Asynchronous Negotiation and Collaboration of Software Requirements for an Emergency Response Information System: An Empirical Investigation

Catherine Lowry Campbell
New Jersey Institute of Technology, USA
campbell@njit.edu

Bartel A. Van de Walle
Tilburg University, the Netherlands
bartel@uvt.nl

Fadi P. Deek
New Jersey Institute of Technology, USA
fadi.deek@njit.edu

ABSTRACT

Negotiation and collaboration during the requirements stage of the software engineering process are fundamental to developing successful software products. Groups of stakeholders work together to come to agreement on the most important requirements to be sent forward for implementation. Distributed software engineering is becoming the norm rather than the exception, yet the requirements elicitation and definition process is normally conducted face to face. This paper describes an empirical study to investigate the relationship between structured task and specified negotiation steps within an asynchronous environment. The results reveal that these structures can have a positive impact on solution quality but a negative impact on process satisfaction, although following a negotiation sequence and task structure can help asynchronous groups come to agreement faster. Details of the experimental procedures, statistical analysis, and discussion of the results of the experiment are presented, as are suggestions for improving this work and a plan for future research.

Keywords

Software requirements, negotiation support, emergency response information systems, asynchronous communication, requirements engineering, collaborative software engineering

INTRODUCTION AND BACKGROUND

The integration of emergency response activities and information management systems has been progressing, but this process has been accelerated in recent years. Much research has been done to define the needs of emergency responders. Current information systems can provide a level of support unprecedented in this field, making coordination between local organizations, state government, and federal authorities routine and seamless. Such collaboration and integration could improve emergency personnel response and bring expertise to a situation in a rapid manner, saving valuable time, equipment, money, and personnel, and preventing the development of larger and more dangerous situations.

Turoff (2001) summarizes the critical concepts that must be addressed when considering the development of an emergency response system. These include assumptions such as the following: (1) everything in a crisis situation can be an exception to the norm, (2) emergency systems are not used on regular basis, (3) people in emergencies work long days and have no tolerance or time for dealing with unrelated matters, and (4) the critical problem of the moment is the primary factor which collects the people, the authority, and the resources that are needed to be brought into play. Also, process improvement depends on knowledge of what actually happened before, during, and after a crisis. Furthermore, establishing and supporting confidence in a decision through supplying the best possible, up to date information is critical to those whose actions may risk lives and resources. Crisis situations involve the necessity for many hundreds of individuals from different organizations to be able to exchange information freely, delegate authority, and conduct oversight, without the side effect of information overload. All of these factors and issues are critical in the planning of emergency response information systems. It is also critical to develop an information system that can be used on a daily basis during non-emergency activities, so that the personnel are familiar with the system and can transition seamlessly to emergency mode (Turoff, Chumer, Van de Walle, Yao, 2004).

Many small communities do not have the resources, personnel, or the expertise to develop a set of requirements to assist them in managing their day-to-day emergency response activities. The development and description of a generic set of requirements that could be utilized by state and local jurisdictions would clearly enhance the effectiveness and possibly reduce cost of any emergency management information system. It would also reduce the incompatibilities between local systems so they could be networked at the state level more easily. In the event of an emergency, it is the local responders that provide the first efforts in containing the event. These critical first minutes could mean life or death in these situations. Having a set of common system requirements will be very valuable in assisting local communities who might not have the resources to do intensive work required for developing a useful emergency response information system for their community.

It is in the best interest of organizations to have relevant stakeholders participate in the requirements definition process and to contribute to the development of any new system. Often this is hampered by the geographic distribution of the organizational entities. This research described here investigates the asynchronous negotiation of software requirements within the emergency response domain as stakeholders from different emergency response organizations collaborate and negotiate a set of software requirements to come to agreement.

NEGOTIATION IN REQUIREMENTS ENGINEERING

Stakeholders and software engineers need to work together to create requirements for a new software system. The software engineer or systems analyst elicits requirements from the stakeholders. After elicitation, there is a period of negotiation where software engineers work with the stakeholder to resolve conflicts. It is important at this stage to ensure that no critical requirements have been overlooked and that the final list contains those requirements that will define a useable functional software information system that meets the client's need (Karlsson and Ryan, 1996). Therefore, negotiating conflicting requirements is an important part of the requirements phase of the software engineering life cycle (Grünbacher, Hofer, 2002, Egyed and Boehm, 1997). Studies have consistently found that clear, consistent, and traceable requirements result in more robust, maintainable software systems (Bray, 2002, Brooks, 1975). In order to provide this quality, consistency, and traceability, conflicts between users, analysts, and managers must be resolved during this important phase of software design and development (In and Roy, 2001).

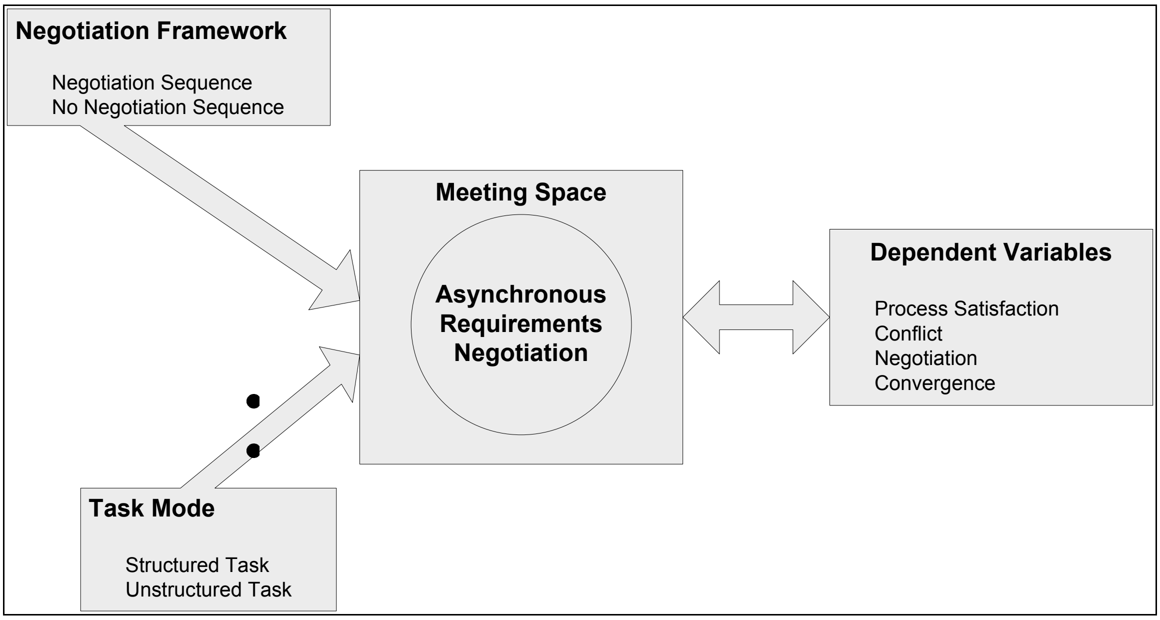
Conflict in and of itself is not a negative activity at this stage of software development. Actually, conflict and its resolution at an early stage in the development process can have the result of saving both time and money further along in the software life cycle (Hall, Beecham, and Rainer., 2002, Damian, 1999, Boehm, 1994). During requirements negotiation, conflicts should be resolved effectively and measures taken to ensure that no critical requirements are lost due to poor negotiation skills on the part of the user or analyst (Blackburn, Scudder, and Van Wassenhove, 2000, Conboy, 2002). Keeping a record of the requirements negotiations provides a method of traceability and allows for cross checking for consistency (Bubenko, 1995). One possibility to do this is to conduct negotiation using an asynchronous bulletin board. This would automatically log each stakeholder’s comments and trace the progress from the initial requirements discussions through completion. Providing a method of capturing the requirements negotiation process eases the pressure in later development stages. When new requirements or modifications to existing requirements are suggested or found to be necessary, it will be possible to return to the records to determine what will be affected by the proposed changes.

RESEARCH QUESTIONS

Virtual collaborative teams are becoming increasingly common in software engineering. These teams have the potential to transform the software development process and increase the quality of software products by providing a stable and traceable development environment (Deek and McHugh, 2003). The ability to comprise a virtual team from the best skilled members is a strong impetus to developing a strong distributed team process. Often workers that have the best suited skills for a project are not co-located, so the only option is to have team members work collaboratively in a distributed mode. Optimally, team members would have opportunities to meet, but since this is not always possible, successful teams must have a robust group process.

The research discussed here investigated the effect of task structure and negotiation sequence on asynchronous groups. The following questions were investigated: 1) How does providing different task or negotiation frameworks affect distributed group discussions and outcomes? 2) Does group task structure influence conflict in asynchronous group communications? 3) Are groups more satisfied with their group process and solution when a structured task and negotiation steps are provided? 4) Is there an interaction effect between task structure and negotiation sequence? 5) Does providing a negotiation sequence or task structure influence the process which asynchronous groups converge to a consensus?

CONCEPTUAL MODEL



EXPERIMENTAL CONDITIONS

In this study, distributed teams were provided with a framework that incorporates established negotiation procedures and that has been shown to be successful in face-to-face negotiations. It is reasonable to expect that if these procedures are adapted for distributed teams, they can be expected to improve the software development process and provide mutual satisfaction for team members, enabling them to move forward in a positive manner. It is also reasonable to believe that adding a task structure to the negotiation activity will enhance the group's results. The goal was to investigate these issues, and therefore to ask whether negotiation theory and distributed group discussions can improve the quality and satisfaction of decision making.

The independent variables are Structured Task and Negotiation Sequence. Both variables can be present or absent. The experimental conditions are shown in the table below.

	With Structured Task	Without Structured Task
With Negotiation Sequence	Structured Task with Negotiation Sequence	Without Structured Task with Negotiation Sequence
Without Negotiation Sequence	With Structured Task without Negotiation Sequence	Without Structured Task without Negotiation Sequence

●

●

TASK DESCRIPTION

The task was a hypothetical situation created for the purpose of this study. Students, used as subjects, were instructed to negotiate requirements for an emergency response system for a fictitious county. The task was created to encourage discussion and negotiation. It is consistent with McGrath's task Circumplex (1984) conceptualization (Figure 2.1). It can be classified according to McGrath's Circumplex as a combination mixed-motive and competitive type of task.

The task focused on the ranking of the specified requirements and the stakeholders' need for each requirement. Each stakeholder's organization has different requirements for the proposed system. The stakeholders do not know the critical needs of the other organizations, as the requirements were elicited individually from each organization by the Project manager's and his/her system analysts. Because it is assumed that the organizations have been provided with state and local government funds to create this system, it is in their best interest for their representatives to work collaboratively and negotiate to find a middle ground that satisfies all stakeholders. Since the funding was insufficient to implement all requirements, the stakeholders had to negotiate and agree on a subset of suitable requirements that will go forward for implementation. Below is a master list of all the requirements needed by the multiple stakeholders. This master list was given to the subjects as a guide. In addition, the subjects were supplied with individual organizational requirement lists categorized as critical, high, or of medium importance. The lists contained an estimate of the man-months needed to implement each requirement. The experimental task required the team of stakeholders, along with a project manager, to choose a list of requirements that came within a restricted man-month allocation – not all requirements could be implemented in the initial system release.

Master List of Requirements

Requirement Name	Requirements Description
Command & Control	The system shall provide a physical command and control system in a centralized location that shall be integrated yet provide individual functionality to each response organization.
Remote Access	The system shall provide remote access to emergency operations center databases and communications by personal computer or Personal Digital Assistant (PDA).
Public Website	The system shall provide a current informational website for public access and separate pages for individual official organizations.
Linked Time Log	The system shall maintain a time log of all events with information linked on the nature of the event, the reporter, status of the event, and allow for links between related events.
Dispatch Service	The system shall provide a dispatch service which will include the origination and tracking of tasks. This system will be used by police, fire services, medical organizations, etc.
Interagency Access	The system shall provide distributed interagency access, allowing authorized users remote access to data collected by any of the agencies during the event.
GIS Database	This system shall provide access to a Graphical Information Database for digital mapping of critical infrastructures, i.e., sewer, utility, roads, river hydrology, etc.
Electronic Bulletin Board	The system shall provide an ongoing asynchronous informational message board which will provide threaded discussions for authorized users.
Resource Database	The system shall maintain a resource database of contact information for local and military personnel. It will also provide equipment and personnel/volunteer databases for each critical infrastructure in the county.
Electronic Command Post	The system shall provide the functionality in R1 as a virtual command and control center independent of any physical location. This functionality shall be accessible to any authorized user within the communication network. *Note: if R1 AND R10 are chosen, then R10 will take only 5 additional man months in addition to the 10 in R1, otherwise R10 consumes 12 man months.
Victim Database	The system shall provide an interactive input form and accompanying searchable database of persons involved in the incident (personal info and photo ID to be taken onsite).
Laboratory Database	The system shall provide a database of laboratory measurements and tests taken from the event.
Whiteboard	The system shall provide a whiteboard collaborative working area where individuals can create and view drawings. The users shall also be able to overlay markups and update/change existing drawings.
Mobile Resource Tracking	The system shall provide a means of tracking mobile physical resources, i.e., ambulances, patrol vehicles, road crews, etc.
Dispatch/Response Database	The dispatch event logs generated by different response organizations shall be designed and stored as one dynamically updated database, yet provide individual functionality to each response organization.

STAKEHOLDER ROLES

Each group had six role-playing members. Each member had a specific description of his/her job and of the requirements that the organization wanted to see implemented in the new information system. The stakeholders are representatives of these organizations: fire containment, law enforcement, public health, public works, state/civil liaison, and a program manager from the development company who also functioned as the group facilitator.

ANALYSIS

Over two semesters, Fall 2003 and Spring 2004, an experiment was conducted to measure the influence of structured task and negotiation sequence on solution and process quality, conflict, and quality of the group's solution. In the formal experiment, there were 32 groups with eight groups per condition. The total subject population was 192, with drop-out the total number participating was 174. The experimental subjects were students taking a software engineering or software-development related courses.

Most hypotheses were tested using items from the post-task questionnaire. These scale items were tested for validity in ensure the responses were reliable. The most commonly used and preferred measure of reliability is the Chronbach's Alpha calculation. This measure has a range from 0 to 1 where zero indicates complete unreliability and a one indicates perfect reliability. In order for a scale to be considered valid and reliable, a Chronbach's Alpha coefficient of 0.70 or higher is necessary (Judd, Smith, and Kidder, 1991). A two–way between-groups analysis of variance was conducted to explore the impact of negotiation sequence and task structure on the variables as measured by the Post-Task questionnaire**.**

Subjects completed and signed a consent form and a background questionnaire prior to starting the experiment. Most variables were measured with an on-line post-task questionnaire provided to each subject after participating in the experiment. Other variables were measured by the output from the asynchronous bulletin board (WebBoard) and from subject-generated documentation (daily list of requirement priorities, final agreed list of ranked requirements, final reports). Negotiation convergence was measured by analyzing the WebBoard conferences in which the group conducted their discussions. The time it took to reach consensus and group agreement were analyzed. Group agreement was measured by the amount of activity within the group, as measured by messages posted, and the ability of the group to come to a final consensus on the list of requirements. Finally, perceived solution quality was measured by post-task questionnaire items and Quality of Solution was measured by the group's final report as assessed by two expert software engineering judges. The table below describes the measures used.

Dependent Variables and Measurement Tools

Variable	**Measurement Tool**
perceived solution quality	Post-task questionnaire
solution satisfaction	Post-task Questionnaire
process satisfaction	Post-task Questionnaire
conflict	Post-task Questionnaire
negotiation convergence	Subject-generated documents and WebBoard conferences
quality of solution	Expert judge's analysis of final reports

RESULTS

Hypotheses	**Result**
Groups following a negotiation sequence will perceive higher quality solutions as compared to groups with no specified negotiation sequence.	*Supported* *p = .031*

H4.b. Groups following a structured task will be at least as likely to be satisfied with the group's solution as those groups with an unstructured task.	*Supported* *p = .003*
H5a. Groups in structured conditions will converge faster to a consensus and show less disagreements in this convergence process than those in unstructured conditions.	*Supported* *p = .030*

DISCUSSION

Three of the twelve hypotheses were supported, two for structured task and one for negotiation sequence. Two were supported for the interaction. The results show that a structured task provides the subjects with more satisfaction with the group solution than those that did not follow such a structure. Groups perceived higher solution quality when they followed a structured task and had no negotiation sequence specified. Hypotheses H3b was reversed indicating that groups that did not follow a negotiation sequence nor structured task achieved a higher satisfaction with the process than those that did. Those groups that followed a task structure and no negotiation sequence had significantly higher solution satisfaction than those that didn't. Finally, conditions that followed a task structure converged faster to agreement in their WebBoard conferences.

The results of the experiment do not support the use of asynchronous negotiation as strongly beneficial. However, due to the nature of the subjects, time constraints, and lack of investment of the subjects, this is can be understood. One of the premises of the exercise is that there is federal or state funding for local government officials to work together to achieve their goals of a collaborative emergency response system. This external funding is a powerful motivator. The subjects in this study understood this important feature. In addition, students are not always motivated to participate effectively in such experiments and therefore do not focus on the task as effectively as industry subjects. The next iteration of this research is planned to involve active emergency responders. Understanding the task and the role of the stakeholders will be inherent with this group. By their nature, this group of stakeholders will understand the task and the purpose of the exercise. Therefore the learning curve in those areas will be greatly reduced, allowing the focus on the actual negotiation to take precedence and increasing the quality of both the process and result of the collaborative exercise. Using the current configuration without changes to the requirements lists will provide more insight into the intricacies of asynchronous negotiation. Along with this goal, a new questionnaire can be designed to elicit opinions from the emergency response personnel about their experiences with current systems and ask them to suggest other requirements they see valuable and may have been missed, and if the current requirements lists needs modification.

Further iterations of this research may modify the structure of the experiment to include an initial group meeting to establish trust and to initiate discussions of the experimental task. Adding other complementary modes of communication will also be considered. A face-to-face group may be added to explore how their negotiations progress as compared directly to the asynchronous groups. At this stage, there will be some modifications to focus more closely on the work of creating requirements that reflect the actual experience of the emergency response stakeholders. This extremely valuable information will be captured due to the asynchronous mode of the experiment and the reasoning behind the choices will provide valuable data for researchers and responders alike. We believe this research provides a rich environment for exploration of emergency response software requirements. Providing guidance for enhancement of such systems and contributing to improved distributed requirements negotiation are goals in future research.

REFERENCES

1. Blackburn, J., Scudder, G., Van Wassenhove, L. (2000). "Concurrent Software Development", Communications of the ACM, 200-214.
2. Boehm, B., Bose, P. (1994, October). "A Collaborative Spiral Software Process Model Based on Theory W", Proceedings, Third International Conference on the Software Process, 10-11, 59-68.
3. Bray, I. K. (2002). An Introduction to Requirements Engineering, Addison Wesley, Essex, UK. Brooks, F. (1975). The Mythical Man-Month, Addison Wesley.
4. Bubenko, J. A., jr. (1995). Challenges in Requirements Engineering (invited talk), Second IEEE International Symposium on Requirements Engineering, 27-29 March, York, U.K., IEEE Computer Society Press.
5. Conboy, K., Lang, M. & Barry, C. (2002). "An Investigation of the Use of Requirements Prioritisation in Web-Based Information Systems Development" In O'Toole, T. et al. (eds), Proceedings of 5th Irish Academy of Management Conference, Waterford, Ireland, September 5-6. WIT Business School: Waterford.
6. Conkin, J, Begeman, M., "gBIS, A tool for exploratory policy discussion." Proceedings of CSCW, October 1988, and ACM Transactions of Office Information Systems, October 1988.

7. Damian, D. (1999). "Challenges in Requirements Engineering", Research Report, 64/08.
8. Deek, F. P., and McHugh, J., "Computer-Supported Collaboration with Applications to Software Development", Kluwer Academic Publishers, 2003.
9. Egyed, A., Boehm, B. (1997, August). "Comparing Software System Requirements Negotiation Patterns", Proceedings of INCOSE-7, 269-276.
10. Green, W. G., (2001), "E-emergency management in the USA: a preliminary survey of the operational state of the Art", International Journal of Emergency Management, Vol. 1, 2001.
11. Grünbacher, P., Koszegi, S., Halling, M., Biffl, S., "Integrating Concepts of Negotiation Theory into Software Requirements Negotiation", Working Paper OP2003-09, 7/11/2003.
12. Grünbacher, P., Hofer, C. (2002). "Complementing XP with Requirements Negotiation", 3rd International Conference on eXtreme Programming and Agile Processes in Software Engineering, Alghero, Sardinia, Italy.
13. Hall, T., Beecham, S., Rainer, A. (2002, April 8-10). "Requirements Problems in Twelve Software Companies: An Empirical Analysis", IEEE Proceedings of the Conference on Empirical Assessment in Software Engineering, Keele University.
14. In, H., Roy, S. (2001). "Issues of Visualized Conflict Resolution", International Symposium on Requirements Engineering, Toronto, Canada.
15. Judd, C. M., Smith, E. R., and Kidder, L. H. (1991). Research Methods in Social Relations, 6th edition, Orlando: Holt Rinehart and Winston, Inc.
16. Karlsson, J., Ryan, K. (1996). "Supporting the Selection of Software Requirements", Eighth International Workshop of Software Specification and Design, 146-149.
17. McGrath, J. (1984). A Typology of Tasks, Groups: Interaction and Performance, Prentice Hall, Englewood Cliffs, NJ, 60-66.
18. Turoff, M. , Chumer, M., Van de Walle, B., Yao, X., "The Design of Emergency Response Management Information Systems (ERMIS)", Prepublication copy submitted to JITTA , (Journal of Information Technology Theory & Application) , 2003.
19. Turoff, M. (2001). "Emergency Response Information Systems: Past and Future", CACM.