Toward an Analytic Framework for Understanding and Fostering Peer-Support Communities in Using and Evolving Software Products

Andrew Gorman
Center for LifeLong Learning & Design
University of Colorado, Boulder
+1 303.492.1677
agorman@colorado.edu

Gerhard Fischer
Center for LifeLong Learning & Design
University of Colorado, Boulder
+1 303.492.1502
gerhard@colorado.edu

ABSTRACT

The fundamental challenge for social computing is to contribute to fostering communities in which humans can transcend the limitation of the unaided, individual human mind by helping each other. Going beyond antidotal examples requires an analytical framework in which to interpret data in order to understand the context- and application-specific nature of these collaborations. We have studied peer-support communities (PSCs) in the context of the SAP Community Network (SCN), which relies on forums and conferences to support their collaboration.

This research attempts to create a deeper understanding of the effectiveness of social support provided by peers in software development communities from the following perspectives:

1. **Responsiveness**—how responsive are communities to the needs of its members?
2. **Engagement Intensity**—how timely is the peer support?
3. **Role Distribution**—how wide is the participation of users and in what kind of roles do they participate?
4. **Reward System**—what is the impact of explicit reward (point) systems on community behavior?

The data gained from analyzing these perspectives (and their comparison with open source software peer-support communities) has provided insights and led to an increased understanding of what works in PSCs. Here we articulate some initial design guidelines to further improve the potential benefits gained from these communities.

Categories and Subject Descriptors

H.5.3 (**Information Systems**): Group and Organization Interfaces, Collaborative computing, Computer-supported cooperative work; K.4.3 (**Organizational Impacts**): Computer-supported collaborative work

General Terms

Design, Human Factors

Keywords

peer-support communities (PSCs), SAP Community Network (SCN), reward systems, cultures of participation

Copyright is held by the author/owner(s).
C&T'09, June 25–27, 2009, University Park, Pennsylvania, USA.
ACM 978-1-60558-601-4/09/06.

1. INTRODUCTION

To better understand the conceptual foundations of communities and technologies, we have focused on *peer-support communities (PSCs),* specifically those that exist among participants in the SAP Community Network (SCN). SAP (www.sap.com/usa/about and en.wikipedia.org/wiki/SAP_AG) is the third largest software company in the world. It produces enterprise software that provides highly integrated information technology solutions for planning and management related to enterprise resources, customer relationships, product lifecycles, supply chains, and supplier relationships. Achieving this integration and providing external simplicity to the end user comes at the expense of great internal complexity in the software development process.

SCN is supported by a variety of different *technologies*, including discussion forums, blogs, wikis, code samples, training materials, and a technical library (www.sdn.sap.com/irj/scn). In addition to the daily interactions that are supported by the technology are international, national, and regional *user group conferences*. Although our research was focused on an analysis of the SCN forums, we became aware of important interactions among the technical media. One example involves a moderator suggesting that certain forum content is more appropriate for another media, such as a wiki. In a second example, an SAP employee invites an outstanding forum contributor to create a blog in an effort to increase interest in one of SAP's product lines. Other examples include interactions between the technical media and the conference events. We observed that there is typically a spike in forum activity prior to conference events. Important research questions are related to the interactions among these venues, but these questions are outside the scope of this paper.

This paper analyzes the SCN forums as a specific PSC. It describes the ecology of participants and includes a comparison with two open source communities (Apache Commons (commons.apache.org) and Lucene Java (lucene.apache.org)). We develop a conceptual framework, and analyze the data collected over several years. We conclude with design recommendations for reducing the burden on the helper community by (1) increasing the reuse of existing solutions in the forums and (2) encouraging users seeking solutions to shift into the role of providing solutions. These changing usage patterns and roles provide insight into how participation cultures [8] might support dissemination of innovations that might otherwise be lost.

2. PROBLEM AND OPPORTUNITY

Many of the most pressing problems facing the world today are systemic problems transcending the individual human mind [2]. Solving these problems requires social structures that enable groups of people to share knowledge and resources in support of collaborative design, working, and learning. The objective of educating "Renaissance scholars" (such as Leonardo da Vinci, who was equally adept in the arts and the sciences [25]) is not reasonable in today's world. We need to invent, design, and assess computationally enabled and enhanced social organizations that will support *"collective comprehensiveness through overlapping patterns of unique narrowness"* [5] by bringing together people who each know something but do not know other things. Exploiting this "symmetry of ignorance" [9] will move beyond isolated individuals and foster a social environment that exploits the "synergy of many" [4], taking advantage of the "wisdom of crowds" [28].

PSCs need to be understood, conceptualized, and designed as *socio-technical systems* [20, 21, 30] for the following reasons:

- The deep and enduring changes are *not just technological, but social and cultural as well*. Changes in complex environments are not primarily dictated by technology; rather, they are the result of an incremental shift in human behavior and social organization [3, 13].
- PSCs require the *co-design* of social and technical systems; they use models and concepts that not only focus on the artifact but also exploit the social context in which the systems will be used.

PSCs must allow participants to act as information seekers and helpers and must foster relationships and give rewards in order to support all participants [12].

3. CONCEPTUAL FRAMEWORK

Companies and user groups in which large numbers of creative individuals are engaged in coping with the challenge articulated by *"If only X knew what X knows"* (originally articulated as the slogan *"If only HP knew what HP knows"* [26]) try to engage their employees in participation cultures and provide them with the computational environment to do so [16, 17].

PSCs address this challenge by breaking down the barriers and distinctions between information seekers and helpers and open up opportunities for *distributed* [15] and *collective* intelligence [19]. The shift from Web 1.0 (focused on broadcast) to Web 2.0 (focused on participation) [23] has created a great interest as well as an important need and opportunity to create a scientific understanding of participation cultures that transcend the currently existing anecdotal evidence [3].

Web 2.0 environments (characterized by social production, mass communication, democratizing innovation, and collaborative design) represent communities of active contributors and designers and have opened new possibilities for producers/consumers ("prosumers") [29] driven by the evolution of socio-technical environments. The opportunity is to understand how to take advantage of these possibilities to reach new levels of innovation, collaboration, and creativity based on distributed and collective intelligence.

Our empirical investigation of SCN was embedded in our ongoing development of a broad-based conceptual framework for participation cultures characterized by:

- **Re-defining the Roles of Users and Developers:** In the world of software, users and developers are conventionally regarded as two mutually exclusive groups of people. Users own problems, and developers construct software systems for the users. However with the emergence of Web 2.0 environments, the *distinction between users and developers* is quickly disappearing. There is an urgent need to understand how the roles of users and developers are transformed during the lifecycle of particular software products.
- **Co-Evolution of Product and User Community:** For a software project to have a sustainable development through the contributions of users, the system and the user community must co-evolve. The evolution of a community is affected by the contributions made by its aspiring and motivated members. Such contributions not only transform the role and influence of their contributors in the community and thus evolve the whole community, they also are the sources of the evolution of the system. The opposite is true as well: any modification, improvement, and extension made to a system not only evolves the system but redefines the role of the contributing members and thus changes the social dynamics of the community.
- **Socio-Technical Mechanisms:** To facilitate co-evolution, the user community needs socio-technical mechanisms for recognizing user contributions. In the case of peer-support communities, it is particularly interesting to understand how community members migrate from their roles as solution seekers into roles of solution providers who can indirectly influence the direction and evolution of the software system.
- **Supporting Evolutionary Growth through Innovative Local Practices and the Long Tail:** SAP products are highly process-oriented, supporting routine cognitive tasks, and they provide little support for idiosyncratic *local needs* driving innovations in work practices [27]. Rather, this evolution takes place at individual SAP customer sites. Beginning with a core SAP software framework, customers simultaneously adapt the software and their business processes to create a socio-technical system that represents a useful mapping between the two. Over time, local innovations and work-arounds make their way into the socio-technical system through adaptations to software and formal processes. This model of evolutionary growth [10] seen in the SAP software development life cycle presents a unique opportunity to apply Long Tail concepts [1]. In this hybrid model of software usage and development, SAP develops and maintains core processes used by the majority of customers (located in the "head"), and SAP customers create local innovative niche solutions (located in the "tail").

4. APPLICATION CONTEXT: THE SAP COMMUNITY NETWORK

For the purpose of this analysis, we describe a broad SAP ecology that is by no means a comprehensive account but nonetheless provides a useful context for this work. We focus on a subset of the ecosystem and the technologies they use to provide peer support.

The broad SAP ecology includes the following groups:

- **Environment and Platform Developers:** These are internal SAP employees who create and evolve complex software

frameworks for supporting process-oriented work. This group focuses on identifying best practices in business processes and creating a technological framework for helping businesses map their processes into these best practices.

- **Developers Using the SAP Framework:** These are software developers and business process experts who create enterprise software solutions for businesses. This group includes IT developers who are both internal and external to the company that is implementing or has implemented an SAP solution. Internal developers and business process experts know the idiosyncrasies of their particular companies. External developers and business process experts know SAP very well and help companies map their IT needs into the SAP frameworks.
- **End-Users:** These are employees, customers, and suppliers of enterprises using SAP software, including clerks responsible for day-to-day activities, financial workers, and business strategists. They all perform quasi-routine tasks, but also develop innovative local work practices to compensate for inadequacies in the process-oriented software.

The *SAP Community Network (SCN)* is a subset of the larger SAP ecology (for additional background on SAP ecologies, see Hagel and Brown [14]). SCN includes members of the second of these three groups: *developers using the SAP framework.* It is a highly active community of developers, consultants, integrators, and business analysts building and sharing knowledge about SAP technologies. SCN provides an interesting ecosystem and rich test-bed for the study of participation cultures.

5. SCN ANALYSIS

SCN forums provide a valuable context in which to study SAP and how its users collaborate. The SCN forums we analyzed included 162 different forums, grouped into 14 categories. In our analysis, we observed interesting behavior that shows a sufficiently mature community, one that exhibits comfortable expression by its members, development of social norms, and self-policing. The forums are used primarily for asking and answering questions, but they also convey a sense of community based on collaborative problem solving. For example, the SCN forum is used as a place to "think-out-loud":

Figure 1 shows an example of a user who posts a question and three hours later posts a solution to his own question, thus sharing the benefits of his investigation with the community. In this case the forum embodies a sense of an extended workgroup that transcends the corporate organizational boundaries of the developer.

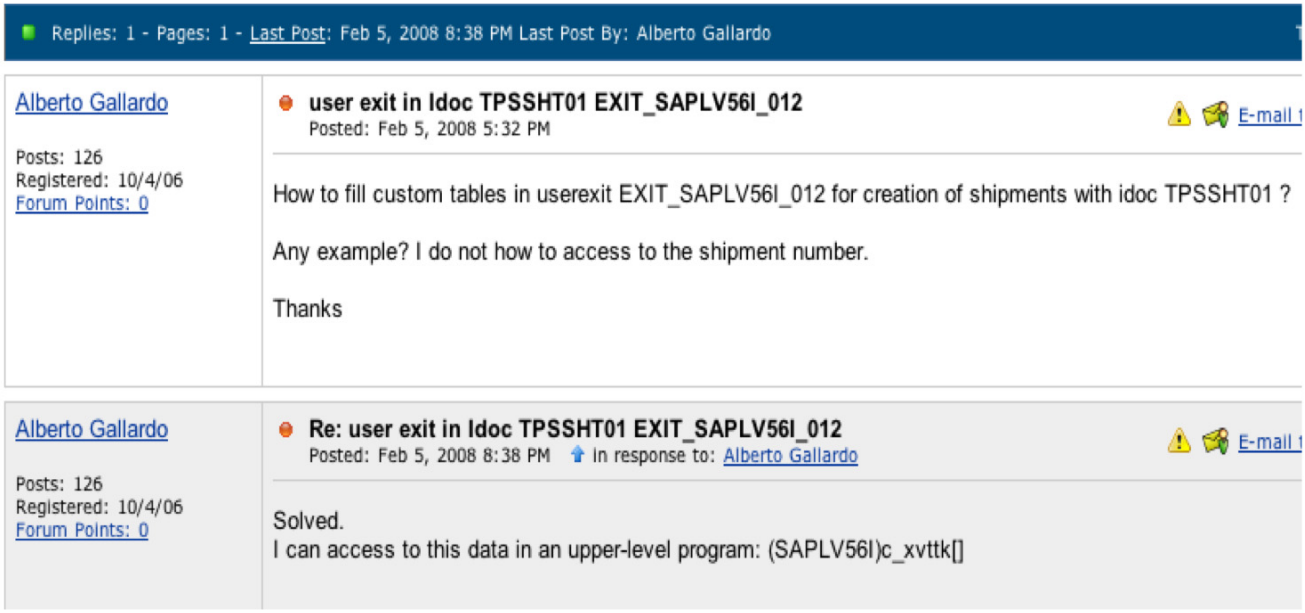

Figure 1 Using SCN to "think-out-loud"

Last Post: May 16, 2006 7:46 AM Last Post By: Maran N

Cheating in BI Forums for Points
Posted: May 15, 2006 6:31 AM

E-mail this message Reply

Hello Friends,

Today I saw a group of users(maybe from the same company) who are copying previously answered threads just to get points.

They are following a system like.."i copy a query from a previously answered thread..post it as a new query..u answer the query by copying answers from the old thread..and i say 'thanks' and award points!".

these 4-5 users have all joined SDN on 14th May..and have already collected good number of points.

the very minute(say 11.39 am) one of them posts a query..3-4 of them post a reply(at 11.39 am)..and that very minute(11.39 am!) the thread is closed!

the same question is posted more than 3-4 times.

to prove my point..see below..

Figure 2: SCN members exhibiting self-policing behavior

Another example shows "cheater detection," which is an important attribute of a sustainable community [7, 24]. Figure 2 shows evidence of the development of social norms and self-regulation. The figure also highlights a contentious component of the SCN forums' motivation structure: for some users, rewards based on awarding points have become a competition that has both positive and negative impacts on the community.

Both of these examples show that the motivation to participate goes beyond the simple need to find technical solutions. There is a greater sense of community. In the first example, the user could have solved his own problem and moved on, but instead he posted his solution, presumably influenced by such a greater sense of community. In the second example, participation is clearly not motivated by a need to find answers to a technical problem. This member is motivated by maintaining justice within the point system.

The Framework Underlying Our Data Analysis. Developers and users of SAP software have to cope with a complex environment (e.g., they need to learn and apply complex application program interfaces (APIs) and quickly overcome obstacles to perform their jobs). To achieve their goals, they rely heavily on *PSCs*. Our empirical investigations focused on the following perspectives:

- **Responsiveness:** *How responsive are communities to the needs of its members?* (Our focus: measuring the response rate at which peers answer questions and the time it takes to get an initial response.)
- **Engagement Intensity:** *How timely is the peer support?* (Our focus: looking at the length of time it takes for a question to be answered, the number of messages needed, and the number of members involved.)
- **Role Distribution:** *How wide is the participation of users and in what kind of roles do they participate?* (Our focus: examining the distribution of the two major roles: those who ask questions and those who answer questions.)
- **Reward System:** *What is the impact of explicit reward (point) systems on community behavior?* (Our focus: examine the relationships among the three aforementioned perspectives.)

These perspectives were originally utilized in an analysis of open source software (OSS) systems [32], whose participants also form a PSC. Although this does not represent a fully comprehensive framework for understanding communities, it provides an initial framework for quantifying peer-support efficacy.

The two OSS systems previously analyzed were:

- **Apache Commons** (http://commons.apache.org) is an Apache project that creates and maintains a library of reusable Java components that provides common functionality to other Apache projects as well as other projects.
- **Lucene Java** (http://lucene.apache.org) is a project that creates and maintains a high-performance and full-text search engine library written in Java. The library has been widely reused by a number of high-profile projects, including the Eclipse IDE and Wikipedia.

For SAP, we studied the SCN forums, which are the primary means of peer support. Our data contain threads for a 5-year period from June 12, 2003 (the forum's inception), through May 6, 2008, when the data collection was performed. For Apache Commons and Lucene Java, we analyzed dedicated mailing lists, which contain data from the very beginning of their projects. The analysis is based on the archives of the two user mailing lists from January 1, 2003, to November 30, 2007, and is included to provide some broad context for the SAP data.

Table 1 shows the scope of the data collection in number of members, discussion threads, and average messages per thread for the three groups. The number of members counts only members who posted at least one message. It does not include forum *lurkers* [22], who never become registered SCN users nor does it include users who register in order to receive other benefits such as downloads or email announcements, but do not post to the forums.

As the data show, the three groups have similar distributions of messages to threads, but the number of members and contributions are two orders of magnitude larger in SCN. Furthermore, the OSS groups use mailing list technologies, whereas SCN uses feature-rich online forums. These factors affect the value of making direct comparisons between the SCN and OSS groups, but a comparison is nonetheless useful for providing context.

Table 1: Number of messages, members, messages threads, and average messages per thread

	Number of members	Number of messages	Number of threads	Average messages per thread
SCN	**120709**	**2954520**	**683901**	**4.32**
Commons	3347	18516	5648	3.28
Lucene	2586	27742	6538	4.24

6. RESULTS

For the sake of this analysis, we make an assumption that threads in these datasets represent technical questions from community members and that replies from other members can be characterized as attempts to provide answers. To validate the assumption that most forum threads represent questions, we performed a qualitative analysis of a random sample of 600 threads from the SCN dataset. The data from this sample suggest that the proportion of questions in these threads is very high; a 95% confidence interval for the proportion runs from .982 to 1.0.

The questions typically ask for:

- solutions to specific code bugs;
- approaches for addressing functional requirements;
- references to documentation on a specific technical topic;
- explanations of specific functionality; and
- opinions about one technical approach versus another.

The following sections describe our results from the four perspectives described above: *responsiveness, engagement intensity, role distribution,* and *reward system*.

6.1 Responsiveness

Table 2 presents a general sense of *responsiveness* as the percentage of threads that get a response from other members. As shown in the table, SCN has an 86% response rate, meaning that 86% of the threads received a reply from at least one other member. Commons and Lucene have respective *response rates* of 67% and 80%. These rates indicate that a very high percentage of askers got help, or at least attempted help, from their peers in the online peer support community. All three communities have *response rates* that are higher than the 61% reported by von Krogh et al. in their study of peer support among users of the Apache Web Server [31].

Table 2: Response Rate

	Total number of threads	Number of "no response" threads		Number of threads with responses	
SCN	**683901**	**92664**	**14%**	**591237**	**86%**
Commons	5648	1870	33%	3778	67%
Lucene	6538	1302	20%	5206	80%

6.2 Engagement Intensity

To understand how quickly members receive responses, we looked at the difference between the time the thread was created and the time that the first response was posted by a member other than the thread originator. Only threads that received a response are included (N = 591237 or 86% of all threads shown in Table 2).

Table 3 presents the *response time* for the three groups: the first quartile time (Q1) at which 25% of the questions got the first response; the second quartile time (Q2) or the median time at which 50% of the questions got the first response; and the third quartile time (Q3) at which 75% of the questions got the first response. The median *response time* in (Table 3) indicates that half of all SCN threads got a response within 23 minutes. This is in contrast to Commons and Lucene, which have median response times of 3 hours, 56 minutes and 1 hour, 27 minutes, respectively.

Table 4 shows the *discussion duration* for the three groups: the first quartile discussion time (Q1) at which 25% of the threads were completed; the second quartile time (Q2), or the median time at which 50% threads were completed; and the third quartile (Q3) time at which 75% of the threads were completed. As can be seen, 25% of the SCN threads require 46 minutes or less to complete, and 50% are completed in 5 hours, 30 minutes or less. These are less than the *discussion durations* for the other two groups.

Table 3 and Table 4 show that SCN requires less time in terms of getting an initial response and also overall duration of the thread.

Table 3: Response Time

Response Time	Q1 (25%)	Q2 (Median)	Q3 (75%)
SCN	**6 m**	**23 m**	**3 h 10 m**
Commons	49 m	3 h 56 m	14 h 15 m
Lucene	24 m	1 h 27 m	5 h 51 m

Table 4: Discussion Duration

Discussion Duration	Q1 (25%)	Q2 (Median)	Q3 (75%)
SCN	**46 m**	**5 h 30 m**	**2 d 3 h 38 m**
Commons	3 h 02 m	13 h 37 m	2 d 0 h 49 m
Lucene	2 h 59 m	9 h 28 m	32 h 53 m

Effect of group size on response time. An initial hypothesis for explaining fast response times in SCN was that a greater number of group members would decrease response time. There is a high negative correlation (−.76) between median response time (Table 3), and number of members (Table 1), suggesting that more users decrease the time it takes to get a response. However, an analysis of variance (ANOVA) shows *no significant effect of group size on response rate*. Additionally, due to the small sample size of N=3 groups, we tested the hypothesis within SCN *between all forums* (N=162). An ANOVA between SCN forums shows *no significant effect* of the *number of members who posted to a forum* and the *mean response time* for that forum. However, a second look at response time shows a more complicated picture. Looking at group size and response time *for all forums together* over time, group size has a significant effect on decreasing response time. This implies differences between the forums that are not seen in the total population. Furthermore, looking at the data at a finer grain shows ***a significant effect in the first few years*** (months 1 – 29) and ***no significant effect in the last few years*** (months 20 – 59). This complex phenomenon suggests patterns of community evolution that are not yet well understood.

6.3 Role Distribution

To explore the evolution of the community, we looked at the distribution of roles over time [11]. We grouped users into two groups: those who ask questions (post threads) and those who answer questions (reply to threads). There is a dramatic change in role distribution over time that is not associated with the point system. As seen in Table 5, the percentage of users who reply to threads has significantly decreased over time, and this has a significant effect on response time. This is likely to be the result of a core group of helpers coalescing over time. Although a smaller percentage of helpers are required to support community, each helper is relied on more heavily.

Table 5: Change in Role Distribution Over Time

Year	Asker Count	Helper Count	Percentage Helpers
June-03–May-04	2963	3781	62%
June-04–May-05	12584	13665	53%
June-05–May-06	21875	18977	52%
June-06–May-07	99472	73315	43%
June-07–May-08	133648	75205	33%
Totals	**270538**	**184936**	**41%**

Table 6: Frequency Distribution of Threads by Helper Count

Helper Count	Number of Thread	Relative Frequency	Cumulative Relative Frequency
0	92752	13.56%	13.56%
1	213314	31.19%	44.75%
2	169879	24.84%	69.58%
3	101783	14.88%	84.46%
4	53884	7.88%	92.34%
5	26822	3.92%	96.26%
6	13047	1.91%	98.17%
7	6258	0.91%	99.09%
8	3091	0.45%	99.54%
9	1440	0.21%	99.75%
10	810	0.12%	99.87%
11	457	0.07%	99.93%
12	247	0.04%	99.97%
13	131	0.02%	99.99%
14	74	0.01%	100.00%

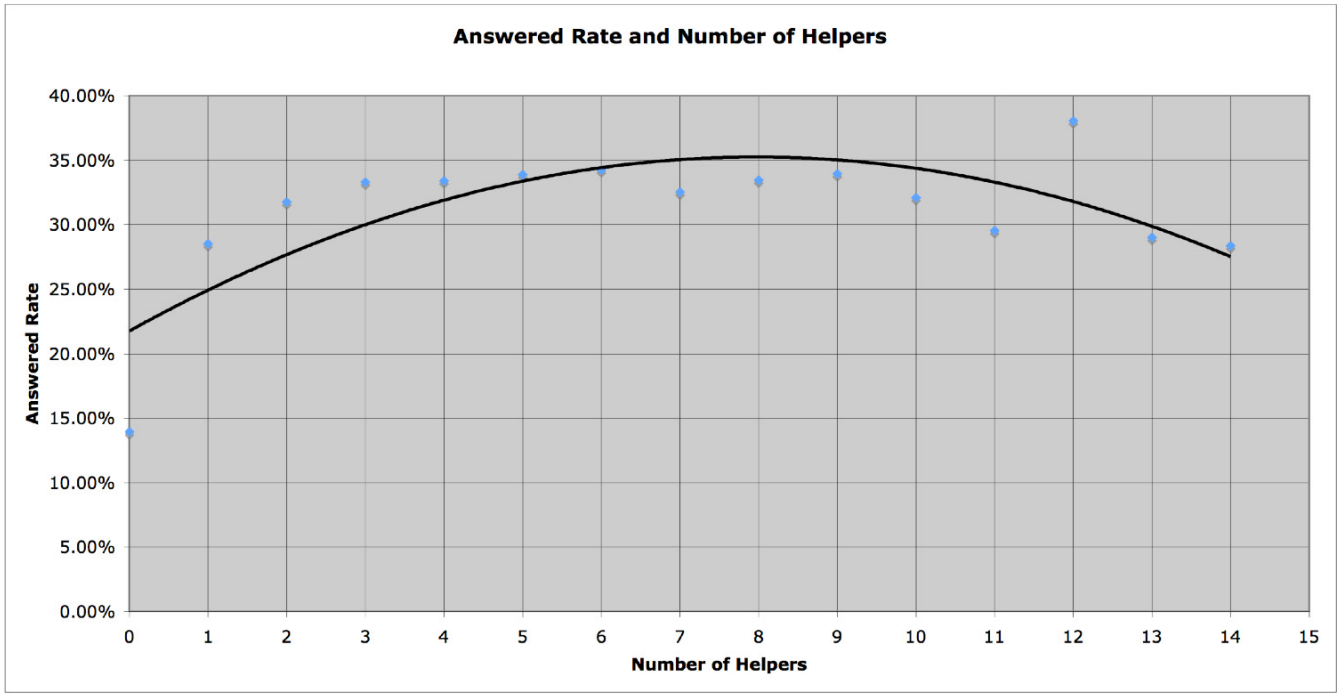

Figure 3: Percentage of Answered Threads by Helper Count

Effect of thread helper count on answer rate. Although an increase in the number of helpers participating in a thread can be accompanied by an increase in the likelihood that the question will be marked as answered, Figure 3 shows that after a point this likelihood actually decreases (e.g., of the threads with 6 helpers, 35% were marked as answered compared to 30% of threads with 11 helpers). This suggests a "too many cooks in the kitchen" effect. It is possible that the thread becomes "hijacked" into discursive discussion rather than focused on answering the question.

As shown in Table 6, only 0.25% (1 - .9975) of the threads have *greater than* 9 helpers. This is about the point where answered rates decline (Figure 3). It is possible that these threads represent complex questions with no clear answer, which could shed light on some of the tough problems being experienced by members. These could also indicate discussions where members express their opinions rather than attempt to solve an immediate problem of one of the community members. A qualitative analysis of a sample of these threads is required to better understand this phenomenon.

6.4 Reward System

In August 2004, SAP implemented a point system in which members who post threads can award points to members who help the most. This was an attempt to motivate members to contribute. We hypothesize that the point system has a significant impact on the knowledge-exchange behavior: members would be more eager and willing to answer questions. To test this hypothesis, a t-test was performed on the SCN forums. Pre- and post-groups were created by selecting all forum threads (that received a response) from the three months before and after the introduction of the point system. A Welch Two Sample t-test showed a significant difference in mean *response time*: it took less time on average to receive an initial reply. t-tests were also performed for *helper count* (number of users who contributed to a thread), *percentage answered* (whether the thread was marked as answered by the user who initiated the question), and *discussion duration* (see Table 7). Significant differences were observed for *helper count and percentage answered*: there were a greater number of helpers and percentage of answered threads after the point system was introduced. There was no significant change in *discussion duration*, which is important because it shows that users experienced better results without requiring more time.

Table 7: Welch Two Sample t-test Comparing Three Months Before and After the Point System Introduction

Group	Before point system (N=3766)	After point system (N=5963)
Mean Response Time ***	51 (min.)	34 (min.)
Mean Helper Count ***	1.892378	2.019339
Percentage Answered ***	12%	30%
Mean Duration	162 (min.)	149 (min.)
***** p-value < 0.01**		

High stakes point system. By scanning the *content* of SCN threads for topics related to the point system, it is clear that the point system is a very strong motivator. The SCN website displays individual and company high point scorers for a rolling three-month period. We speculate that some SAP developer consulting firms link point awards with job performance measures. Consulting firms can use this distinction to attract customers in a competitive market. The high stakes in the point system are seen in the following types of examples:

1. helpers remind askers to reward points;
2. helpers complain if no points are awarded;
3. users cheat to gain points; and
4. other users detect and report cheaters.

Faster doesn't always mean better. While SCN has faster response rates since the point system was introduced, it also requires more members to answer the questions. From a productivity standpoint, this can have drawbacks. There appears to be a "race" condition in a competition to score points. Helpers may be less motivated by the desire to help someone solve a problem. Instead, their motivation may be to score points. In this case, the motivation to help solve a problem and the motivation to reply to a question could be two different things.

7. CONCLUSION

The analysis shows that the point system that was in introduced in August 2004 did not cause an increase in contributors, but did improve response time and the percentage of threads that were marked as "answered." It also revealed a changing distribution of user roles as the community evolved. There was a significant decrease in percentage of users who replied to threads compared to those who only initiated. This shows a core user group emerging as the community matures, but it also highlights the fact that there is an increasing burden on each helper. These findings resulted in design recommendations for reducing the burden on the helper community by (1) increasing the reuse of existing solutions in the forums and (2) encouraging users to shift into helper roles. There may be many ways to approach these goals; here we identify and propose the following design changes.

Turn questions into search queries. A simple approach that could reduce the burden on the core helper community is to add a small step to the question submission process. Often helpers complain that askers have not "done their homework" by searching the forum before asking a question. This is seen in responses that direct askers to other threads that have already resolved the question. Coincidently, asking questions that have already been asked is also a method for cheating. Turning questions into search queries could also help automatically detect cheaters, similar to techniques used to detect plagiarism in academic settings.

Provide links between related threads. Another approach, which is a variation of turning questions into queries, is to provide links to related threads. Thread pages can simply provide links to the most closely related threads (by using a technique such as latent semantic analysis [6, 18]). This has the advantage of providing two benefits: It benefits the asker who posts a question and sees existing related threads, and it benefits the searcher who finds a partial answer to a question by searching the forum but also finds other threads that are more closely related. This can be seen as a form of refining a query: the user performs a search, selects a search result that seems to fit the question, and

automatically sees additional related links that could be different than those showing up in the original search.

Push questions to willing helpers. Table 3 and Table 4 show that while 75% of threads have a response time of less than 3 hours, 10 minutes, 25% of the threads also have a duration of more than 2 days. This suggests that initial responses do not always adequately answer the question. By sending questions to select users, it might be possible to increase response effectiveness. It can also increase role distribution (the ratio of the number of members who ask questions to those who answer questions). One approach could be to send questions to (1) users who have *answered similar questions*, thus increasing effectiveness, and (2) users who have *had similar questions answered for them*, thus increasing role distribution by encouraging askers to become helpers.

8. ACKNOWLEDGMENTS

This research was sponsored by SAP Research and was conducted in collaboration with SAP Labs in Palo Alto, California, U.S. We especially thank Joerg Beringer for his intellectual contributions as well as Paul Hofmann and Ike Nassi for their support. We also thank Yunwen Ye from the SRA Technology Strategy Division, Tokyo, Japan, whose work on open source software communities provided a basis for our analytical framework.

9. REFERENCES

1. Anderson, C. *The Long Tail: Why the Future of Business is Selling Less of More*. Hyperion, New York, NY, 2006.
2. Arias, E.G., Eden, H., Fischer, G., Gorman, A. and Scharff, E. Transcending the Individual Human Mind—Creating Shared Understanding through Collaborative Design. in Carroll, J.M. ed. *Human-Computer Interaction in the New Millennium*, ACM Press, New York, 2001, 347-372.
3. Benkler, Y. *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. Yale University Press, New Haven, 2006.
4. Bennis, W. and Biederman, P.W. *Organizing Genius: The Secrets of Creative Collaboration*. Perseus Books, Cambridge, MA, 1997.
5. Campbell, D.T. Ethnocentrism of Disciplines and the Fish-Scale Model of Omniscience. in Derry, S.J., Schunn, C.D. and Gernsbacher, M.A. eds. *Interdisciplinary Collaboration — An Emerging Cognitive Science*, Lawrence Erlbaum, Mahwah, NJ, 2005, 3-21.
6. Deerwester, S., Dumais, S., Furnas, G., Landauer, T. and Harshman, R. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, *41* (6). 391-407.
7. Dunbar, R.I.M., Knight, C. and Power, C. *The evolution of culture : an interdisciplinary view*. Rutgers University Press, New Brunswick, N.J., 1999.
8. Fischer, G. End-User Development and Meta-Design: Foundations for Cultures of Participation. in Pipek, V., Rossen, M.B., deRuyter, B. and Wulf, V. eds. *End-User Development*, Springer, Heidelberg, 2009, 3-14.
9. Fischer, G. Social Creativity, Symmetry of Ignorance and Meta-Design. *Knowledge-Based Systems Journal (Special Issue on Creativity & Cognition), Elsevier Science B.V., Oxford, UK*, *13* (7-8). 527-537.
10. Fischer, G., Grudin, J., McCall, R., Ostwald, J., Redmiles, D., Reeves, B. and Shipman, F. Seeding, Evolutionary Growth and Reseeding: The Incremental Development of Collaborative Design Environments. in Olson, G.M., Malone, T.W. and Smith, J.B. eds. *Coordination Theory and Collaboration Technology*, Lawrence Erlbaum Associates, Mahwah, NJ, 2001, 447-472.
11. Fischer, G., Piccinno, A. and Ye, Y. The Ecology of Participants in Co-Evolving Socio-Technical Environments. in Forbrig, P., Paternò, F. ed. *Engineering Interactive Systems (Proceedings of 2nd Conference on Human-Centered Software Engineering)*, Springer, Heidelberg, 2008, 279-286.
12. Fischer, G., Scharff, E. and Ye, Y. Fostering Social Creativity by Increasing Social Capital. in Huysman, M. and Wulf, V. eds. *Social Capital and Information Technology*, MIT Press, Cambridge, MA, 2004, 355-399.
13. Florida, R. *The Rise of the Creative Class and How It's Transforming Work, Leisure, Community and Everyday Life*. Basic Books, New York, NY, 2002.
14. Hagel, J., and Seely Brown, J. Innovation on the Edge: How SAP Seeds Innovation, *Business Week*, 2008. URL= http://www.businessweek.com/print/innovate/content/jul2008/id20080723_353753.htm
15. Hollan, J., Hutchins, E. and Kirsch, D. Distributed Cognition: Toward a New Foundation for Human-Computer Interaction Research. in Carroll, J.M. ed. *Human-Computer Interaction in the New Millennium*, ACM Press, New York, 2001, 75-94.
16. IBM. Project Beehive, 2008. URL= http://domino.research.ibm.com/cambridge/research.nsf/99751d8eb5a20c1f852568db004efc90/8b6d4cd68fc12b52852573d1005cc0fc?OpenDocument
17. Kellogg, W.A. Supporting Collaboration in Distributed Teams: Implications for e-Research (Contribution to the ECSCW Workshop "Realising and Supporting Collaboration in e-Research"), 2007. URL= http://www.e-researchcommunity.org/docs/ecscw07/submissions/Kellogg.pdf
18. Landauer, T.K., McNamara, D., Dennis, S. and Kintsch, W. (eds.). *Latent Semantic Analysis.* Erlbaum, Mahwah, NJ, 2007.
19. Malone, T.W. What is collective intelligence and what will we do about it? in Tovey, M. ed. *Collective Intelligence: Creating a Prosperous World at Peace*, Earth Intelligence Network, Oakton, Virginia, 2008, 1-4.
20. Mumford, E. A Socio-technical Approach to Systems Design *Requirements Engineering,*, 2000, *5*(2), 59-77.
21. Mumford, E. Sociotechnical Systems Design: Evolving Theory and Practice. in Bjerknes, G., Ehn, P. and Kyng, M. eds. *Computers and Democracy*, Avebury, Aldershot, UK, 1987, 59-76.
22. Nonnecke, B. and Preece, J. Lurker Demographics: Counting the Silent. in *Proceedings of CHI'2000*, ACM Press, The Hague, 2000, 73-80.
23. O'Reilly, T. What Is Web 2.0—Design Patterns and Business Models for the Next Generation of Software, 2006. URL=

http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

24. Pinker, S. *How the Mind Works*. W. W. Norton and Company, New York, 1997.
25. Shneiderman, B. *Leonardo's Laptop — Human Needs and the New Computing Technologies*. MIT Press, Cambridge, Mass, 2002.
26. Sieloff, C.G. 'If only HP knew what HP knows': the roots of knowledge management at Hewlett-Packard. *Knowledge Management*, *3* (1). 47-53.
27. Suchman, L.A. *Plans and Situated Actions*. Cambridge University Press, Cambridge, UK, 1987.
28. Surowiecki, J. *The Wisdom of Crowds*. Anchor Books, New York, 2005.
29. Tapscott, D. and Williams, A.D. *Wikinomics: How Mass Collaboration Changes Everything*. Portofolio, Penguin Group, New York, NY, 2006.
30. Trist, E.L. The Sociotechnical Perspective: The Evolution of Sociotechnical Systems as a Conceptual Framework and as an Action Research Program. in VanDeVen, A.H. and Joyce, W.F. eds. *Perspectives on Organization Design and Behavior*, Wiley, New York, NY, 1981.
31. von Krogh, G., Spaeth, S. and Lakhani, K.R. Community, Joining, and Specialization in Open Source Software Innovation: A Case Study. *Research Policy*, *32* (7). 1217-1241.
32. Ye, Y. and Gorman, A. Measuring Peer Support Activities in Reusing Open Source Software Libraries, forthcoming.