

A Dynamic Feedback Framework for Studying Growth Policies in Open Online Collaboration Communities

Vedat G. Diker

College of Information Studies - University of Maryland
vdiker@wam.umd.edu

ABSTRACT

This paper reports the findings of a study which developed a theoretical framework for studying the growth policies in a special kind of virtual communities: open online collaboration communities. The study made use of dynamic feedback modeling and simulation, and a series of interviews with the members of an open online collaboration community that specializes on instructional material development and dissemination. The paper provides recommendations for practice in online communities that specialize in open online collaboration.

Keywords

Virtual communities, online communities, collaborative content development, open source, feedback dynamics, system dynamics, simulation.

INTRODUCTION

Open online collaboration communities (OOCs) are virtual communities that specialize on developing and disseminating stand-alone information products such as software and courseware. This paper introduces a dynamic feedback simulation model, which represent the causal relationships between determinants of success in OOCs such as community size, product size and product quality. The simulation model, which replicated the behaviors of successful and unsuccessful OOCs, was used as a basis for developing and administering a set of interviews with the members of an actual OOC that specialized on developing and disseminating instructional materials. The findings of the simulations with the model and the interviews were then synthesized to build a causal framework for explaining the dynamics of growth in OOCs. Initial findings about strategic growth problems and remedial policies are also discussed in this paper.

RESEARCH MOTIVATION AND CONTRIBUTION

Open online collaboration communities (OOCs) may have substantial influence on how we develop, disseminate and access digital content in the near future. However, the dynamic interactions between the determinants of success in OOCs, such as barriers to entry, motivation, participation, collaboration and the quality of products, have not been fully explored and theorized. The stakeholders in OOCs do not have means to test policies for improving performance. Instead, they rely on a combination of personal experience, intuition and anecdotal guidelines derived from the experiences of other, similar communities.

One important reason for the existing void in the literature about OOCs is that these communities have not been studied as a distinct type of online communities. The general approach in the literature is either to study online communities in a “wholesale” fashion, without regarding their different characteristics, or to study only a very limited subset that would fall into the definition of OOCs, as in the case of open source software development communities.

As a consequence of all these considerations, the study summarized in this paper had two main research purposes:

1. to develop and establish a definition of OOCs, supported by a dynamic feedback framework that is applicable to a range of open online collaboration communities,
2. to outline and analyze several policy options for improving the performance of OOCs.

OPEN ONLINE COLLABORATION COMMUNITIES

The study defines open online collaboration communities (OOCs) as online communities that are formed by loosely connected groups of people, who use the Internet as a medium for carrying out collaborative projects for producing and improving a wide range of stand-alone information products, such as software, and courseware. The study uses two taxonomies that exist in the literature for placing OOCs within the overall population of online communities. First is Hagel and Armstrong's (1997) classification, which classifies online communities based on the needs of the community members.

OOCs fit in the definition of communities of transaction, based on Hagel and Armstrong's classification. The other classification that the study uses was suggested by Stanoevska-Slabeva and Schmid (2001). From Stanoevska-Slabeva and Schmid's classification's standpoint, OOCs fall in the design communities sub-class within the task-and-goal-oriented class (Stanoevska-Slabeva and Schmid, 2001). Due to limited space, readers are referred to related work about the details of these classifications.

Arguably the most prominent examples of OOCs are open source software development (OSSD) communities. OSSD communities are groups of loosely connected programmers, who use the Internet as a medium for collaboratively developing, improving, and disseminating software (O'Reilly, 1999). Only a small fraction of those programmers gain direct tangible benefits in return to their contributions. The majority are motivated by indirect or intangible benefits (Raymond, 2001). Despite the lack of monetary motivation factors, Open Source software movement gave way to the production of high quality free software which can compete with leading proprietary software, as in the example of the Linux operating system (Diker and Scholl, 2001).

RESEARCH DESIGN AND METHODOLOGY

The study was carried out in two phases. The first phase involved developing a dynamic feedback simulation model (or a system dynamics model, as it is alternatively called) of a hypothetical OSSD community, which was used as an example of OOCs. During the second phase, the members of an actual OOC that specializes on instructional material development were interviewed in order to test the applicability of the simulation model. Findings of the interviews were integrated with the findings of the model simulations to develop a dynamic framework for explaining the dynamics of growth in OOCs.

System dynamics methodology is widely used to analyze complex, large-scale, non-linear, partially qualitative dynamic socio-economic systems. System dynamics "assumes direct causal relationships between variables that form the system being analyzed, and interdependence of causal factors through feedback loops. Almost every system includes a number of both negative and positive feedback loops [that consist of several variables], which interact and operate simultaneously" (Diker and Scholl, 2001). System dynamics models are represented by structure diagrams, which show the causal relationships in the form of directed arrows connecting the cause variable to the dependent variable. A set of mathematical equations, which demonstrate the algebraic relationships among the variables, accompany these diagrams (Diker and Scholl, 2001).

The open source software development (OSSD) model represents causal relationships between different elements that drive the dynamic behavior of a generic OSSD community. The structural and behavioral details of the model are discussed further in this paper.

The OSSD model was tested extensively for internal validity using standard system dynamics model validation tests such as extreme condition tests, sensitivity analyses and policy analyses. Due to space restrictions, the readers are referred to the source text (Diker, 2003) for the details of the testing methodology and the results of the tests.

After the internal validity tests, the OSSD model was simulated to test a set of policy options for improving community growth, product functionality and product quality. The implications of those policy simulations are discussed further in this paper.

The second stage of the research involved the development, administration and analysis of structured interviews with the leading members of a specific instructional material development community, in order to test the applicability of the OSSD model and its policy implications to other OOCs. The specific community in question is a group of teachers and researchers who develop and disseminate instructional materials for K-12 students. The community in question has gathered around four main organizations or groups.

The interview instrument was designed to gather data that would be fit for qualitative analysis. A purposive, snowball sample of 10 experts from the overall population of the community were used for the interviews. Kvale (1996) found that the number of interviews in current qualitative interview studies tend to be between 5 and 25, with an average of roughly 15. Kvale attributed this to the limited time and resources available for carrying out such interviews. Kvale also suggested that each additional interview will add less to the findings, and the contribution of an additional interview will be negligible once a number roughly between 5 and 25 is reached.

Nine of the ten interviews were administered over the telephone. One interview was administered face-to-face at the request of the subject. The interviews were recorded on audiotape with the approval of the interviewees. For further methodological details about the interviews, see the source text (Diker, 2003).

LITERATURE REVIEW

The OSSD model was based on implications derived from parallels drawn between the literature on theoretical approaches to the study of online communities, and the literature on practices in OSSD communities. Technical components of the model, which were related to software development, were based on existing system dynamics models of software development projects.

Several theoretical approaches were suggested for studying online communities, such as gift economies, public goods, social informatics, and social networks. Some of these approaches were deemed useful for the purposes of developing the model in question, while others were not useful in providing implications for the model.

A theoretical approach suggested for studying online communities that is particularly relevant to this study is the concept of gift economies (Barbrook, 1998, Ghosh, 1998, Kollock, 1999, Bays and Mowbray, 2001). Raymond (2001) suggests gift economies as a viable theoretical approach to the study of OSSD, as well. Gift economies are based on gift exchange as opposed to commodity exchange (Bell, 1991, Carrier, 1991, Gregory, 1982). Gift exchange takes place between parties who have an existing relationship, or are aiming to build an ongoing relationship (Bell, 1991, Carrier, 1991). Furthermore, a gift is not necessarily reciprocated by the giving of a ‘counter-gift’ right away (Bourdieu, 1997). However, the giving of a gift generally implies an unstated expectation of a reciprocation at an indefinite time (Carrier, 1991). An important implication of the gift economies concept for the OSSD model is that a relatively larger community would motivate contributors more, since the probability of generalized reciprocation increases as the number of contributors in the community increases.

The concept of public goods is another theoretical framework suggested for explaining phenomena related to online communities (Millen, 2000, Wasko and Teigland, 2002, Kollock, 1999). Several authors have suggested public goods as an approach for studying OSSD, as well (Bessen, 2002, Hawkins, 2001). Public goods differ from private goods in two ways. First, public goods are “non-excludable”; that is, it would be too hard or too costly to exclude the non-payers from benefiting from a public good. Second, the consumption of public goods is on “non-rival” basis; that is, the consumption of a public good by an individual does not hinder other individuals’ consumption of the same good (Cowen, 1993). Since it is infeasible to exclude non-payers from benefiting from public goods, it is also not feasible to charge for their use (Cowen, 1993). This brings about the problem of lack of interest towards producing and distributing public goods. Kollock (1999) outlines the possible motivation factors for participation in the production of public goods within the context of online communities as such: expectation of generalized reciprocation, building reputation, gaining a feeling of self-efficacy, and altruism. These motivation factors together provided certain implications for the OSSD model:

1. Larger contributor population may decrease motivation.
2. Larger user population may increase motivation.
3. Visibility may motivate contributors more.
4. Feedback channels may increase motivation

Several authors approach to the study of online communities from a perspective called “social informatics” (or “social impacts”) (Hiltz, 1986, Turoff and Hiltz, 1982, Preece, 2000). Social informatics research focuses on the social impacts of information systems (Preece, 2000). The basic argument of this approach is that the design and use of information systems have an impact on the social processes that govern the context in which those information systems operate. Furthermore, information systems, together with social processes, have an impact on the social structures and relationships. Based on these premises, several authors argue that while designing an information system, the effects on the social processes, structures and relationship should be taken into account, and the information system should be designed as a part of the social process it will be “embedded in” (Preece, 2000, Turoff, 1997). Several authors approached the OSSD from the same perspective (Fogel and Bar, 2001, Raymond, 2001). The implications of these approaches for the model in question are:

1. Low barriers to entry to the community and contribution may increase participation.
2. Accessibility and usability of end-products may increase user population.

THE OPEN SOURCE SOFTWARE DEVELOPMENT (OSSD) MODEL

The OSSD model represents the causal relationships among a variety of factors that govern the growth dynamics in an OSSD community, such as the number of developers and users, level of participation, functionality and quality of the product, barriers to entry and contribution. The OSSD model was based on the implications of the literature review summarized above.

The OSSD model was organized in seven sub-sections, and it involved more than 270 variables. Since the scope of this paper is not wide enough to discuss the entire model, only one sub-section is presented here as an illustrative example of the structure of the model. The reader is referred to the source text (Diker, 2003) for a more detailed discussion on the structure and behavior of the model.

Figure 1 portrays a sub-section of the model which represents the developer population, production, product functionality, and motivations related to joining and leaving the community. Here, developers participate in production and add new functionality to the software product. It is assumed that the functionality of a given software product has an upper limit, which would increase gradually. As the achieved functionality approaches that limit, attractiveness of the product for developers decreases. This relationship is based on Raymond's (2001) arguments that a developer joins an open source software project in order to "homestead" a certain portion of the software. By developing the "homesteaded" portion of the software the developer builds reputation and self-efficacy sentiments. When the achieved functionality approaches the functionality limit, the project does not offer many portions to homestead, and this decreases the attractiveness of the project. This also accelerates the rate of 'leaving developers'. As the project approaches the end, more developers leave the project, leaving a smaller number of developers for maintenance purposes.

POLICY RUNS (SIMULATIONS) WITH THE OSSD MODEL

Policy runs involve simulating a dynamic feedback model under a set of alternative policy settings. Policy runs aim to compare the level of improvement in the behavior of key performance measures of the model under each policy setting. In this context, "policy setting" denotes a specific setting of one or more parameters that can be determined by the policy or decision makers of the real system the model represents. For the purposes of the policy runs, the key performance measures of the OSSD model were identified to be community growth (in terms of both users and developers), product functionality and product quality.

The policy runs were based on four basic policy options. Each policy run involved either a single basic policy options or a combinations of these. The four basic policy options were:

Barriers to entry: Potential developer candidates are screened before joining the community and a certain percentage of them are refused.

Barriers to contribution: Contributions from inexperienced developers are screened before being added to the product, and a certain portion of those contributions are rejected.

Debugging emphasis: More developer time is dedicated to debugging activities than under the base conditions of the model.

Coaching emphasis: More developer time is dedicated to coaching of inexperienced developers by leading developers than under the base conditions of the model.

As a general finding, the policy runs showed that any policy aimed at product quality improvement has the potential of slowing product functionality and community growth when pushed beyond a certain level. Furthermore, the marginal quality improvement may decrease substantially as the policy level increases. These two findings together imply smaller quality gains at the expense of larger functionality losses as the policy level increases.

These findings clearly showed that an OSSD community has to consider the trade-off between building functionality and improving quality while developing policies. Based on these findings, this study defines the underlying policy problem in an OSSD community as the tension between building product functionality and improving product quality while sustaining community growth.

In terms of the specific policy options, the two best choices emerged as a pure barriers to entry policy, and a combination of debugging and coaching emphases. These two options provided substantial improvements in product quality (Figure 2) without critically slowing the growth in product functionality (Figure 3) and the developer and user populations. (Figures 4 and 5.)

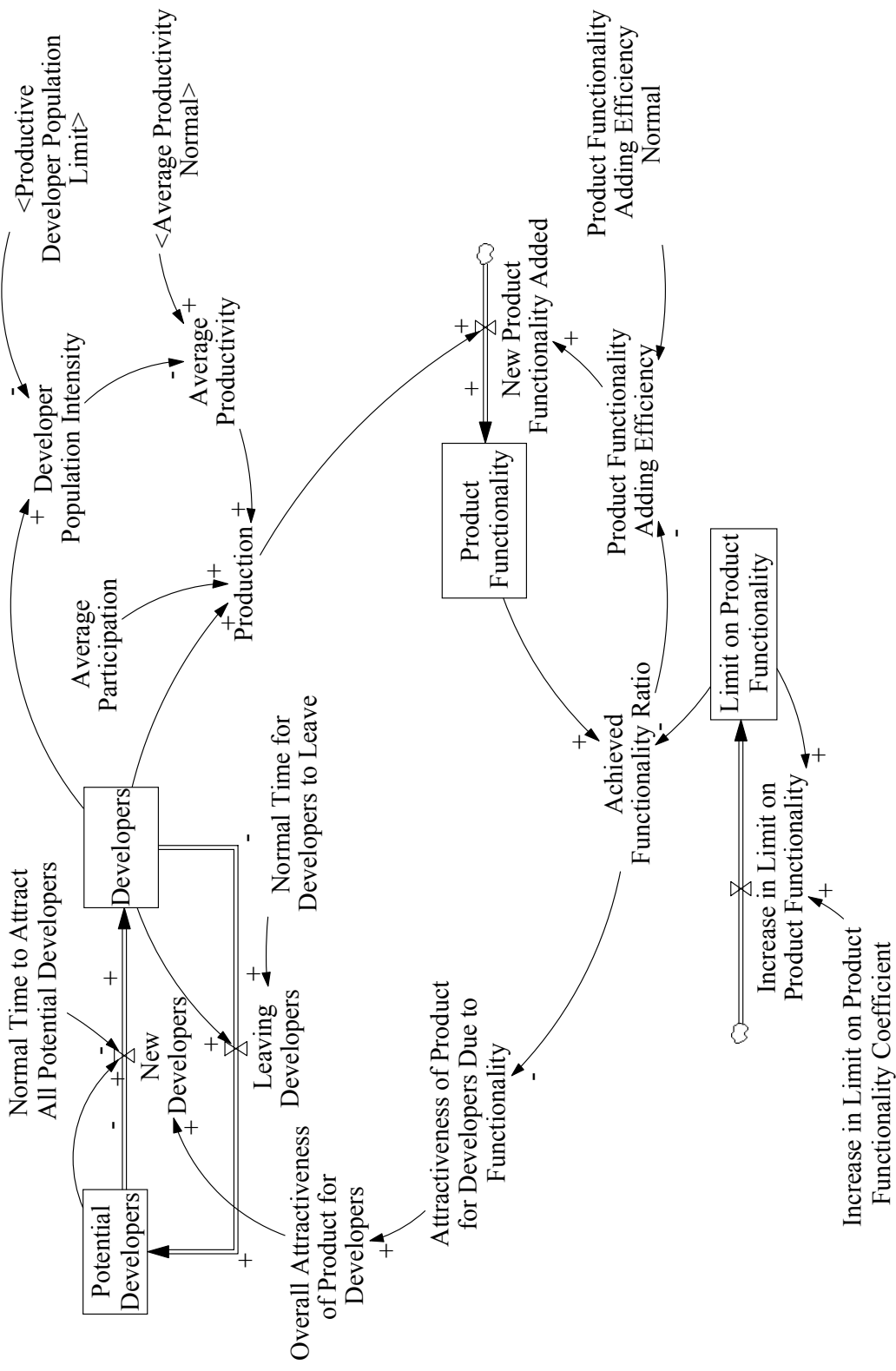


Figure 1. Developers Section of the OSSD Model.

An overall combination of the barriers to entry and debugging and coaching emphases policies yielded higher improvements in perceived product quality than the two alternatives. (Figure 2.) However, the product functionality growth (Figure 3) and community growth (Figures 4 and 5) became much slower under the overall combination policy settings.

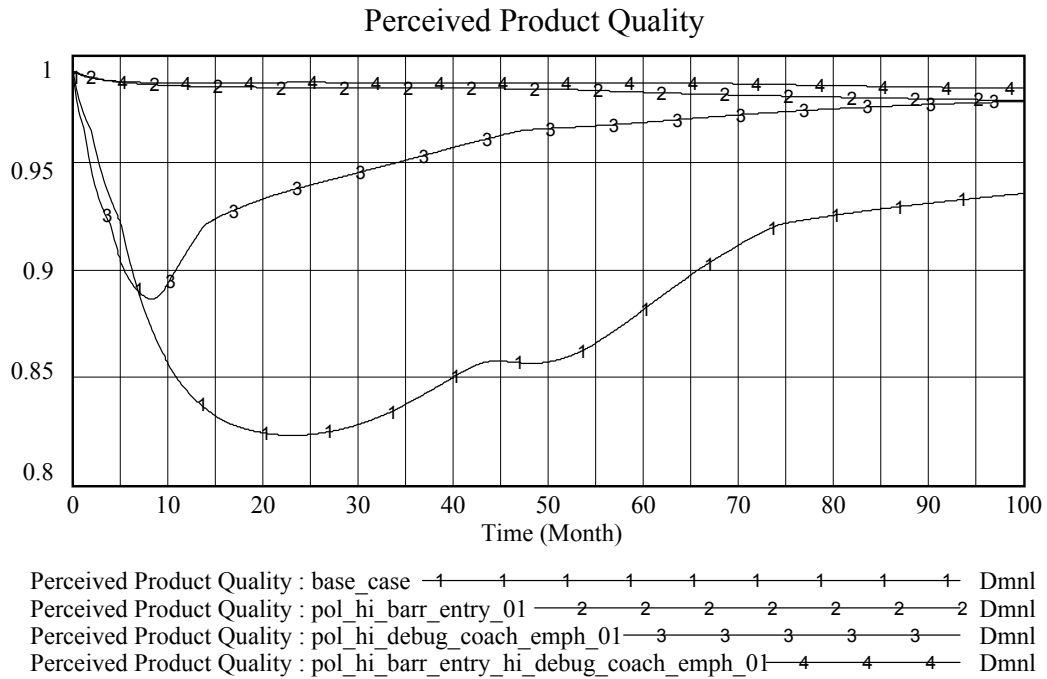


Figure 2. Perceived Product Quality under Higher Barriers to Entry, Higher Debugging and Coaching Emphases, and Overall Combination Policy Settings

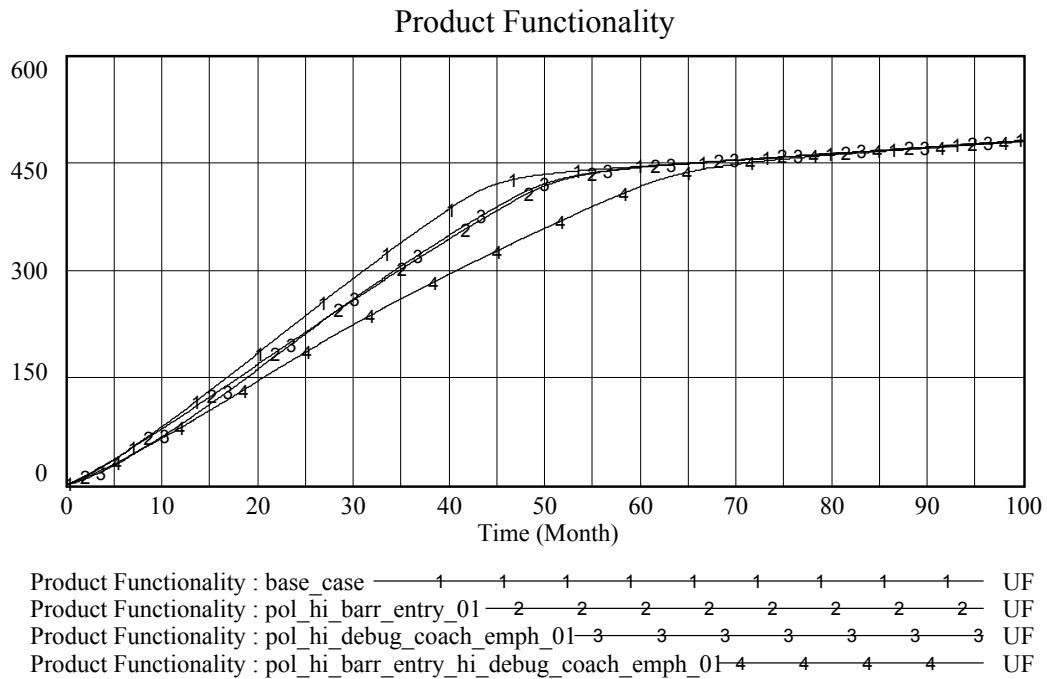


Figure 5. Users under Higher Barriers to Entry, Higher Debugging and Coaching Emphases, and Overall Combination Policy Settings

ANALYSIS OF THE INTERVIEWS

In this study, the main function of the interviews was to test the applicability of the hypothetical open source software development (OSSD) model to the case of a specific instructional material development community. Accordingly, the interviews were analyzed in order to see whether the personal observations and mental models of the interviewees supported or refuted the assumptions and the structure of the model. The analysis involved testing the key model structures and policy options against the personal observations and mental models of the interviewees.

The model was introduced to the interviewees as simplified diagrams of the main positive (reinforcing) and negative (balancing/limiting) loops in the model. Policy options were also introduced using diagrams supported by narrative segments. The interviewees were asked to comment on the model structure and policy options based on their observations and experiences in their community.

The analysis of the interviews showed that the interviewees identified certain structures within the OSSD model as representative of their community. However, some structures of the model were judged by the interviewees as not representative of their community. Some interviewees also suggested additional structural elements to be added to the model in order to bring it closer to a representation of their community. As an example, one specific change suggested explicitly by four interviewees was to link the inflow of new developers/contributors to the stock of users. These interviewees argued that only existing users became new developers in their community, not people from outside of the community.

Most of the interviewees suggested that they had observed some combination of barriers to contribution, debugging emphases and coaching emphases policies implemented in their community. Only three interviewees argued that they had observed a form of barriers to entry implemented in the community; however, they added that that was not done in an *overt* manner. Most interviewees suggested that coaching emphasis would be the most effective option either as a pure policy or in combination with debugging emphasis (reviewing and editing, in the context of the instructional material development community), especially in the long run. More than half of the interviewees argued that implementing a barriers-to-entry policy would be detrimental to community growth, since potential developers/contributors would be turned off by such a practice. However, several interviewees suggested that such a policy would improve product quality.

A DYNAMIC FEEDBACK FRAMEWORK FOR OPEN ONLINE COLLABORATION COMMUNITIES

Implications of the OSSD model and the findings of the interviews were integrated to build a dynamic feedback framework that represents the dynamics of growth in OOCs. The framework is a concise representation of the dynamic feedback structure that underlies OOCs. It has the potential of explaining the phenomena that determine the growth or decline of an OOC. The feedback framework can be used as a basis for developing a generalized dynamic feedback simulation model of an OOC. The causal relationships between the variables of the framework or the feedback loops can be used as hypotheses for empirical research studies. The framework can be further refined based on the findings of such research studies.

Figure 6 shows the dynamic framework. The members of the community are grouped into three: Users, Inexperienced Authors, and Experienced Authors. When users decide to make contributions to the collection they become inexperienced authors. Inexperienced authors become experienced authors as they mature in authoring. There are members that leave at every stage of the maturing process. Inexperienced and experienced authors contribute to the production and build the product or the materials collection. The size and the functionality of the collection influence the number of new users. A larger and more functional (more useful) collection brings more new users, thus forming the main positive (reinforcing) loop.

As authors produce content, they add quality problems to the collection. Experienced authors review the collection, discarding materials that are of very low quality, and choose some other materials for rework. Inexperienced and experienced authors revise and improve the materials chosen for rework. Discarding and reworking materials eliminate certain portions of the quality problems. The amounts of discarded and reworked materials are determined by the quality threshold, which is used by the experienced authors as the benchmark for evaluating the collection. This threshold also affects the ratio of users who become authors. A higher quality threshold means more discarded and reworked material, thus yielding a higher quality level. However, it also means a lower number of new authors. The rationale is that a higher probability of their work being discarded or sent for rework will decrease users' motivation to make contributions and become authors. This follows from the findings of the policy runs based on the higher barriers to contribution option, which suggested that higher rejection ratios yield lower number of new authors (developers). Also, the discussions with the interviewees supported the hypothesis that a considerably high rejection ratio would decrease the motivation to participate in production.

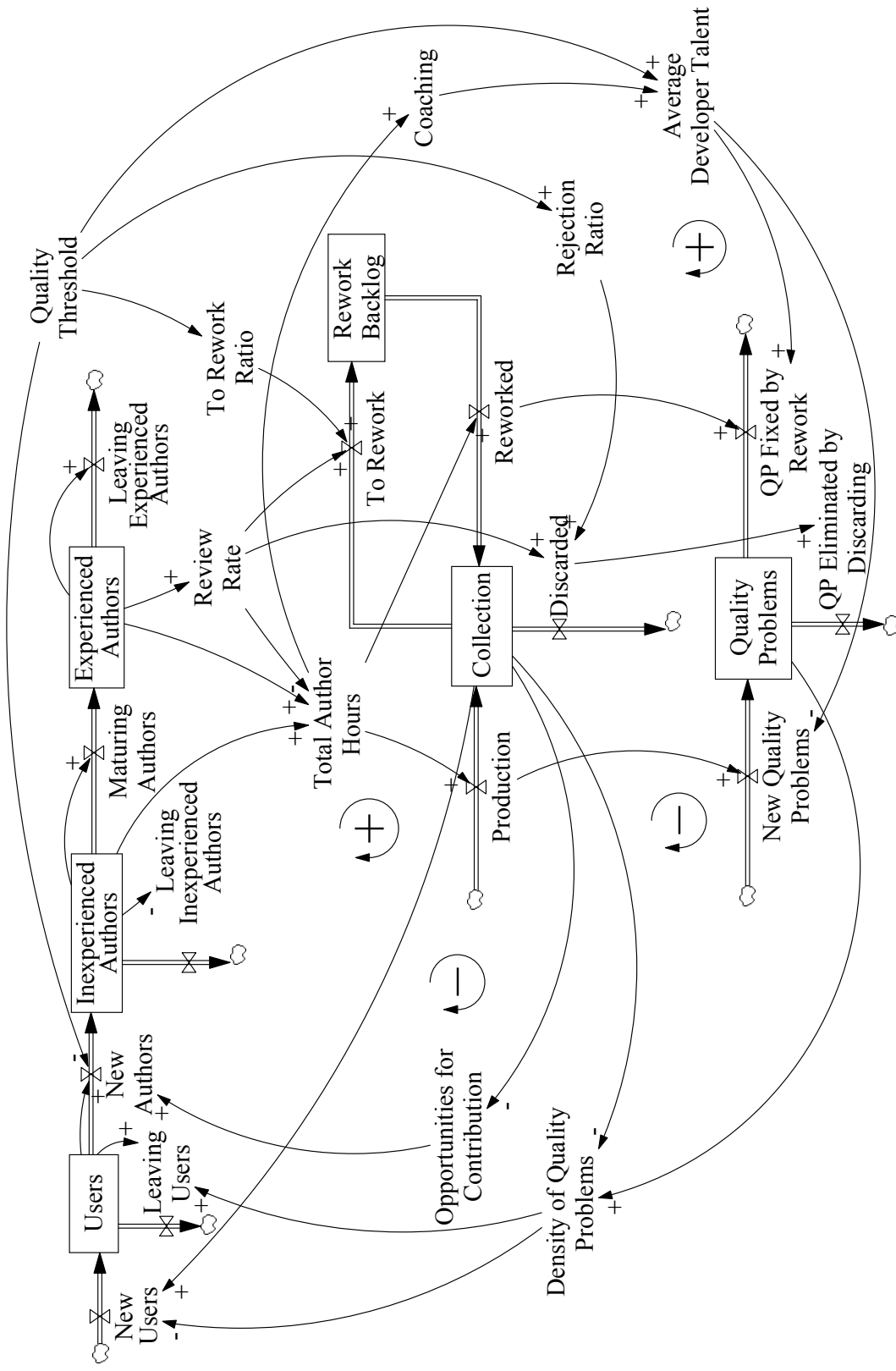


Figure 6. The Dynamic Feedback Framework for OOCs.

Quality threshold also determines the barriers to entry. As the quality threshold increases, the community becomes more selective in accepting new authors, and the number of users accepted into the inexperienced author pool decreases. The density of quality problems, which is defined as the number of quality problems per unit of the collection, determines the rates with which new users join the community, and exiting users leave. A higher density of quality problems would yield a lower number of new users, and a higher number of leaving users. These links form the second main loop, which is a negative (limiting) one. The third main loop of the framework, which also is a negative (balancing) one, is formed by the opportunities for contribution. As the collection gets larger and more functional, opportunities for contribution decrease. Decreasing opportunities for contribution decrease the number of new authors, since potential authors may be discouraged by the lack of vast opportunities for making contributions. This was one of the main limiting loops in the OSSD model. However, the discussions with the interviewees about the existence and effects of such a limiting loop suggested that this loop might come into effect considerably late in the process for some OOCs. In fact, it may not come into effect for some communities that focus on divergent tasks such as instructional materials collections, rather than convergent tasks such as software products. Although many interviewees suggested that this loop was plausible theoretically, they emphasized that they have seen no indication that this loop exists in their community. Thus the link from the collection to the opportunities for contribution and the link from the opportunities to new authors are marked as “questionable” in the final framework, and shown in dashed lines.

Average developer talent and coaching form the fourth loop, which is reinforcing. Coaching increases average developer talent, and as average developer talent increases quality problems decrease, causing a lower density of quality problems. A lower density of quality problems brings more new users, and slows the leaving of the existing users, thus increasing the number of users more quickly. More users mean a higher number of new authors, which increases the number of authors more quickly, and thus provides more author hours available. More author hours close this reinforcing loop by giving way to more coaching. This loop shows its reinforcing effect in the long run, since average talent takes time to build. This was a point that the interviewees emphasized about the effects of coaching. Most of the interviewees suggested that coaching is more effective in the long run.

IMPLICATIONS FOR PRACTICE AND CONCLUSIONS

This paper summarized a research study which developed a dynamic feedback framework to study growth problems in OOCs, and potential policies to deal with those problems. The implications of a system dynamics model of a hypothetical OSSD community and a series of interviews with the leading members of an actual instructional material development community were integrated to build the final dynamic framework.

The study provided implications for practice in OOCs. One important implication is that there is a fundamental tension between building functionality and improving quality of products developed in an open online collaboration community. Furthermore, any policy aiming to improve either one of product functionality or product quality has the potential of impeding the other, if the policy is pushed too far. Consequently, such policies should be implemented with caution and moderation, and both performance measures should be monitored for undesired changes and unintended effects on community growth.

The study also provided certain recommendations for practice in terms of specific policies for improving product functionality, product quality and community growth. Among the many policy combinations tested, the most effective one for product quality improvement was a combination of selecting new contributors, increasing contributor hours dedicated to revising and editing existing materials, and increasing coaching activities between experienced and inexperienced contributors. However, this combination policy had a considerable potential of slowing down product functionality and community growth. This was caused by a decreased total participation level due to lower number of new contributors.

On the other hand, a combination of increasing contributor hours dedicated to revising and editing existing materials, and increasing coaching activities between experienced and inexperienced contributors provided a relatively slower quality improvement. However, this combination policy did not impede product functionality and community growth considerably. Accordingly, it was identified as a potentially attractive policy option whenever a steady growth of users and contributors is desirable.

An overarching policy finding, which was supported strongly by both the simulation runs and the interview data, was that increasing coaching activities between experienced and inexperienced contributors was the most effective policy option in the long run. One particular strength of this option was that it can be implemented in conjunction with any other policy option, and still contribute to the overall improvement in product quality, product functionality and community growth.

REFERENCES

1. Barbrook, R. (1998) The Hi-Tech Gift Economy, *First Monday*, 3 (12), Accessed: January 10, 2003, Available: http://www.firstmonday.org/issues/issue3_12/barbrook/index.html.
2. Bays, H. and Mowbray, M. (2001) Cookies, Gift-Giving, and Online Communities, in *Online Communities* (Eds, Werry, C. and Mowbray, M.) Prentice Hall PTR, Upper Saddle River, NJ.
3. Bell, D. (1991) Modes of Exchange: Gift and Commodity, *Journal of Socio-Economics*, 20, 155-167.
4. Bessen, J. (2002) Open Source Software: Free Provision of Complex Public Goods, *Proceedings of the Open Source Software: Economics, Law and Policy Conference*, Toulouse, France.
5. Bourdieu, P. (1997) Marginalia--Some Additional Notes on the Gift, in *The Logic of the Gift: Toward an Ethic of Generosity* (Ed, Schrift, A. D.) Routledge, New York, NY.
6. Carrier, J. (1991) Gifts, Commodities, and Social Relations: A Maussian View of Exchange, *Sociological Forum*, 6, 119-136.
7. Cowen, T. (1993) Public Goods and Externalities, in *The Fortune Encyclopedia of Economics* (Ed, Henderson, D. R.) Warner Books, New York, NY, pp. 74-77.
8. Diker, V. G. (2003) *Toward a Dynamic Theory of Open Online Collaboration Communities*, Ph.D. Dissertation, School of Information Science and Policy, University at Albany-SUNY, Albany, NY.
9. Diker, V. G. and Scholl, H. J. (2001) The Art of Leveraging : How Powerful Nonlinear Feedback Processes Can Restructure Rapidly Growing Technology and Knowledge Industries, *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, Maui, HI.
10. Fogel, K. and Bar, M. (2001) *Open Source Development with CVS*, Coriolis Technology Press, Scottsdale, AZ.
11. Ghosh, R. A. (1998) Cooking Pot Markets: An Economic Model for the Trade in Free Goods and Services on the Internet, 3 (3), Accessed: January 10, 2003, Available: http://www.firstmonday.org/issues/issue3_3/ghosh/index.html.
12. Gregory, C. (1982) *Gifts and Commodities*, Academic Press, London.
13. Hagel, J. and Armstrong, A. (1997) *Net Gain: Expanding Markets through Virtual Communities*, Harvard Business School Press, Boston, MA.
14. Hawkins, R. (2001) The Economics of Free and Open Source Software, *Proceedings of the 7th International Conference of the Society for Computational Economics*, New Haven, CT.
15. Hiltz, S. R. (1986) *Online Communities: A Case Study of the Office of the Future*, Ablex, New York, NY.
16. Kollock, P. (1999) The Economies of Online Cooperation: Gifts and Public Goods in Cyberspace, in *Communities in Cyberspace* (Eds, Smith, M. and Kollock, P.) Routledge, London, pp. 220-239.
17. Kvale, S. (1996) *Interviews: An Introduction to Qualitative Research Interviewing*, Sage, Thousand Oaks, CA.
18. Millen, D. R. (2000) Community Portals and Collective Goods: Conversation Archives as an Information Resource, *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Maui, Hawaii.
19. O'Reilly, T. (1999) Lessons from Open-Source Software Development, *Communications of the ACM*, 42, 33-37.
20. Preece, J. (2000) *Online Communities: Designing Usability, Supporting Sociability*, John Wiley & Sons Inc., New York, NY.
21. Raymond, E. S. (2001) *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, O'Reilly and Associates, Sebastopol, CA.
22. Stanoevska-Slabeva, K. and Schmid, B. F. (2001) A Typology of Online Communities and Community Supporting Platforms, *Proceedings of the 34th Hawaii International Conference on System Sciences*, Maui, HI.
23. Turoff, M. (1997) Virtuality, *Communications of the ACM*, 40, 38-43.
24. Turoff, M. and Hiltz, S. R. (1982) The Electronic Journal: A Progress Report, *Journal of The American Society for Information Science*, 33.
25. Wasko, M. M. and Teigland, R. (2002) The Provision of Online Public Goods: Examining Social Structure in a Network of Practice, *Proceedings of the 23th International Conference on Information Systems*, Barcelona, Spain.