% or 80% approached the new release during the slow release mode, while it reached the peak almost at any stage during the fast release mode. In the figure, the vertical axis represents the progress from one Firefox release to the next.

Figure 6-2. Number of Messages Generated every Month
We also noted that the number of messages generated per month significantly increased after changing to the fast release model ($t = -6.60, \text{df} = 143.4, p < 0.001$). One plausible explanation is that fast (and short) release cycles can trigger more bugs (and therefore, more messages) compared to the slow release mode. Other explanations can be inferred from the rise of Google Chrome as a competitor, and complaints about the fast release model by both developers and users (Reed, 2012).

![Plot of Peak Conflict Levels](image)

**Figure 6-3. Plot of Peak Conflict Levels**

**Causes of Conflicts**

We investigated the reasons that caused conflicts on Bugzilla from two perspectives: subject matter and nature of conflict. To do this, we examined the association of reported bugs against 67 Firefox components. The specified component indicates the subject matter of a bug report as well as any related conflicts. We grouped the components into 4 categories based on their basic function and potential participants. Core Functions refer to components that are essential in the operation of Firefox and are mainly manipulated by relevant developers. Developer Tools refer to a series of components that support web-developers to run and debug code through Firefox. End-user functions refer to those that can be controlled by end users and
therefore influence user experience. The General category include two original types of bugs reports (General and Others) that are not directly associated with a particular Firefox component.

The other perspective is the nature of conflict. Every bug thread on Bugzilla is organized to fix a reported bug. When conflicts arise in a thread, they are initially task-related. However, as tension accumulates among participants, the nature of messages changes to include strong, offensive language. In order to identify such affective conflicts, we implemented an offensive language detection tool. The tool implements the LSF-based Offensive Language Detection Framework (Chen et al., 2012). It can detect offensive messages based on the appearance of lexical and syntactic features. Both strong offensive language (e.g. **Guys, can't you all get a f**kin' grip) and weak offensive language (e.g. [The design] would have to be the stupidest things done in mozdev history) are captured. A conflict was determined to be affective conflict when offensive messages were identified in its discussion.

![Figure 6-4. Causes of Conflict](image)

Figure 6-4 shows the distribution of conflicts with respect to subject matter and affective conflict. It shows that threads associated with end-user components accounted for a large portion of identified conflicting threads. Components related to end-user functions (e.g. Bookmarks, History, Search, and Theme) accounted 40.7% of conflicting bug threads. Affective conflict appeared in the threads of all categories. Overall, 18,138 or 78.5% of threads included affective conflict.
Participants in Conflict

Third Party

We defined a participant in a conflicting bug thread as a third party if he or she only sent non-argumentative message(s). Figure 6-5 shows the distribution of bug threads with conflicts examining the proportion of third parties. It shows that 15.9% bug threads progressed with no third party involvement although most bug threads (84.1%) had third parties involved, while most had 20% to 40% participants as third parties. On average, 27.8% of participants in a conflict situation on Bugzilla played the role of third party.

Collaborators and Contenders

Unlike Third Parties, Conflict parties are those who send messages that contain arguments. While there are five types of conflict management styles (Olekalns et al., 2008; Rahim, 2010), we classified conflict parties based on the two styles observable in online communication: collaborating and contending. By the definitions of two styles, collaborators refer to the parties who spend a great deal of time engaging in the conversation and who develop solutions that will appeal to all parties. Contenders are those who attempt to prevail or win their
position through forcing behaviors. We used two measurements classify them. The first measurement measures non-forcing behaviors, measured by whether a conflict party sends non-argumentative messages in a bug thread. The second is the engagement level of a party in a bug thread, computed as the following:

\[ E_{p,b} = \frac{(M_{last} - M_{first}) + TotalMessages_p}{TotalMessages_b * 2} \]

where \( TotalMessages_{p/b} \) is the total number of messages sent by the participant or sent in the whole bug thread. \( M_{first/last} \) is the sequential number of the first or the last message sent by the participant. The equation is designed with the assumption that the longer a participant stays in a bug thread and the more messages they send, the more effort the participant spent engaging in the bug discussion. The possible value of \( E_{p,b} \) is within the area of \((0, 1)\).

We selected 0.5 as the cutoff point for engagement level, and classified conflict parties with 0.5 or higher engagement level (contributing both argumentative and non-argumentative messages in a bug thread) as collaborators, and the rest as contenders. Figure 6 provides a summary of participants in the conflict repository. Among 22,609 unique participants, 24.7% of them once played the role of third party and 88.2% of them once engaged in conflicts in the bug threads. Contender is the most frequently employed role. 77.4% of overall participants once played the role of contender.

Contenders also dominated in most conflict situations (see Figure 6-6). In 34.2% of conflicting bug threads, contenders account for about half of the overall participants. And in 37.4% of them, contenders account for more than 60% of all participants.

Figure 6-7 shows the distribution of bug threads with respect to the number of collaborators in a bug. The results show that the maximum number of collaborators in a thread was 4, regardless the total amount of participants in the bug. However, in spite of the small absolute number, collaborators were found in most (16,642, 72%) conflicting bug threads. This
implies that in most conflict situations, there was at least one participant stayed in the discussion for a comparatively long time during the conflict process.

Figure 6-6. Contender in Bug Threads

Figure 6-7. Collaborators in Bug Threads

Figure 6-8. Types of Participants in a Thread
Figure 6-8 shows the three types of participants in bug threads. It consistently shows that contenders often compose half of all participants in a bug thread, followed by a significant proportion of third parties and a limited number of collaborators.

**Process of Conflict**

**Duration of Conflict**

We found from our results that threads on Bugzilla that contained conflicts took longer to resolve than threads that did not contain conflicts (see Figure 6-9 and 6-10). The difference is statistically significant for both closed and open bugs ($t = 27.9$, $df = 24870$, p-value < 0.01; $t = 31.4$, $df = 3005.9$, p-value < 0.01). With respect to conflicting bug threads, while some (10.8%) were addressed within few days, most of them could not reach a resolution without a prolonged discussion. For example, our analysis results show that 30.0% of conflict situations took more than 1 year to reach a resolution. The percentage of prolonged bug threads is surprisingly high in open bugs. It shows that among 2,776 open, conflicting bugs, half of them had been discussed for more than one year without resolutions. By contrast, Figure 10 shows that a large portion of closed routine bugs (44.7%) were addressed within one week, while a small portion (15.3%) bugs remained open after a year.
Conflict Escalation

To capture changes in conflict level in a bug thread, we designed two measurements. According to the conflict escalation theory (Pruitt, 2008), a conflict situation is intensified as more contentious actions are taken and more participants join the conflict. For each message in a bug thread, we define its contribution to the overall conflict level as:

\[
C_m = \begin{cases} 
1 & \text{if } m \text{ is arg. & sender is new} \\
0.5 & \text{if } m \text{ is arg. & sender is not new} \\
-1 & \text{if } m \text{ is not arg. & sender is not new} \\
-0.5 & \text{if } m \text{ is not arg. & sender is new}
\end{cases}
\]
To calculate the conflict level at each stage marked by message number, we used the following function:

\[ I_m = \max(0, I_{m-1} + C_m) \]

We examined how conflicts escalated by examining the maximum, average, and ending conflict level of each bug thread using the two measurements. Comparing the measurements reveals the evolution of a conflict situation, e.g., the time of escalation (Kriesberg, 2007; Pruitt, 2008). Figure 6-11 shows when bug threads reached their highest conflict level. It shows that the chance that a bug thread reached the highest conflict level gradually increased as the bug thread approached closure. About half (48.8%) of observed bug threads escalated to the highest conflict level at the second and third quarter of the conversation. A comparatively smaller portion of bug threads reached the highest conflict level at the beginning, and a larger portion escalated when the bugs approached closure.

![Figure 6-11. Escalation Levels across Thread Duration](image)

Figure 6-12 shows the distribution of bug threads with respect to ending conflict level. It shows that bugs tended to end with a comparatively higher conflict level. According to the analysis, more than half of bug threads ended with a conflict level higher than the average level across the whole bug process. 18% reached the highest intensity level near the end of the
conversation. Only 27% of bugs ended with a conflict level lower than the average level across the duration of the thread.

Figure 6-12. Ending Conflict Levels in Threads

**Interpretation and Theoretical Propositions**

The findings from our data analyses provide an overview of the conflict phenomenon in crowd collaboration settings. In this section, we interpret these findings by incorporating prior work in conflicts and conflict management. Table 6-2 presents a summary of the findings (Research Propositions) and prior work.

**Occurrence of Conflicts**

Our analysis results show that from a longitude perspective, there are two situations when a crowd collaboration community faces the most conflict. One is at the middle points between milestones. According to early studies (Gersick, 1988), the midpoint refers to a crucial time for teams to adopt new perspectives while achieve consensus. High-performing teams are observed to have concreted debate and discussion of their tasks with the increase of conflict level at the
middle point (Jehn & Mannix, 2001). In the case of Firefox, this pattern is observed during the time when Firefox constantly increased its market share from 2002 to 2011 (based on (“Browser Statistics and Trends,” n.d.)). From this perspective, we contend that (RP1a) a crowd collaboration community will experience increase in conflict at the middle point between milestones. A following question then emerges (RP1b) whether this pattern is caused by a shared awareness about project progress (Hinds & Mortensen, 2005), or enforced by the core members (Yoo & Alavi, 2004).

Another situation is dramatic internal and external changes, such as a change in Firefox release mode and the rise of its competitor Chrome. Dramatic changes can break the stability of a community which is built upon shared mental models about the task and its goals, as well as the methods of coordination and collaboration (Majchrzak et al., 2005). We propose that (RP2) an increase in conflict is a natural means to re-establish stability in a community, allowing peripheral members to understand changes and rebuild consensus.

Our analysis showed that task conflicts are often accompanied with affective conflict in crowd collaboration. Tension between the participants was observed when offensive language was used in arguments. Although “flamed” messages only accounted for a small portion of the overall messages in a conflict, they were widely observed in the debates. This is consistent with previous findings that people tend to hold negative attitudes toward opponents and to adopt a harsh communication style in the virtual context (Hinds & Mortensen, 2005). In addition, Filippova and Cho (Filippova & Cho, 2015) supplement that affective conflict can be “transformed” from task conflict due to frustrations about lack of progress and power imbalance in a project. Therefore, we propose that (RP3) affective clash is more likely to emerge during the task conflict in long duration threads.
Table 6-2. Research Propositions

RP1a: a crowd collaboration community will experience increase in conflict at the middle point between milestones.

RP1b: this pattern may be caused by a shared awareness about project progress, or enforced by the core members.

Findings: Fig. 2 (2002-2011) & Fig. 3


RP2: an increase in conflict is a natural means to re-establish stability in a community, allowing peripheral members to understand changes and rebuild consensus.

Findings: Fig. 2 (2011-2016)

Literature: Majchrzak et al. (2005)

RP3: affective clash is more likely to emerge during task conflict in long duration threads.

Findings: Fig. 4


RP4: new approaches are needed to interpret and rebuild arguments made by a crowd of contenders.

Findings: Fig. 6

Literature: Wang et al. (2015)

RP5: the presence of third parties is beneficial to the resolution of conflict in crowd collaboration.

Findings: Fig. 5

Literature: Crowston et al. (2012)

RP6: collaborators will play a pivotal role in influencing the progress and outcome of conflict in crowd collaboration.

Findings: Fig. 7


RP7: the distribution of three types of participants in a conflict influences the outcome.

Findings: Fig. 5, 6, 7, 8


RP8: investigate novel structures and mechanisms for efficient conflict resolution.

Findings: Fig. 9 & 10

Literature: Crowston & Howison (2006)

RP9: Proactive actions are needed to encourage or elicit conflict at the early stages of crowd collaboration projects.

Findings: Fig. 11 & 12

Literature: Hinds & Mortensen (2005)

Participants

We classified participants in a conflict situation in three groups: third parties, contenders, and collaborators. Among them, contenders often compose the main boy of the participants. We see that the presence of contenders is critical for crowd collaboration. Their participation brings
different perspectives about the task in hands and contributes alternative solutions (J. Wang et al., 2015). However, because each one often only contributes a few messages and exhibits lower engagement in the conversation, he or she contributes only a fragment of a complete argument. It poses new challenges to evaluate their claims/evidences and aggregate them into coherent argumentation. Thus, future research needs to address the question (RP4) how to interpret and rebuild arguments based on pieces of statements made by a crowd.

Third parties are found to widely exist in conflict situations. Taking no specific standpoints, third parties join the conversation to share comments/observations on the ongoing work, provide updates about the bug thread, and answer questions posed by other participants. Their participation can be explained by both intrinsic motivations, such as fun and sharing opportunities, and extrinsic motivations, such as establishing reputation in the community (Crowston et al., 2012). Serving the roles of information provider and conversation facilitator, we hypothesize that (RP5) the presence of third parties is beneficial to the resolution of conflict.

While not uncommon, the absolute number of collaborators in a conflict situation is found to very small. This finding is consistent with an early study by Crowston and Howison (Crowston & Howison, 2006) which finds that the core group size in a crowd collaboration project tends to be around three. Based on our preliminary analysis, collaborators were often “Need-Driven Participants,” who had a large stake in the ongoing project and thus were inclined to control the project process as well as results (Shah, 2006). By arguing with others, collaborators are able to achieve better solutions as well as to externalize rationales behind the solutions (Barcellini et al., 2005). However, when collaborators are also core members, their insistence might be taken as stubborn by peripheral members and impair their long-term relationships (J. Wang et al., 2015). We, therefore, propose that future research needs to address (RP6) how collaborators will play a pivotal role in influencing the progress and outcome of conflicts.
Finally, early studies show that the way of approaching a conflict situation has significant influence on the outcome of the conflict (De Dreu & Weingart, 2003). It would be important for future research to explore (RP7) how different participant types and their composition in a group influence the conflict outcomes.

Conflict Process

The analysis results indicate that conflict situations in crowd collaboration tend to take more time, and prolong the project lifecycle. It may be the case that projects that have conflict involved are hard to cope with by their nature. However, we argue that it is necessary to facilitate the conflict resolution process, especially when conflict reaches stalemate. In crowd collaboration, the final output is built upon a fair amount of outputs provided by inter-related ad-hoc projects. Unresolved conflict can cause delay of the overall project and impair the quality of the final output. Since there is no formal hierarchical structure in the online community (Crowston & Howison, 2006), it would be helpful for future research to (RP8) investigate novel structures and mechanisms for efficient conflict resolution.

In spite of the finding that conflict in a project can escalate at any stage, we also find that a large portion of bug threads end with a comparatively higher conflict level. This indicates that a fair amount of conflicts in crowd collaboration are closed without resolution. Such unresolved conflicts can impair the quality of crowd collaboration outcomes, lead to the accumulation of dissent among participants, and cause damage on the dynamics of the community. Drawing upon the early discussion about the temporal pattern of conflict (Hinds & Mortensen, 2005), we propose that (RP9) it would be helpful to take proactive actions to encourage or elicit conflicts at the early stages of the project.
Conclusion

In this paper, we explored the conflict phenomenon in the context of crowd collaboration. Following a grounded theory development approach (Corbin & Strauss, 2007), we allowed our work to be informed by data obtained from the crowd collaboration platform Bugzilla. Our analyses were informed by design science efforts (Gregor & Hevner, 2013; Hevner et al., 2004), which included the development of a set of design artifacts to analyze conflict situations in Bugzilla. We revealed patterns associated with the occurrence of conflict in crowd collaboration, participants and their behaviors, and temporal evolution. Drawing upon the findings, and incorporating prior work, we develop several theoretical propositions that can guide future research.

Our study makes several contributions to research and practice. To the best of our knowledge this is the first study that examined communication threads in a collaboration community to investigate conflict. In doing so, it supplements prior studies, which have mainly relied on interviews and case studies (Crowston et al., 2012). The computational artifacts developed in this study can also be used for future research and developing tools to examine conflicts in crowd collaboration.

We acknowledge that there are several limitations to our work. The tools we have constructed have some limitations. For example, the argument detector can only do binary classification. Further effort is required to create corpuses that have fine-grained conflict information annotated and to employ them to update the detector (Lippi & Torroni, 2015). Such information would allow us to identify allies in conflict and get closer view of conflict parties’ behaviors. Second, every crowd collaboration community is unique with respect to task and participants. While big, our data set contains data collected from only one source. Future effort is needed to further refine the findings. In spite of these limitations, we hope that our work
demonstrates a path for analyzing big data to describe patterns, which may then be useful for exploratory theorizing by incorporating the literature.
Chapter 7
Conclusion

Conflict is a substantial, pervasive active in team collaboration. With the growth of online collaboration, it has become a prominent concern in distributed teams, in which participants are geographically dispersed and heavily rely on computer-mediated communication. While the new setting causes extra burden to detect and understand conflicts, it also makes possible to rebuild conflict scenarios by analyzing archived communication data. The motivation of this dissertation is to advance the design of IT artifacts to enable automatic conflict detection and analysis with computer-mediated communication data. In this dissertation, I present three sets of design artifacts which together serve the purpose.

First, I introduce the meta design of a novel conflict management approach along with an instantiation called CM2. The foundation for the design is the notion of the conflict Vignette – vivid yet analytical, theory-laden descriptions of conflict episodes. The model presents a new way to capture and present key information pertaining to a conflict, such as cause, claims, transitions, actions, strategies, and other relevant information.

The meta design has the potential to make several contributions to theory and practice. The design provides a framework for capturing and representing conflict scenarios. It also points toward a design for automatic conflict extraction and analysis tools, as developed in Essay 2 and Essay 3. In addition, the notion of the Vignette is, in itself, an important contribution because it suggests a new structure that provides a new foundation for case-based reasoning systems. Further, it also provides an approach that can be leveraged for capturing and representing theory-laden episodes in other domains beyond conflict management. I also claim that practitioners may benefit from CM2 in that it can improve their conflict management strategies, and may eventually develop more constructive attitudes toward conflict.
Second, I describe the design of a theory-based argument detection model (TAD), which can effectively detect argumentative messages in CMC. Conflicts in a distributed team develop and manifest through argumentative messages exchanged among the members. Detecting these messages is the first step toward automatic conflict extraction. In line with the argumentation theory, TAD meets the goal based on (a) features that reflect five categories of argumentation functions and that are informative for recognizing argumentative language, and (b) leveraging language features draws from classical text mining techniques. The empirical evaluation results show that the use of TAD models can achieve higher Accuracy and Recall rates in detecting argumentative messages, compared to baseline models.

To the best of our knowledge, TAD is the first model built upon features of argumentation theory, combined with classical text mining functions. It also includes the first fine-grained lexicon of argumentative language in online discussion. The lexicon can be continuously enriched and applied for other argumentation mining tasks, such as argument decomposition. The model can be directly applied in the development of argumentation theory-based learning tools, such as Argumentation Web (Bex et al., 2013), and used to enrich other collaboration tools (Barcellini et al., 2005).

Third, I complete the puzzle with a set of instruments which can be used to measure conflict elements as specified in the meta-model and based on automatically detected argument information. Following a grounded theory approach, I employ the instruments to analyze 23,094 conflict situations in Bugzilla, an open source software development community. The analysis results reveal patterns associated with the occurrence of conflicts, participants and their behaviors, and the temporal evolution of conflicts. We interpret these findings with the help of prior work to develop theoretical propositions that explain conflicts in crowd collaboration settings that can provide directions for future research.

Investigating the conflict phenomenon in crowd collaboration communities presents another way to apply the proposed design artifacts. It also supplements prior studies, which have
mainly relied on interviews and case studies to understand conflicts in distributed teams (Crowston et al., 2012).

In conclusion, I design, implement, and apply a novel conflict detection and analysis approach, composed of a meta-model, an automatic argument detection solution, and a set of instruments for capturing and measuring the conflict phenomenon. With the growth of distributed teams, I believe these designs are beneficial in the design of conflict management systems with automatic analytical tools and leveraging CMC data to advance our understandings about conflict in the virtual workspace.
Bibliography


sciences of design: observations on an emerging field. *Communications of the Association for Information Systems, 23.*


VITA

Guangxuan Zhang

Education

PhD Candidate in Information Science and Technology, August 2010 ~ Present
Pennsylvania State University

M.S. in Applied Economics, May 2011
University of Vermont

M.S. in Management, June 2008
Yangzhou University

B.E. in Computer Science, June 2005
Shandong University

Research Interests

My research interest is in data mining and computer-supported collaboration. My current research focuses on designing effective IT solutions to detect and visualize conflicts in virtual teams.

Publications


