

# Asynchronous Decision-Making in Distributed Teams

Qing Li  
qli03@syr.edu  
Kevin Crowston  
James Howison

Robert Heckman  
rheckman@syr.edu  
Syracuse University  
Hinds Hall  
Syracuse, NY 13244  
1-315-443-2911

Eileen Allen  
eeallen@syr.edu  
U. Yeliz Eseryel  
Andrea Wiggins

## ABSTRACT

Extensive use of CSCW applications can influence group decision-making practices. Unlike previous research focused on the influence of synchronous ICTs, our study examines how group decisions are made in asynchronous communication channels. Our inductive qualitative analysis of 360 decision episodes of six FLOSS projects revealed diversity in decision-making practices, which appears to be related to differences in project effectiveness and task type. We also find that standardization of procedures through CSCW tools transforms the nature of some software development work from non-routine to standard procedure.

## Categories and Subject Descriptors

K.4.3 [Computers and Society]: Organizational Impacts – computer-supported collaborative work.

H.4.3 [Information Systems Applications]: Communications Applications – electronic mail.

## General Terms

Human Factors.

## Keywords

CSCW, Decision Making, FLOSS, Open Source Software Development

## 1. INTRODUCTION

An important stream of Computer Supported Collaborative Work (CSCW) research examines ways to support group decision-making [6]. Much of this research has focused on the influence of synchronous information and communication technology (ICT) on decision outcomes. Little prior research has studied the use of asynchronous ICTs to support group decision-making processes [1].

We examine the effects of technology on decision-making in Free/Libre Open Source Software (FLOSS) teams that work primarily through asynchronous ICTs. FLOSS is a broad term encompassing software developed and released under licenses allowing access to and use of the source code. FLOSS developers contribute from around the world, rarely meet face-to-face, and

coordinate activity asynchronously [7].

Software development is usually considered a non-routine task: it is unstructured, uncertain, and requires nontrivial communication and coordination among developers [3]. Understanding decision processes in these naturally occurring distributed teams may provide insight into the work practices of other types of distributed teams that depend on asynchronous ICTs for task support.

## 2. GROUP DECISION-MAKING IN CSCW LITERATURE

Widespread use of computer-supported decision systems has generated research attention to the impact of technology on group decision-making. Synchronous and asynchronous technologies lead to different group decision-making processes from those observed in traditional face-to-face groups [4]. Although email is a commonly used tool in organizations, most prior research focuses on synchronous decision-support systems [1]. Our research contributes a longitudinal study of the asynchronous decision-making interactions that occur in six naturally occurring teams, focusing on participation patterns and processes.

## 3. METHODS

We adopted a multiple case study methodology, primarily focused on content analysis of decision-making discussions. Developer email lists and forums were used as the data source since they are the primary communication channels for FLOSS teams. Data were collected for six FLOSS projects established in SourceForge, selected for maximum variation in project success and product complexity. Project success was measured according to Crowston et al.'s FLOSS multivariate effectiveness model, which includes downloads, page views, online activities and retention of developers [2]. Three projects develop Enterprise Resource Planning (ERP) systems (Compiere, WebERP, Apache OFBiz) which exhibit high product complexity, and three develop Instant Messenger (IM) clients (Gaim, aMSN, Fire), which has low product complexity.

The decision episode was adopted as the primary unit of coding and analysis. A decision episode is a sequence of messages that begins with a triggering message presenting an opportunity for choice, followed by discussion and an announcement of a decision. We randomly sampled 20 decision episodes from three comparable time periods in each project's lifetime: beginning and ending periods, comprised of the first and last twenty decision episodes as of May 2006, and twenty episodes surrounding a major release approximately halfway between the beginning and ending periods. Episodes were coded on number of messages per episode, duration of the episode (in days), and number of participants in the episode.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Conference '04, Month 1–2, 2004, City, State, Country.  
Copyright 2004 ACM 1-58113-000-0/00/0004...\$5.00.

## 4. FINDINGS

Among 360 decision-making episodes, we found 258 software-modification decision episodes and 102 non-software decision episodes. Software modification (SM) episodes focus on daily technical decisions that are the primary work of the team. The second type, non-software (NS) episodes generally do not result in a change in software code, though they may exert an influence on the future of project development. Types of decision opportunities that triggered decision episodes are shown in Figure 1.

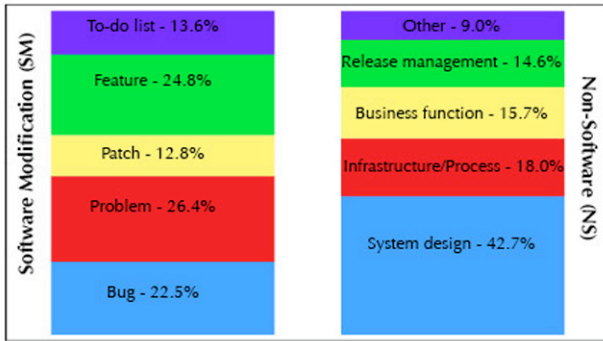


Figure 1: Triggers of decision episodes by type.

We found significant differences between SM and NS decision episodes in terms of duration, number of messages, message density (how many messages per day) and number of participants (Table 1). The NS episodes are longer, more active and attract more participants than SM episodes.

Table 1. Comparison of decision episode types

Item	Episode Type	Mean	Significance level
Duration	SM	4.11	F=1.76; df=1; p=.19
	NS	5.21	
N Messages	SM	5.60	F=14.11; df=1; p<.01
	NS	8.21	
Density	SM	2.78	F=7.11; df=1; p<.01
	NS	3.99	
N Participants	SM	3.05	F=21.99; df=1; p<.01
	NS	4.15	

In comparing IM and ERP projects, we found that their participation patterns were similar for SM issues but more variance was seen in NS episodes. There were also differences in the overall participation patterns between the projects for both ERP and IM. For example, Compiere required more time but fewer messages to reach a decision than any of the other projects. By contrast, aMSN's decisions were accomplished with more messages over fewer days than the other IM projects.

## 5. DISCUSSION

The majority of FLOSS project decision opportunities (72%) focused on daily technical discussions, while the rest (28%) were more complex. We found significant differences in participation patterns between SM issues and NS issues. Generally speaking,

the decisions related to software-modification were made individually, and independently. They needed less time and less coordination among developers, and the results of a decision are presented as lines of source codes checked into CVS. However, decisions related to issues beyond the software seemed to be more complicated, uncertain and beyond an individual's capabilities. The results of these decisions will not immediately be committed into CVS, and may influence software infrastructure and project future. These issues attracted more attention from the extended community with attempts to reach group consensus.

These findings also reflect on the nature of task. Task routineness refers to the level of task variety and analyzability [5]. Routine tasks have low variety and high analyzability, while non-routine tasks require more interactions. Software development is generally considered to be non-routine, but with the use of modularity and version control technologies, FLOSS teams transformed some traditionally defined non-routine tasks into routine tasks. Daily technical decisions are often made with few interactions, and sometimes none at all.

In addition, project characteristics and audiences may affect decision-making processes. IM and ERP showed similar participation patterns in software modification decisions through standardized working procedures. In participation patterns for non-software decisions, however, significant differences were evident as IM projects were more active and energetic in these discussions than ERP projects. Additional work is needed to examine the sources of diversity in decision-making processes.

## 6. ACKNOWLEDGMENTS

This research is supported by a grant from the National Science Foundation, NSF IIS 05-27457.

## 7. REFERENCES

- [1] Baltes, B.B., Dickson, M.W., Sherman, M.P. and Bauer, C.C. Computer-Mediated Communication and Group Decision Making: A Meta-Analysis. *Organ. Behav. Hum. Dec.* 87, 1(2002), 156-179.
- [2] Crowston, K., Howison, J., and Annabi, H. Information Systems Success in Free and Open Source Software development: Theory and Measures. *Lect. Notes. Comput. Sc.* 11, 2(2006): 123-148.
- [3] Kraut, R.E. and Streeter, L.A. Coordination in Software Development. *Commun. ACM* 38, 3(1995), 69-81.
- [4] McLeod, P.L. New Communication Technologies for Group Decision Making. In Hirokawa, R.Y. & Poole, M.S. (Ed.) *Communication and Group Decision Making* (2nd). Sage Publications, Inc (1996), 215-224.
- [5] Perrow, C. A. Framework for the comparative analysis of organizations. *Am. Soc. Rev.* 32 (1967), 194-208.
- [6] Pinelle, D. and Gutwin, C. A Review of Groupware Evaluations. *WET ICE'00* (2000).
- [7] Raymond, E. S. The cathedral and the bazaar. *First Monday*, 3, 3(1998).