

Stability and change in self-organizing technology-supported groups
Evidence from genres of communication in free and open source
software development

(Author suppressed for review)

29 October 2007

Introduction

New organizations supported by information and communications technologies, such as open source software development and Wikipedia, have recently emerged with great and somewhat surprising success. These organizations are emergent, self-organizing and appear able to change and adapt as success breeds new challenges, all without undermining the creative—even emancipatory—collaborative experience.

This abstract proposes a poster presentation for the iConference 2008, drawing on dissertation work in progress with an intended completion date of April 2008. Therefore the poster would present the conceptual framework, method and preliminary results at a time when the author can still incorporate feedback from the iSchool community.

The dissertation work described in this proposal will further our understanding of the emergence, stability and change in the practices of self-organization, through a systematic historical study of two comparable Free and Open Source Software development projects. Specifically it examines emergence, stabilization and change in the genres and genre-systems the projects use to communicate.

The research questions

1. What are the organizational practices of these FLOSS projects?
2. In what way are these practices linked to effectiveness over time?
3. How and why do these practices emerge, stabilize and change?

The importance of the study

This study is important and timely in three ways. Firstly self-organizing teams, supported by information and communications technologies, are rising in importance throughout organizations as their number and areas of involvement rise. Secondly, the study of organizational change (and stability) is independently important as organizations attempt to improve in rapidly changing contexts. Together the study of organizational change in self-organizing teams provides the right context to examine a key theoretical question in the study of organizations, whether (or under what circumstances) organizations are essentially stable, with short periods of change, or essentially always in flux, with periods of seeming stability disguising continuous, incremental change (Orlikowski, 1996; Pettigrew et al., 2001). Finally Free (Libre) and Open Source software (FLOSS) development is not only important in itself but is also a canonical and useful domain to study the issues outlined above.

Conceptual Framework and Literature Review

In order to empirically answer these research questions a solid, operationizable conceptual framework, or theoretical base, is needed. The conceptual framework must be able to a) conceptualize organizational practices so that stability and change can be observed, b) understand effectiveness in the FLOSS context. and c) explain both ‘how’ and ‘why’ stability or change occurs.

The genre approach to organizational practices

Yates and Orlikowski define genres of organizational communication as “socially recognized types of communicative actions—such as memos, meetings, expense forms, training seminars—that are habitually enacted by members of a community to realize particular social purposes” (Orlikowski and Yates, 1994, p 54). They are a type of institution, which Barley and Tolbert (1997) identify as “historical accretions of part practices and understandings that set conditions on action” (p. 99). Genres provide an operationalization of organizational routines (Feldman and Pentland, 2003) appropriate for the distributed team context.

Every domain of activity has characteristic patterns of communication. In the academic world the submission of a research paper follows a characteristic sequence of documents from many participants, from blinded submissions, reviews, editor’s letters, response to reviewers and acceptance or rejection letter. The application of the genre approach to the FLOSS context proposes that the communications that FLOSS members craft also follow particular patterns seen as legitimate by teams.

Orlikowski and Yates (1994, p. 54) argue that genre, and genre systems, can be discovered by observing “why, what, who/m, how, when and where” of communications and sequences; the features of genre are purpose, content, participants, form, rhythm and timing, and venue. Together “the set of genres that are routinely enacted by members of the community” form a “genre repertoire.” Stability and change is possible to identify because, with historical detail extracted from archives as described below, a catalogue of genres in use at different times will reveal change, both in the relative use of different genres but also within genres (and genre systems), as features are increasingly or decreasingly present.

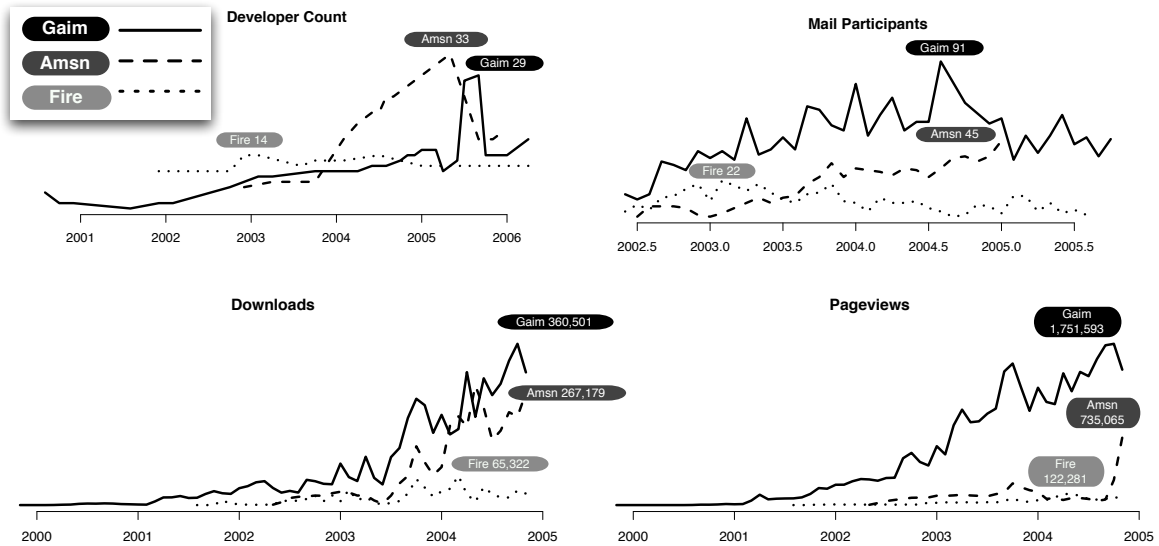
Emergence, stability and change

The genre approach provides an appropriate conceptualization of practices and an explanation of ‘how’ they both achieve some stability (through structural observation and repetition) and change (through importation and improvisation). However the genre approach has to date only discussed ‘immediate’ causes, or perhaps opportunities, for genre change, drawing on Gersick and Hackman (1990). However, particularly in a self-organizing group, deeper explanations of the underlying drivers of these opportunities are desirable, allowing the research to explain, for example, why novelty is encountered or milestones chosen or the structure of the group altered.

While there is not sufficient space here to describe it the literature touching on change in the FLOSS context does suggest a generative model of change (Senyard and Michlmayr, 2004; Oh and Jeon, 2004; Koch and Schneider, 2002; Robles et al., 2005; O’Mahony, 2003), which draws on the useful taxonomy of organizational change “generative mechanisms” provided by Poole and Van de Ven (2004).

These studies suggest a compound generative mechanism for change which turns on a dialectical tension between the desire to attract participants and the need to maintain cohesion in the team. The lifecycle effects of code growth and audience diversification eventually tip the balance so that the key organizational challenge shifts from inclusion to exclusion. Such changes ought to be visible through the method below as certain genres ‘wax and wane’. Failure to adequately adjust might be the cause of reduced effectiveness. It should be noted, however, that this research is not designed to test this specific model directly. Different models of generative change might emerge from the method described below, but they would be similarly conceptualized and supported with evidence of trends in practices and effectiveness.

Figure 1: *Gaim and Fire differ in their ability to sustain early success*



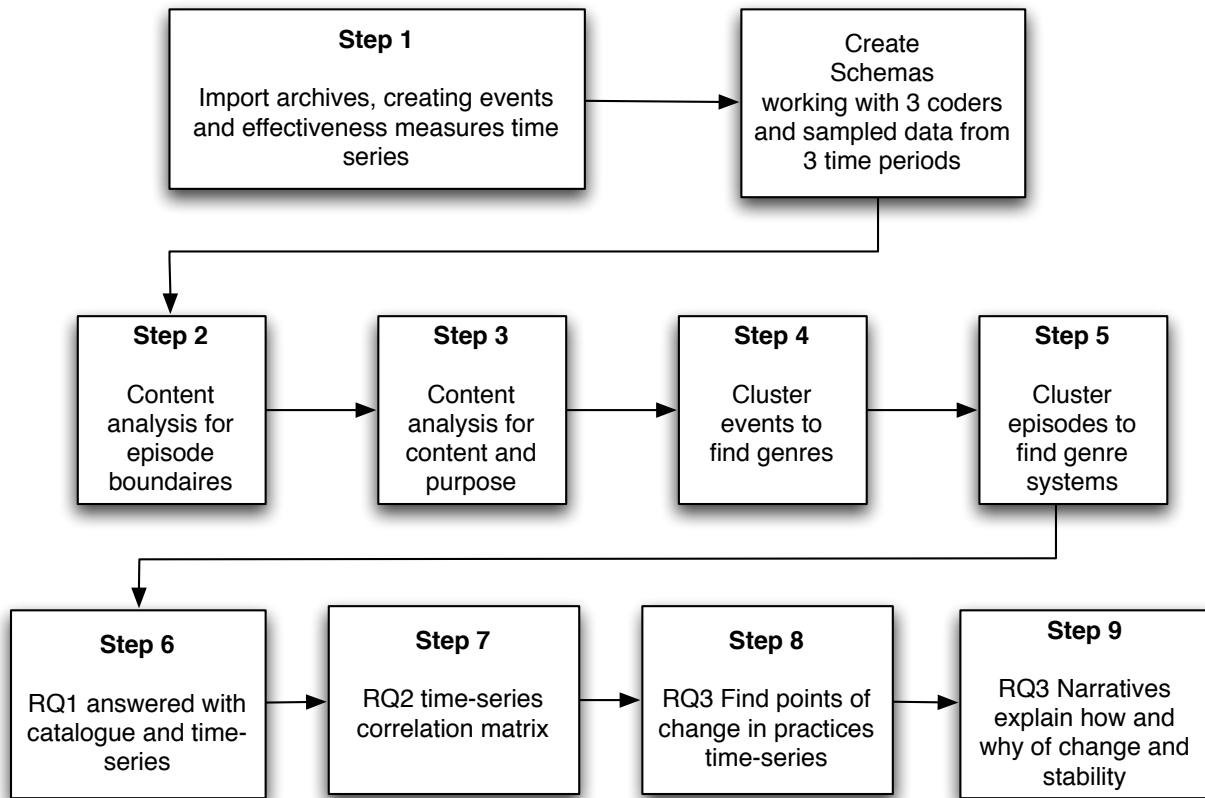
Conceptualizing effectiveness

The third research question links practices and effectiveness over time. The FLOSS context provides a challenge for creating appropriate measures of effectiveness. The multiple measures chosen for this study reflect the four categories suggested by Crowston et al. (2003, 2006, 2006): a) system creation and maintenance (such as code growth and developer attraction), b) system quality (such as complexity metrics), c) system use (such as downloads and pageviews) and d) system consequences (such as competitor reactions). Figure 1 shows sample time-series for comparable projects.

Proposed Method

It is proposed to pursue the research questions above through a comparative case study of two comparable FLOSS instant messaging projects, Gaim and Fire, which differ in their ability to sustain their early success; Gaim has gone from strength to strength, while Fire eventually stalled. An outline of the overall method is shown in Figure 2.

Figure 2: *The proposed steps of this research*



Two content analytic schemes are being developed. The first allows the recognition of *episodes* in the data: sets of communications linked by topic and shared attention. The second scheme describes the *purpose* of the emails, based on speech acts theory (Goldstein and Sabin, 2006; Searle, 1969), as well as the *content* (a ground up literal description, such as “description of application behavior”). These codes are combined with automatically genre attributes (“where, how, when and who”) to describe each communication on the six dimensions of genre. Clustering techniques will then be used to identify repeated patterns and these clusters will be interpreted as genres. Genre systems will be identified through clustering based on sequence analysis of genres in episodes (Bakeman and Gottman, 1997; Abbott, 1995, Abbott, 1995).

The question of practices in use, RQ1, will be answered with a typology of genres and

genre-systems that make up the project’s “genre repertoire”, together with a deeply linked catalogue of examples. This catalogue can be presented over time using both absolute and relative frequency to see how genre and genre-system use has changed.

RQ2 will be answered by regressing these time-series against the time-series of effectiveness measures. The effectiveness time-series may also indicate ‘breaks’ which show rapidly rising or falling effectiveness and such ‘breaks’ can be compared with changes in genre, or genre features, around those periods. For example it would be possible to test whether an observed lengthening of particular genres (perhaps discussions about including new features, or interminable procedural discussion) was associated with effectiveness trends such as declining developer participation.

The question of emergence, stability and change, RQ3, will be answered in two ways, one quantitative and one qualitative. Quantitative indicators of stability and change will be discovered by analyzing the genre and genre-system time-series to discover periods with statistically significant ‘breaks’, using interrupted time-series experiment techniques (Hamilton, 1994). The question of continuous vs punctuated change ought to be visible in the smoothness of the time-series, especially those graphing features of particular genres.

Such ‘breaks’ will indicate periods for further qualitative investigation producing micro-narratives of the emergence and change of particular genres, and macro-narratives of change in the projects, drawing on evidence from the relative frequency of particular genres and genre-systems. For example intentional change may be explicitly discussed and planned just prior to a ‘break.’ Or the arrival of a new genre may be precipitated by the arrival of a new participant particularly associated with it, indicating importation. The emergence of quality assurance messages may be accompanied by the emergence of an audience for stability.

Conclusion

The research will further our understanding of change in self-organizing groups as well as the ‘via document’ collaboration that increasingly constitutes work today.

Bibliography

- Abbott, A. (1995). Sequence analysis: New methods for old ideas. *Annual Review of Sociology*, 21:93–113.
- Bakeman, R. and Gottman, J. M. (1997). *Observing interaction : an introduction to sequential analysis*. Cambridge University Press, New York, 2nd ed edition.
- Barley, S. R. and Tolbert, P. S. (1997). Institutionalization and structuration: Studying the links between action and institution. *Organization Studies*, 18(1):93–117.
- Crowston, K., Annabi, H., and Howison, J. (2003). Defining open source software project success. In *Proc. of International Conference on Information Systems (ICIS 2003)*.
- Crowston, K., Howison, J., and Annabi, H. (2006). Information systems success in free and open source software development: Theory and measures. *Software Process: Improvement and Practice*, 11(2):123–148.
- Feldman, M. S. and Pentland, B. (2003). Reconceptualizing organizational routines as a source of flexibility and change. *Administrative Science Quarterly*, 48(1):94–118.
- Gersick, C. J. G. and Hackman, J. R. (1990). Habitual routines in task-performing groups. *Organizational Behavior and Human Decision Processes*, 47:65–97.
- Goldstein, J. and Sabin, R. E. (2006). Using speech acts to categorize email and identify email genres. In *HICSS '06: Proceedings of the 39th Annual Hawaii International Conference on System Sciences*, page 50.2, Washington, DC, USA. IEEE Computer Society.
- Hamilton, J. D. (1994). *Time series analysis*. Princeton University Press, Princeton, N.J.
- Koch, S. and Schneider, G. (2002). Effort, cooperation and coordination in an open source software project: Gnome. *Information Systems Journal*, 12(1):27–42.
- Oh, W. and Jeon, S. (2004). Membership dynamics and network stability in the open-source community: the Ising perspective. In *Proceedings of International Conference on Information Systems (ICIS) 2004*.
- O’Mahony, S. (2003). Guarding the commons: how community managed software projects protect their work. *Research Policy*, 32(7):1179–1198.
- Orlikowski, W. J. (1996). Improvising organizational transformation over time: A situated change perspective. *Information Systems Research*, 7(1):63–92.
- Orlikowski, W. J. and Yates, J. (1994). Genre repertoire—the structuring of communicative practices in organizations. *Administrative Science Quarterly*, 39(4):541–574.
- Pettigrew, A. M., Woodman, R. W., and Cameron, K. S. (2001). Studying organizational change and development: Challenges for future research. *Academy of Management Journal*, 44(4):697–713.

- Poole, M. S. and Van de Ven, A. H. (2004). Theories of organizational change and innovation processes. In Poole, M. S. and Van de Ven, A. H., editors, *Handbook of Organizational Change and Innovation*, pages 374–397. Oxford University Press, New York, NY.
- Robles, G., Amor, J. J., González-Barahona, J. M., and Herraiz, I. (2005). Evolution and growth in large libre software projects. In *The 8th International Workshop on Principles of Software Evolution*, Lisbon, Portugal.
- Searle, J. (1969). *Speech Acts*. Cambridge University Press, Cambridge, UK.
- Senyard, A. and Michlmayr, M. (2004). How to have a successful free software project. In *Proceedings of the 11th Asia-Pacific Software Engineering Conference*, pages 84–91, Busan, Korea. IEEE Computer Society.