

The Diffusion of ReachOut: Analysis and Framework for the Successful Diffusion of Collaboration Technologies

Vladimir Soroka, Michal Jacovi
IBM Haifa Research Lab
Haifa – 31905, Israel
+972-4-8296311
vladi@il.ibm.com

ABSTRACT

While virtual communities become more and more dominant, little attention has been directed towards understanding the conditions for creating a successful community. Significant progress has been made in understanding the diffusion of collaborative tools in the workplace. We read stories about the extraordinary success of some communities, and about the harsh failure of others. This paper argues that lessons learnt from these stories should be analyzed using the theoretical foundations of Diffusion of Innovations theories, and systematized to create a set of guidelines for community creators to make their efforts more efficient. We begin by presenting a theoretical background for analyzing technology diffusion. We then analyze the stories of diffusion of ReachOut – a tool for peer support and community building developed in our Research Lab – in two different communities, using this theory. Finally, we propose a framework for planning for successful diffusion of collaborative tools, using our experiences with ReachOut.

Categories and Subject Descriptors

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

General Terms

Measurement, Design, Experimentation, Human Factors, Theory.

Keywords

Peer support, community building, diffusion of innovations.

1. INTRODUCTION

Virtual Communities are a powerful concept. In their first incarnations virtual communities helped people deal with everyday problems more effectively or fight for a common goal [28]. While the early communities were non-commercial or even anti-commercial, their business value is being recognized more and more [10]. Many workplaces use collaboration tools, such as forums or instant messaging, to make their employees more effective [4, 13, 29]. It is more or less clear now that a successful virtual community provides numerous benefits to its participants and to their organization (in the case of a business community).

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.

CSCW'04, November 6–10, 2004, Chicago, Illinois, USA.
Copyright 2004 ACM 1-58113-810-5/04/0011...\$5.00.

The challenge of creating successful communities becomes more and more relevant to Chief Information Officers, especially in large enterprises.

The task of creating a successful virtual community is not as trivial as it may seem. Even provided the right technology, many communities fail to “stick”, and participation in them steadily declines or does not start at all. There can be many reasons for the success or failure of virtual communities. Possible reasons include psychological aspects like individual reluctance to actively participate in virtual communities [23, 24]; sociological aspects like negative dynamics of the group of users [36]; technological aspects like an immature technology that makes participating in the community difficult [3]; and various combinations of these. It would be extremely helpful to systematize the reasons for success and failure of virtual communities, and to provide virtual communities' administrators with tools for successful community building.

One of the ways to look at successful virtual community creation is to analyze it as a successful dissemination of both the technology supporting the virtual community and the concept of the specific virtual community. This way of analysis allows us to use various theories of Diffusion of Innovations research [30]. These theories provide explanations using various techniques, such as the context of diffusion, the social networks involved in the diffusion and the qualities of the product itself [7]. While it is not always possible to determine all the factors that lead to a success or a failure, using the above theories can at least provide a range of possible causes.

Though different aspects of the above theories are not always easy to control, they give the community builder an opportunity to use as many of them as possible to achieve success. While the majority of the work in the field of diffusion of innovations focuses on **understanding** the diffusion process in retrospective, not enough work has been done towards drawing a systematized approach to planning for successful collaboration technology diffusion. For example, Sonnenwald et al. [33, 34] draw an experimental and survey based framework for analyzing innovation properties before trying to disseminate it for large audiences. While it is an excellent start, a broader approach is required. In this paper we try to give a first sketch of such an approach, based on the theory and available body of research work, and our personal experiences in disseminating collaboration technology.

Our group had two experiences building virtual communities within IBM. We created these communities using ReachOut, the tool for peer support and community building, developed in our Research Lab [29]. We observed successes and failures during a period of over two years in one community, and slightly over a

year in the other. We did not use a systematic approach for disseminating ReachOut, but we kept a journal describing various actions that were taken and events that took place during ReachOut dissemination. We also have a full log of activities that occurred in the tool itself during the described period. Using this information we can analyze the different aspects of technology and virtual community diffusion, to draw several lessons learnt from the process.

The rest of this paper is organized as follows. The next section provides a description of the related theoretical and practical work in the area of community building and technology diffusion. We then briefly describe ReachOut. We continue with the stories of the two communities in the light of the related research. Finally we conclude with the lessons we learnt and a proposed framework for planning a successful diffusion of collaborative tools.

2. RELATED WORK

The problem of dissemination of new tools in organizations is not new. Zaltman, Duncan and Holbeck [37] in their book "Innovation in Organizations" (1973) pointed out how important it is to study the process of a single innovation diffusion in an organization to understand the diffusion dynamics. Factors like the structure of an organization and its size have been identified as major predictors of its innovativeness [30].

Modern communication technologies (such as email) introduced new aspects into the diffusion research. For one, most of these technologies suffer a critical mass problem [19]. A single user technology doesn't necessarily require widespread adoption in the first implementation stages; however, technologies like fax or email bring little or no benefit to their early adopters, until a significant number of users start using the technology. Then, there is the cost/benefit question. Grudin [9] points out that users who need to invest the most work into adopting the new tool, often benefit the least from the tool itself and provides an electronic calendar as an example to this phenomenon. Moreover, when facing a need for extra work for unclear reasons, people tend to oppose the new technology even more [31]. Even when users do start using the new technology, they may often adopt it in ways convenient to them which are not necessarily the intended or the most efficient for the organization as a whole [26]. Finally, the lack of common goals or common ground can lead to users' unwillingness to collaborate which in turn causes collaborative applications to fail [25].

The problem of implementation of groupware technology can also be seen from a different perspective. For example Mark, and Poltrok [17] looked at the diffusion of Net.Meeting™ in Boeing in the light of the Social Worlds theory. Social Worlds are defined as "group[s] with shared commitments to certain activities" (p.285) and is proposed to better describe the fluid boundaries of distributed organizations. Authors state that this theory is superior to the Diffusion of Innovations theory (DOI) for studying innovation diffusion in a distributed organization, because the latter has many assumptions that are not true for networked systems where the decision to adopt is interdependent among many users. Gallivan [6] also describes a similar problem and offers a combined framework for analyzing organizational diffusion, using both organizational and individual level factors that explain the adoption decisions. This framework incorporates various factors of such complex adoption networks. Gallivan uses

three stages – from managerial decision to adopt, to secondary adoption when organization gradually adopts the technology and to the organizational consequences.

The direction we follow is similar but slightly different than that of the above framework. As CSCW application developers we often face the problem of disseminating our own technologies. However, we rarely use the frameworks suggested to evaluate the diffusion of collaboration technologies to **plan** for a successful diffusion. In this paper we aim at creating a framework, based on previous research, including the Diffusion of Innovation theory, and our own experiences to be used for planning for successful innovation diffusion

Diffusion of Innovations theory uses a wide variety of social science tools to study the process of diffusion of new products and technologies and is excellently summarized by Everett Rogers in his book "Diffusion of Innovations" [30]. The first and most famous studies that used this theoretical framework were the Iowa hybrid corn adoption study [32] and Coleman, Katz and Menzel's study of the diffusion of tetracycline [2]. The theory examines the individual level (types of technology adopters); the organizational level (innovation in organizations); the innovation property level (which properties innovation should have to succeed) and the network level (diffusion networks). We briefly describe some of the concepts relevant to our research in the next sub-sections.

2.1 Innovation in Organizations

Before going into the details of the actual process of diffusing a new technology in an organization, we would like to examine what the main stages of such a process are. Rogers [30] points out two main stages of organizational innovation process – *initiation* and *implementation*.

Initiation is divided into *agenda-setting* and *matching*. Agenda-setting is a process in which a specific problem is identified and a need to solve it is raised. Matching is the process in which a specific innovation is matched to solve the problem from the previous step. Implementation is divided into three stages: *redefining*, *clarifying*, and *routinizing*. Redefining deals with initial implementation of the innovation and redefining it to fit the specific organization. Clarifying is a process where the innovation is widely used and becomes clear to more and more members of the organization. Finally, routinizing occurs when the innovation stops being considered as such – that is it loses its identity as a new technology and becomes a part of the everyday routine of the organization.

It is important to realize the effects of these stages and plan for them in advance, otherwise problems pointed out by Grudin [9], Rogers [31] and others will inevitably occur. Failing to set up an agenda in the organization and match the innovation to it will lead to lack of understanding of the importance of the innovation and eventual unwillingness of potential users to perform actions necessary for innovation success. Without closely monitoring the actual implementation stages, the innovation can easily go off the planned path.

2.2 Property of the Innovation

Before starting the diffusion process, one needs to verify that the technology itself is "sticky" [7]. By "sticky" we mean the ability of the tool to attract the users and make it easy for them to stay on

the tool. Some examples of the lack of stickiness are annoying software bugs, unintuitive features and unreasonable requirements from the users. This is important both from the technology point of view, since finding bugs is the base of the evaluation process for CSCW tools [3] and from the social point of view, since annoying bugs will drive users away. It is also important, of course, that users understand which problem the new tool intends to solve or which process it can enhance, so they can clearly evaluate the cost/benefit of the tool.

Accordingly, Rogers [30] defines five properties a successful innovation technology has to possess:

- *Relative advantage*: the degree to which the innovation is perceived as being better than the practice it supersedes. The promotion of the innovation should play a major role in explaining the benefits of the innovation and affecting potential users' perceptions about it.
- *Compatibility*: the extent to which adopting the innovation is compatible with what people do. It is closely bound to matching and redefining stages in the organizational diffusion process – the tool needs to be changed to match the organizations' policies and practices.
- *Complexity*: the degree to which an innovation is perceived as relatively difficult to understand and use. Complex and unintuitive user interfaces, annoying software problems and other technological factors might become the main obstacle for technology's success [25].
- *Trialability*: the degree to which an innovation technology may be experimented with on a limited basis before making an adoption (or rejecting) decision. Mark, Pollock and Fisher [18] note that for collaboration technologies in distributed environments trialability is very important, since those technologies are more complex than traditional ones and people need a way to try them out or see someone else using them before actively engaging in them.
- *Observability*: the degree to which the results of an innovation are visible to others. Sometimes this measure is also operationalized as "results demonstrability" [34, 21] – that is how easy it is to demonstrate the benefit of the technology. This is always a tricky part in the collaboration technology case, since it is not always easy to convince people why collaboration is good, especially in the organizations that do not have a supporting culture [26].

Sonnenwald, Whitton and Maglaughlin [33, 34] have drawn a survey approach for analyzing the collaboration technology properties described above. This approach taken in the early stages of innovation implementation can help community builders to understand the drawbacks of the technology and fine tune it for the better diffusion.

2.3 Diffusion Networks

The concept of Diffusion networks can be traced back to the 1940s, to the most influential work of Lazarsfeld, Berelson and Gaudet [16]. Beforehand, the common belief was that communication works as a hypodermic needle – straight to the heart of each member of the audience. In their study, Lazarsfeld et al. found that the information about elections flows in two steps. Apparently, the media is only followed closely by those who are specifically interested in politics. These people, called “opinion

leaders”, become the source of information for their peers. This finding was one of the first leading to the understanding that knowledge flows over social networks.

The concept of networks raised further interest later when sociologists and computer scientists started to study the networked nature of information flow. The famous study about Small Worlds, conducted by Milgram [20], is yet another proof of the power of networks. Milgram showed that it took less than six steps on average for a package sent from a group of randomly selected people residing in different parts of the USA to reach a merchant in Boston. Granovetter [8] has shown that weak ties (casual acquaintances, rather than close friends) are very important for disseminating information in social networks, particularly when searching for a job.

In his excellent book "The Tipping Point" [7], Gladwell describes three kinds of people who play important roles in the networked diffusion: a) *connectors* – who have many connections in different worlds (Mark and Pollock [17] also describe their importance in the diffusion of communication technology); b) *mavens* – people who have a deep understanding in a subject matter (they probably are identical to opinion leaders, described above); c) *salesmen* – people who can "sell" the innovation to other people and convince them to use it.

Understanding the networked nature of the diffusion process is very important in evaluating the process of implementation of a communication technology and will be an important part of our framework. Social networks in the organization [27] should be carefully studied and the correct people should be chosen as the champions of the new technology.

2.4 The Power of Context

Analyzing the context in which the innovation spreads is also very important. Community features, organizational culture, work habits and even the structure of the building where an organization resides can have a vital influence on the success or failure of the new technology. In general, people are very sensitive to their environment; Zimbardo and his colleagues [12] performed an interesting experiment, where they put ordinary students into a prison setting. They were astonished to find out that those regular students that were put as wardens started to develop sadistic tendencies and those put as prisoners became distant and self absorbed and started to behave like real prisoners. Another good example of the importance of the environment is the "Broken Windows" theory [15]. This theory suggests that if people see broken windows around them, they assume that nobody cares and thus are less reluctant to engage in antisocial behavior. This theory was a base for the New York City fight against crime [1]. So to ensure the success of the new technology the environment should be analyzed and, if possible, controlled.

Understanding the context can help in correctly evaluating how successful the technology is. Halverson et al. [11] analyzed the patterns of use of a collaborative tool called Loops in a community in which it seemingly failed. The researchers were surprised to realize that once they took the Loops server down, assuming it is not needed, community participants began to complain about it. It turned out that this particular community used Loops in a scattered manner – only in times of software integration and some other parts of development process. Thus

analyzing the overall usage of the tool was not a good indicator of how successful the collaborative tool was. Rather, understanding the context of usage would have been much more helpful.

We will now proceed to describing the ReachOut tool and telling the story of its dissemination in two different communities with regards to the DOI theory.

3. THE REACHOUT TOOL

ReachOut is a tool for peer support and community building, created at our Research Lab. While the implementation details and the theoretical background of this tool are described elsewhere [14, 29, 35], we provide a short description of the ReachOut components that are relevant to this paper.

ReachOut is a methodology and a chat-based tool for peer support and community building. ReachOut bridges the gap between newsgroups, which focus on particular subjects and often retain discussion history, and real-time synchronous chats, which offer immediate awareness of new questions, a lighter tone, and a more interactive mode. ReachOut takes the best of both options, and adds push technology to portray new, by-topic awareness and mid-level persistence.

By logging in to ReachOut, users make themselves available for sharing their knowledge with their peers. On their first logon, users are encouraged to subscribe to groups of interest. Users are not expected to be experts of a domain in order to subscribe to it and in order to contribute to discussion – on the contrary, it is our belief that peer support is often much more suitable than expert support

A ReachOut discussion is very similar to a conference chat, though it is persistent through time; new participants can see full discussion transcripts. Users can also see the history of participation; a discussion transcript contains past entries, and the participant list does not only show active participants but also people who contributed to the discussion in the past and are not currently online. These persistent group chats enable the exchange of ideas and formulation of knowledge in the open; they allow the fine tuning of details and even the correction of occasional misconceptions that may be raised by participants. The persistence of discussion allows people from different time zones and different work habits, to still conduct fruitful discussions that may span over days.

ReachOut introduces a new paradigm for tacit knowledge management. It challenges the notion of managing knowledge by explicating it; and facilitates publishing information-needs and initial ideas, rather than trying to capture peoples' knowledge in profiles that keep getting outdated. While today's 'common practice' is to first look for information on the Internet or the organization's intranet, ReachOut suggests turning directly to people. This concept of "Ask before you search" is at the core of the tool, and we consider it its main innovation.

4. THE DISSEMINATION OF REACHOUT

After we designed and implemented ReachOut, we disseminated it in two different IBM communities. One dissemination process – in a Global community of Technical Sales Support people (GTSS) - started two and a half years ago, the other – in a local community of IBM Haifa Labs - a bit over a year ago. We did not use any systematic approach for dissemination; we were led by

common sense as well as field circumstances. We were quite keen to already have a running community and in retrospect we can identify many points in the process where we were rushing and neglecting important steps.

In this section we tell the stories of the two communities and try to point out the lessons we learned. We project these lessons on existing literature, to come up with a framework that would assist the planning of future dissemination processes.

At some stage of writing this paper we tried to make a comparison between our two communities, to decide which one was "more" successful. We discovered that the definition of a successful diffusion is not an easy one. Fichman and Kemerer [5] term the concept "assimilation gap" referring to the cumulative difference between two assimilation events, typically acquisition and non-trivial deployment of an innovation. However, the temporal aspect of diffusion – the actual adoption and use of an innovation over time – is not easy to quantify, and so non-trivial deployment is hard to define.

The following chart shows the levels of weekly online people (darker curve) vs. the size of install base in our two communities in similar periods of time since their beginning. The chart shows very different figures in the two communities. The top one (GTSS) shows a steadily growing install base, with online figures being relatively low; though the online levels are quite stable over a hundred weekly users. The bottom one (Haifa) shows a rather speedy growth of the install base in the first few weeks, and little growth in later weeks. The online figures in this chart are higher relative to the install base, yet the decline of this figure is quite clear.

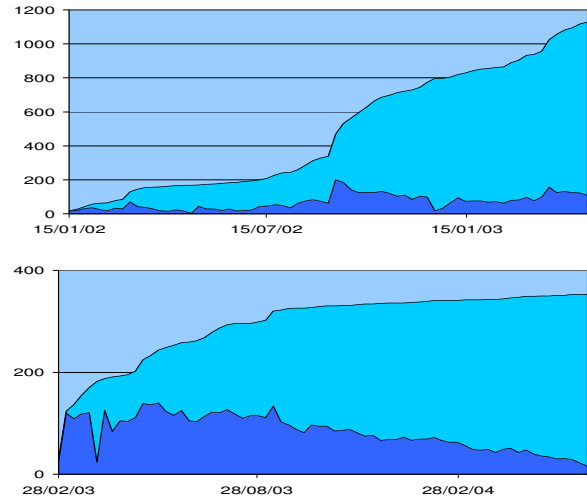


Chart 1: Online people per week vs. install base
top: GTSS bottom: Haifa

Many factors influence the differences between the two charts. The two communities are very different in essence: GTSS is a global community with potentially tens of thousands of users, while Haifa is a local community of a potential of 500 users. The Haifa community was launched more than a year after the launch of the GTSS community, and the tool it received was much more mature than the one initially given to GTSS users. The dissemination process was different in the two communities, as we will next tell in more detail.

Both communities had some level of success and yet neither one fully adopted ReachOut. While this fact may be attributed to the dissemination process as well as other factors such as the complexity of the tool and external factors such as competing tools, we still believe that carefully analyzing our dissemination process and reflecting it against the theories of diffusion in the literature would be beneficial to other CSCW developers.

4.1 The Chronicles of ReachOut

The idea of ReachOut – of the need of a tool that allows asking questions in the open – was conceived back in 1999 by our colleague, Amnon Ribak. In our paper [29], he phrased it as follows: “The dominating paradigm is that of a lonely explorer or navigator, and the implied motto is ‘Search before you ask’”. His dream was to implement the idea of ReachOut and introduce it as a solution for this problem of ‘lonely explorers’ and a lack of knowledge of what people next door work on.

In April 2001, ReachOut was introduced in an internal knowledge management conference of our company. This was obviously not the first time the idea was brought – in the form of a presentation with scenarios and mockup screens – in front of different executives in the hope to get it funded. However, it was the first time that brought results.

In August 2001, four months later, a group of Global Technical Sales Support (GTSS) people was looking for a solution that would allow practitioners performing similar tasks (sales support) all over the globe, to turn to their peers with questions, and be able to conduct group discussions with them, over time. A person, who took part in that correspondence and happened to have heard of ReachOut at the conference made the connection and suggested to turn to us. We did not have any concrete solution to offer at that point, but we jumped at the opportunity nevertheless, and by December 6th 2001, the first pilot of ReachOut was launched.

In the three months that passed from our first connection with GTSS and until the pilot took off, we rapidly implemented an alpha level prototype of ReachOut, patching together different pieces of earlier prototypes, and wrapping them with new code for the missing functionality. The version that was launched on December 6th, with an initial core team of 20 users, was very different than the ReachOut version we are running these days. The improvements in the tool were mostly achieved through participatory design with the growing audience of the pilot – and this intriguing process of co-design is outside the scope of this paper.

The audience of this instance of ReachOut was practitioners from different Communities of Practice that were invited to join the pilot at different times, as the word of ReachOut was spreading through the organization.

While the ReachOut pilot was taking place with the global community of practitioners, the original dream of using it for internal communication within our lab in Haifa did not go away. During the first year of the pilot, as ReachOut was improving and we got more and more proof to its usefulness, we periodically approached management in our lab proposing to launch another instance of ReachOut internally for the use of our lab’s people only. While the idea was well received, nothing materialized until November 2002.

In November 2002, Boaz Mizrahi from the Information Systems (IS) group in our lab, in charge of support and administration of Lotus applications, with a personal interest in collaboration technologies, heard about ReachOut and was given a one-on-one walkthrough demonstration of the tool by one of us. It was this serendipitous demonstration that gave the push for setting up an instance of ReachOut in our labs. Boaz took the matter to his managers, who were somewhat familiar with the notion of the tool but had never devoted it much thought. In the following couple of months, education sessions were held in several settings to IS personnel and management; a machine was dedicated to become the server of a local ReachOut instance; and an organized Web site was built for “getting started” information as well as methodological information on ReachOut .

By March 2003, we were ready for deployment, and Boaz sent a mass e-mail to all employees in our lab, inviting them to install ReachOut and take part in this collaboration experience. The population that received the invitation included about 500 employees – all lab employees – the majority of whom are co-located within one seven-story building. An analysis of the different functions that ReachOut was used for in the first two months of deployment in our lab appears in our paper [14].

4.2 Lessons Learned

In GTSS, the problem that the tool comes to solve was identified by the community people, and ReachOut was suggested as a possible solution and was decided to be adopted. This is a good example of *locating* the problem and *matching* a solution. In later stages of the diffusion process – as more Community of Practice leaders got to hear of ReachOut and decided to invite their community to join – we often heard people exclaim “why, this is exactly what we need” – people who used the common collaboration tools such as team-spaces or IM, but lacked the combination of persistence with light tone blended synchronicity. *Agenda setting* took place in this community – both in the case of first users who anticipated the implementation of the tool, and in the case of later invitees who learned of the tool as a good solution to a problem they experience. In Haifa, on the other hand, a problem was identified by a small group of employees, and was in fact not shared among too many others. Projects in Haifa are conducted in small groups of people, and it is generally acceptable by people to approach their few close peers for support, or be left on their own. The rest of the people in the building are not generally considered a source of knowledge. Employees either do not feel there is a problem in lack of lab-wide communication, or if they do they do not raise it as an issue and do not engage in trying to find a solution. Management may have been aware of the problem, but never classified it as an acute problem that must be solved. The adoption of ReachOut never seemed to be very high on the *agenda*.

The GTSS community lacks the typical organizational structure. Although it is a community within an organization, it seems that for the diffusion process it does not match the model of diffusion in an organization. A few champions in the community were the driving force behind it, and neither of them was in managerial position. No centralized IS group existed for the support and infrastructure – a dedicated server and some support were donated by one of the champions, and in fact the maintenance people were never really introduced to the functionality of ReachOut. However, once the server was installed and running and the

community started rolling, community leaders who saw the potential of ReachOut turned to their communities on their own and encouraged their participation. The weekly numbers of newly joined people in the GTSS ReachOut community show three very distinct peaks on March 2002, September 2002, and April 2003. These are the result of invitations of specific community leaders to their people. We see these as examples of the benefits of getting *managerial support*.

In the Haifa community that has a clear organizational structure, we did not enjoy an active support of management, but were allowed to disseminate ReachOut and were given stage time in all-hands meetings and approval to send mass e-mail invitations. Some managers, including the lab director himself, took personal part in ReachOut by logging in, raising questions, and taking part in discussions. We identified small peaks in activity (more postings and more viewing) on days in which the director was active on the tool. Getting the *support of the IS group* in the Haifa community turned out to be a more complicated issue than we had anticipated. As said earlier, it was Boaz from the IS group who gave ReachOut its biggest push. However, as we have more profoundly described in our paper [14], one of the functions that ReachOut happened to serve – one that we did not fully predict – is that of “Problem Alert”. ReachOut was discovered as a natural place where people can raise ideas or report problems in the open, allowing peers to support, object, and even improve their ideas, while making management aware of the idea or the problem. A bit late in the diffusion process did we discover that some of the “public reports” made over ReachOut were offending to some of IS people (while some others were offending to some of the HR people) and so these two groups – IS and HR – which are so important to the success of a diffusion process, actually nurtured feelings towards the tool. Had we predicted this difficulty early enough, and realized the *potential opposition* to the tool, we could have practiced some damage control such as better education and preparation for these two groups as well as some moderation of discussions.

The champions of the GTSS community used many different tools for *promoting* ReachOut. Personal one-on-one demonstrations as well as group conference calls for remote demonstrations were used. Mass e-mails of community leaders were sent to their people, and ReachOut announcements were published in different community places such as discussion forums and team-spaces. ReachOut was presented in at least three internal IBM conferences. As a result of these promotion measurements, the potential population of ReachOut kept growing and so the growth of its install base was natural. However, looking at Error! Reference source not found., we can see that while the install base is growing, the online presence does not grow as much. Additional *promotion for keeping* people online was missing. The dissemination actions in the Haifa community were very different than those taken in the GTSS community. Quite confident in the effectiveness of our tool after over a year in the GTSS community (maybe too confident), we neglected the step of *agenda setting* and nurturing the understanding of the problem ReachOut comes to solve.

In neither of our communities did we follow an organized process of identifying *social networks* and important figures that could assist in the push of the tool. In the GTSS community, as the diffusion process was gradual and at some point started happening naturally without our constant interference, we can point out

opinion leaders who influenced others to join ReachOut. But, as said, this was a natural process and not our doing. In the Haifa community we approached the whole community at once, in one mass e-mail. As we never used any organizational knowledge to identify opinion leaders, we can not tell if these had an effect on their close peer and their use of ReachOut. We did however notice a sad phenomenon of an opinion leader with a negative attitude towards ReachOut who at some points actively preached against the tool. Had we approached opinion leaders prior to dissemination, we may have had a chance to eliminate this negative effect which we know for a fact had a bad influence on our install base and usage levels.

ReachOut discussions are directed at interest groups to which users may subscribe as potential advisers. The interest groups are organized in a hierarchy and the hierarchy of a community is, in a sense, its profile of topics of interest. The hierarchy may only be modified by a moderator – users do not add new groups so as not to clutter the tree with too many similar groups. But users are welcome to raise new topics and suggest setting up new groups. The hierarchies of the two communities were naturally very different from each other. The GTSS hierarchy was composed solely of professional groups – of different industries and professions – while the Haifa community contained social interest groups as well as local administration groups. Both communities were launched with an initial hierarchy that was defined by its champions, and both hierarchies evolved as the usage of the tool became clearer. This evolution of the hierarchies is part of the *redefinition* of ReachOut and its adaptation to the communities.

In addition to the typical groups of each community, two interest groups were added artificially as part of the tool. One of these groups was titled “Sandbox – testing area” and was positioned as the first child of the root. The idea behind this group was to allow users to give ReachOut a try prior to asking “real” questions. People had no reason to subscribe to the sandbox group as it was not intended for real discussions, and so any newcomer could direct questions at this group, knowing it would not bother anyone. A “Resolve issue” button allowed the asker to remove a discussion from the system. Both these features – the sandbox group and the resolve button – contributed to the *trialability* of the tool.

The other special interest group was titled “The ReachOut tool”, positioned as second child of the root, and its purpose was to host discussions on the tool itself. We were actively monitoring this group and found it an intriguing channel of communication between users and developers. Discussion types on this group included newbie questions and bug reports, but also feature requests from users, and solicitation of comments that we as developers raised for getting user feedback. Especially in the first year of the GTSS community, we used the content of these discussions as the main factor of assigning our feature priorities. This involvement of real users in the improvement of the tool turned out to be a very important experience of *efficacy*, making users feel they have a part in the tool.

Through the first year of deployment – that is, in the GTSS community before the Haifa community was even launched – we released two new versions of ReachOut and a few minor fixes. In between versions the user experience of ReachOut evolved considerably – its main window became a slick bar that serves as a one-stop-shop for all activities and for getting notifications on

new and updated questions; the different statuses of questions (new, read, updated, etc.) were fine-tuned, and the notification mechanism was refined; more cues were added for supporting better awareness of people and questions; and the whole application became much more stable than the alpha level prototype that it was on the first release. These developments contributed to the *compatibility* and lack of *complexity* of the tool, as well as to its *observability* – with better awareness to online people and active discussions.

When the Haifa community was finally launched, we were quite pleased with the ReachOut version that was distributed there. It was much more mature compared to the alpha level pilot that was initially launched in GTSS. However, it was still not perfect, and lacking the correct agenda setting, people were not very tolerant to the few annoying bugs of the tool. By this time, ReachOut was not fully funded anymore, and new versions were seldom released. It turned out that the Haifa people, that received an “almost ready” technology, missed out on the *redefinition* phase of *efficacy*, as they did not get a chance to influence the usability of the tool. Moreover, perceiving they were taking part in a pilot, contributing ideas to the ReachOut tool interest group, and then hardly ever getting to see their requested improvements implemented, contributed to some resentment. We know of people who decided to stop using the tool “for now”, till a promised better version is released. Unfortunately we never got to release that version, and those people never came back. Another retrospect observation that we can make regarding the *efficacy* of our diffusion process has to do with our almost fanatic insistence to not save ReachOut discussions and “not add to the information overload and the ‘search before you ask’ paradigm”. We believe that saving the discussions would result in fewer open questions and less activity on ReachOut, but we now realize that this strong standing lost us users who had strong opposing beliefs.

The promotion and education method we used most in both communities is “*show by example*”. Initially, our champions actively posted ReachOut questions in order to add volume of activity to give others a place to take part. In fact, some of our champions harvested interesting and relevant questions from discussion forums of their communities and posted them on ReachOut – sometimes indeed proving the tool useful by getting answers faster. In later stages we realized that people are simply not used to asking their questions on ReachOut: when facing a question, they often do not even remember the option of ReachOut. *Routinizing* needed a push. In the Haifa community – which was our local community – we found ourselves educating people in the corridors by pointing out “why, this is a good ReachOut question” when overhearing people discussing an interesting issue. We then made it a point to use ReachOut for asking any appropriate question that we encountered in our work, just so that people realize what types of questions may be asked (any type, we believe). The GTSS champions used a similar approach.

5. DISCUSSION AND DISSEMINATION FRAMEWORK

In this section, we would like to discuss the findings from our experiences, to draw a sketch for a successful diffusion planning framework. We are organizing our framework using the theoretical foundation of the DOI theory. We take as the basis the

organizational innovation framework and discuss our recommendations in two stages – *initiation* and *implementation*. Within these stages we use features of both organizational structure and classic DOI concepts.

In organizations, management is likely to endorse “core” CSCW technologies, like email, calendaring or instant messaging and use its authority to make every user install and use the technology. However, for more innovative tools, management is much more likely to quietly support the technology, while not actively endorsing it and relying on people's judgment. Our framework facilitates the steps that need to be taken to try and achieve maximal success in this semi controlled process, while using the fact that people adopting the technology belong to one organization.

1) *Initiation*

• **Locating a problem and/or a need for a possible enhancement in organizational processes, which can be solved by introducing a new collaborative tool.**

As we have seen from our experience, this stage is very important in the diffusion process. One of our big obstacles in the Haifa community was potential users who did not understand the purpose of the tool and did not see how it can be useful to them. This stage was carried out in the GTSS community, since many practitioners were aware of the problem even before ReachOut was introduced. We saw a clear growth of the install base in GTSS community and almost complete halt in the Haifa community mostly due to this fact.

• **Matching a collaborative tool to the problem and changing it if required to fit the solution better.**

One of the clear differences between Haifa and GTSS communities was that while ReachOut was created to solve the Haifa's information sharing problem, it was fully designed and changed to fit the GTSS community needs. We can assume that this is another reason why people in Haifa did not associate themselves with the tool at the early stage.

• **Promoting – using different organizational channels to put the problem/enhancement on the agenda of potential users; and making sure it is not forgotten before starting the dissemination of the tool.**

We did not build the promotion correctly in either community. We bombarded the GTSS community with promotions using various channels in the early stages and we clearly saw peaks in the install base growth. We failed to keep the level of awareness high though, and this resulted in many users neglecting to come back. In the Haifa community we had many plans of using the right promotion. For example, we have a local electronic bulletin board where we planned to post announcement about new ReachOut discussions and invite people to join the tool. We also wanted the tool to be a part of the standard installation disk image in the Haifa Lab. But we never got to implement these intentions, for different reasons. This resulted in a steady but not growing user base.

• **Getting the support of the management and the Information Systems group – do not expect active endorsement, just ensure general support and the lack of objection. Do so by making sure everyone in the organizational structure – from**

management to IS – understands the purpose of the tool and does not object to it.

While management's support is certainly not a sufficient condition for a successful diffusion of a collaboration tool, it is definitely a necessary one [6]. Management does not necessarily need to actively endorse the tool (though it can be very helpful), but it is crucial that management understands the problem and the purpose of the tool and does not object to it. We had some managers literally forbidding people to use ReachOut, considering it a waste of time. Groups of such managers form "islands" where ReachOut was not even installed. Support from the IS group could have provided us with more efficient channels for the promotion of the tool and also have a commitment to support potential problems. Objection on the part of the IS group, on the other hand, led to some movement against the tool's installation.

1.1) *Properties of Innovation*

1.1.1) *Relative Advantage*

We do not have a specific suggestions regarding *Relative Advantage* of the innovation. It is clear from our experience that our tool by nature had intentions to supercede its predecessors – newsgroups and chat rooms. However, relative advantage may not always be clear and it should be examined, for example, by tools provided by [33, 34]

1.1.2) *Compatibility*

• **Start with the small pilot of users to get a feedback on how the tool serves its purposes.**

This experience in the GTSS community was a very successful one. While not eliminating all the problematic issues from ReachOut, it certainly improved the tool drastically and removed all the obvious problems.

1.1.3) *Complexity*

• **Try to make the tool as unobtrusive and intuitive as possible. Do not assume that users will perform much extra work.**

In our many iterations of improving the UI of ReachOut, we realized that people really wanted to do as little as possible to make the tool useful for them. Our assumptions, such as that users will tolerate a (very slim) always-on-top window sitting on their screen all the times were ultimately proven wrong. Our redesign efforts were focused on making the tool as unobtrusive as possible. Positive user's feedback immediately followed.

• **Test the tool to find all obvious and annoying bugs before going public.**

As we described in the chronicles of ReachOut, our tool was quite hastily implemented and actually contained many patches. We never managed to really leave the prototype level and productize the tool. The Haifa community was very intolerant to the tool's bugs. For example, visible reconnections to the Internet: a popup window with the connection information popped up in the middle of the screen while ReachOut was reconnecting to the network. This behavior has drawn many users away from ReachOut, just because they were annoyed by the tool. Some of them told us that they would definitely have stayed otherwise, but because they were mobile (disconnected often), they could not afford windows popping out on their screen so often.

• **Make sure that obtaining the tool requires a small or no effort from the end users.**

We had a central Web site where people could download ReachOut and install it. After installing, ReachOut used standard IBM databases and passwords for authentication and security. However, it was too much for some users, especially with slow connectivity. Again, having ReachOut as part of a standard install image, at least in the case of the Haifa community would eliminate the process of installation and create even more potential users.

1.1.4) *Trialability*

• **Add features that allow users to try the tool without leaving too much of a trace, before they actually start using the tool.**

We added the Sandbox option and it proved to be a very useful feature. Many people used it to play with ReachOut without interfering with others' activities. It was often used by moderators to show people how ReachOut can be used and helped us to promote the tool.

1.1.5) *Observability*

• **Make sure that the success stories of the tool are publicized widely**

We felt that ReachOut had many success stories in both communities. However, we never really made any effort to make these success stories public. ReachOut was not perceived as a very successful tool. The success was evident for the participants but not for outsiders. Such publicity would definitely contribute to a greater success of the tool.

1.1.6) *Efficacy* – allowing users to influence the functionality and design of the tool

• **Constantly verify the actual use of the tool to identify potential problems – modify the tool if required**

• **Let users influence the design of the tool by suggesting potential new features and improvements.**

People used the ReachOut tool group to let users discuss the current and future features of the tool. As we describe elsewhere [14], many discussions has been conducted in this group and at least in GTSS community there was a clear satisfaction and sustainability of users after new versions. In Haifa community we often did not deliver new features requested, factor that contributed to users' leaving.

2) *Implementation*

2.1) *Diffusion Networks*

• **Giving the tool to early adopters first, so you have a user base, but in parallel trying to use organizational knowledge to identify the early adopters and determine information flow networks – find opinion leaders.**

We did not identify opinion leaders in neither of our communities, but we did see both positive and negative results of their influence. In the GTSS community a single e-mail by a clear opinion leader resulted in the largest peak in the install base growth in just one day (see clear acceleration in install base growth on September 15 2002 on Error! Reference source not found.). In the Haifa community, an opinion leader with a negative attitude created a strong opposition to the use of the tool. Locating opinion leaders is not a trivial task. But generally, managers know these people and can easily name them. Using the organizational knowledge to carry out this stage may be very helpful.

- **When "selling" the technology to opinion leaders, do not necessarily use technologists – try to find "salespeople" – people who are good in convincing other people.**

While in the GTSS community the technology was “sold” by champions, who often had good selling skills, in Haifa the developers themselves usually promoted the tool. Many people (rightfully) saw the developers as biased, so our argument was not very convincing. Using people with the right skills who are not so intimate with the tool can help promoting it better. These people might be champions, like Boaz in the Haifa community or they might be representatives from IS or HR group (this was also found by Mumford and McDonald [22]).

2.2) *Managing the Context*

- **From the early pilots try to see who in the outside setting is mostly opposed to the tool and address their issues. Explain the benefits to the opposing people and try to prevent the situations causing the problem, either by educating users or by changing the tool.**

As we learnt from ReachOut experience in Haifa, some people felt threatened by the tool, since the context of its use became dangerous to them professionally. Not all, but some of these settings may be noticed during the pilots and preventive steps (like discussion moderation) can be taken to prevent the unintended or harmful usage.

- **Foster collaboration over the tool:**
 - **When relevant, Show by example how the tool may be effectively used, so people clearly understand the intentions of the tool**

Moderators should show by example how the tool can be used. We and our champions often asked example questions and showed to the wider user community how fast and efficient the information discovery process can be using ReachOut. We also tried to provide the right and efficient context for using the tool by setting type of questions that can find a quick answer in the tool.

- **Welcome newcomers graciously to encourage their return**

As we described elsewhere [35], first experience often shapes the future behavior on the collaboration tool. This is why it is very important to give newcomers the warmest welcome and let them remember their first experience as a pleasant one. Our champions in the GTSS community did a great deal of effort to pursue this goal and in our other paper there is a statistical evidence that people who had a more pleasant first experience had greater chances to come back.

Note that this framework does not take into consideration an individual's characteristics that no doubt influence decisions regarding adoption. It is very difficult to control individual characteristics in a large organization. For example, a person that objects to Instant Messaging for privacy reasons may immediately reject the new IM based tool disregarding any action taken. However, if all suggested stages are successfully performed, it is expected that the vast majority of potential users, who do not have personal bias toward the technology, will adopt it.

6. CONCLUSIONS

As time passes, more and more virtual communities are being created. Collaboration technologies become an integral part of our lives – both at home and in the workplace. This is why the problem of diffusion of collaboration tools is so relevant today.

We feel that there are a lot of experiences – some good and some bad – in disseminating CSCW applications and putting a theoretical framework behind them can be very helpful.

This paper proposes a framework based on the Diffusion of Innovations theory. It is only natural to deal with the application diffusion problem using this theory. However our work is only preliminary. It can be enriched by analyzing each of its steps in depth. It can be fine tuned by applying it to real applications and analyzing the outcomes. Other frameworks can be proposed to plan for a successful dissemination. One thing is important though – dissemination of CSCW tools, especially in large organizations, should be done very carefully with the support of strong theoretical tools.

We are now working on many technologies. The user experience of ReachOut, in different implementations, is being integrated into solutions, and our portfolio of collaboration technologies has grown. One thing is certain - the next time we face the challenge of collaboration technology diffusion, we shall follow the proposed framework, and target for success.

7. ACKNOWLEDGMENTS

We thank Amnon Ribak who first came up with the idea of ReachOut. We are grateful to Joi Grieg for connecting us to the GTSS team. Sigalit Ur, Noga Meshulam, Raanan Nevet, and Sagee Rosen are the real people behind ReachOut and we are indebted to them. Mark Palmer and Boaz Mizrachi deserve special thanks for being the power behind the deployment of ReachOut in IBM. Dianne Sonnenwald provided a lot of useful feedback to us and we are indebted to her. Finally, we wish to thank IBM for letting us run the pilots, and all participants, for contributing to the community building through ReachOut.

8. REFERENCES

- [1] Bratton, W. *How America's Top Cop Reversed Crime Epidemic*. Random House, New York, 1998.
- [2] Coleman, J.S., Katz, E., and Menzel, H. *Medical Innovation: A Diffusion Study*, Indianapolis: Bobbs-Merrill, 1966.
- [3] Dugan, R.F., Glinert, E.P., and Rogers, E. H. CAMELOT: Technology Focused Testing of CSCW Applications, *Proceedings of 36th Annual Hawaii International Conference on System Sciences (HICSS'03)*, January 06 - 09, 2003, Big Island, Hawaii, 328c.
- [4] Erickson, T., Smith, D. N., Kellogg, W. A., Laff, M., Richards, J. T., and Bradner, E. Socially Translucent Systems: Social Proxies, Persistent Conversation, and the Design of “Babble”, *Proceedings of ACM CHI '99*, ACM Press, 1999, 72-78.
- [5] Fichman, R.G. and Kemerer, C.F., The Illusory Diffusion of Innovation: An Examination of Assimilation Gaps, *Information Systems Research*, 10, 3, 1999, 255-275.
- [6] Gallivan, M. J. Organizational adoption and assimilation of complex technological innovations: Development and application of a new framework. *Data Base for Advances in Information Systems*, 32, 3, 2001, 51-85.
- [7] Gladwell, M. *The Tipping Point: How Little Things Can Make a Big Difference*, Little Brown, New York, 2000.

- [8] Granovetter, M.. The Strength of Weak Ties, *American Journal of Sociology*, 78, 1973, 1360-1380.
- [9] Grudin, J., Why CSCW Applications Fail: Problems in the Design and Evaluation of Organizational Interfaces, *Proceedings of CSCW 1988*, 1988, 85-93.
- [10] Hagel, John III, and Armstrong, A.G. *Net.Gain: Expanding Markets Through Virtual Communities*, Harvard Business School Press, Boston, MA, 1997.
- [11] Halverson, C., Erickson, T. and Sussman, J. What counts as success? Punctuated patterns of use in a persistent chat environment, *Proceedings of the 2003 International ACM SIGGROUP Conference on Supporting Group Work*, 2003, Sanibel Island, Florida, 180-189.
- [12] Haney, C., Banks, W. C., and Zimbardo, P. G. Interpersonal dynamics in a simulated prison. *International Journal of Criminology and Penology*, 1, 1973, 69-97.
- [13] Isaacs, E, Walendowski, A, Whittaker, S, Schiano, S. J. and Kamm C. The Character, Functions and Styles of IM in the Workplace. *Proceedings of CSCW 2002*, New Orleans, LA, 2002, 11-20.
- [14] Jacovi, M., Soroka, V. and Ur, S. Why do we ReachOut? Functions of a semi-persistent peer-support tool. In *Proceedings of the ACM International Conference on Supporting Group Work*, Sanibel Island, Florida, USA, 2003, 161-169.
- [15] Kelling, G.L. and Coles, C.M. *Fixing Broken Windows*. Touchstone, New York, 1996.
- [16] Lazarsfeld, P., Berelson, B., Gaudet, H. *The People's Choice*. New York: Duell, Sloan and Pearce, 1944.
- [17] Mark, G., and Poltrock, S. Shaping technology across social worlds: groupware adoption in a distributed organization, *Proceedings of the 2003 International ACM SIGGROUP Conference on Supporting Group Work*, 2003, Sanibel Island, Florida, 284-293.
- [18] Mark, G., Poltrock, S., and Fisher, D. Remote communication and technology diffusion, *Proceedings of IEEE Professional Communication Conference*, 2001, 53-70.
- [19] Markus, M. L. Toward a critical mass theory of interactive media: Universal access, interdependence, and diffusion, *Communications Research*, 14(5), 1987, 491-511.
- [20] Milgram, S., The Small World Problem, *Psychology Today*, 61, 1967, .60-67.
- [21] Moore, G.C. and Benbasat, I. Development of an instrument to measure the perceptions of adopting an information technology innovation, *Information Systems Research*, 2, 3, 1991, 192-222.
- [22] Mumford, E., & McDonald, B. *XSEL'S progress*. London: Wiley, 1989.
- [23] Nonnecke, B. *Lurking in email-based discussion lists*, Unpublished Ph.D., South Bank University, London, London, 2000.
- [24] Nonnecke, B., and Preece, J. Shedding light on lurkers in online communities. In Buckner, K. (ed.), *Ethnographic Studies in Real and Virtual Environments Inhabited Information Spaces and Connected Communities*, Edinburgh, 1999, 123-128.
- [25] Olson, G. M., & Olson, J. S. Distance Matters. *Human-Computer Interaction*, 15, 2/3, 139-178.
- [26] Orlikowski, W J, Learning from Notes: Organizational issues in groupware implementation. *Proceedings of CSCW'92* October 31-November 4, Toronto, Canada, 1992, 362-369.
- [27] Pickering, J. M. and King, J. L. Hardwiring weak ties: Individual and institutional issues in computer-mediated communication. *Proceedings of CSCW'92* October 31-November 4, Toronto, Canada, 1992, 356-361.
- [28] Rheingold, H. *The Virtual Community: Homesteading on the Electronic Frontier*, MIT Press, Cambridge, MA, 2000.
- [29] Ribak, A., Jacovi, M., & Soroka, V. Ask before you search: Peer support and community building with ReachOut. In *Proceedings of CSCW 2002* New-Orleans, Louisiana, USA, November 16-20, 2002. ACM Press, New York, USA, 2002, 126-135.
- [30] Rogers, E. M. *Diffusion of Innovations*, 5th Ed. New York: The Free Press, 2003.
- [31] Rogers, Y. Exploring obstacles: Integrating CSCW in evolving organizations. *Proceedings of CSCW 1994*, 1994, 67-78.
- [32] Ryan, B. and Gross, N. The diffusion of hybrid seed corn in two Iowa communities. *Rural Sociology*, 8, 1, 15-24.
- [33] Sonnenwald, D. H., Maglaughlin, K. L., and Whitton, M. C. Using innovation diffusion theory to guide collaboration technology evaluation: Work in progress. *Proceedings IEEE 10th International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, 2001, 114-119.
- [34] Sonnenwald, D.H., Whitton, M.C., Maglaughlin, K., Evaluating a scientific collaboratory: Results of a controlled experiment. *TOCHI*, 10, 2, 150-176.
- [35] Soroka, V., Jacovi, M. & Ur, S. We can see you: a study of the community's invisible people through ReachOut. In *Proceedings of C&T '03*, Amsterdam, The Netherlands, 2003, 65-79.
- [36] Stone, A.R. Will the Real Body Please Stand Up?: Boundary Stories about Virtual Cultures, *Cyberspace: First Steps*, Michael Benedikt (ed.), The MIT Press, Cambridge, Mass, 1994, 88-92.
- [37] Zaltman, G., Duncan, R., and Holbeck, J.. *Innovations and Organizations*, New York: Wiley & Sons, 1973.