

COLLABORATION AND OPENNESS IN LARGE CORPORATE SOFTWARE DEVELOPMENT

Catharina Melian,¹ Cathy Burles Ammirati,²

Pankaj K Garg,³ and Guje Sevón⁴

Abstract

Software program development is an intensively personal, innovative process, sometimes compared to an artistic process or a play (La Plante and Seidner 1999). Large-scale software development, however, is by its very nature a collaborative effort (Brooks 1975). In this paper, we start a dialogue to address the issue of individual versus collaborative effort in software development. In particular, we raise the issue with respect to Open Source development process: at what time in the process of software development does it make sense to “open” the process to collaborators? An early opening may result in a poor project, where the original idea gets diffused and never reaches a useful conclusion. A late opening can result in the project being too late to “market,” and face stiff competition from earlier, more established ideas. We explore such issues in the context of adopting Open Source practices within a corporate environment, through “Progressive Open Source.”

¹ Stockholm University School of Business, Sweden, cme@fek.su.se

² HP, Vancouver, WA, USA, cathy_ammirati@hp.com

³ HP Labs, Palo Alto, CA, USA, garg@hpl.hp.com

⁴ Stockholm School of Economics, Sweden, guje.sevon@hhs.se

1. Introduction

We report on an ongoing research to understand how openness, i.e. opening up for partners to engage in joint or collaborative work, effect the process of innovation. *The aim of the research is to study institutionalization processes within a certain open source development process. Specific interest will be devoted to the impact of the local context and the significant actors' experiences and their perceptions about what constitute innovation in different phases of the process.*

Collective work is to a large extent depending on convention, but it is also spurred through innovations (Mukerji 1998). A recent trend in the software community is to utilize a virtual workspace on the Internet, facilitated by common toolboxes and techniques imported from the open source community. People closely related to the Open Source Movement have founded commercial entities offering infrastructure solutions to large corporation, e.g., CollabNet. Brian Behlendorf , the CTO of CollabNet:⁵

“Creating a virtual work space. Making it as easy as picking up a phone to start collaborating with (developers) and sharing ideas. That’s pretty powerful. Then if I can take what’s developed there and put it in some permanent archive someplace where I can refer to it later, then that’s pretty cool. That plays in very well to the type of stuff we’re doing here: Building software for building infrastructure that allows groups of people to collaborate on software development. This kind of real-time, shared white board chat is next stage.”⁶

The implications of opening up strategic processes to actors outside the formal organizational entity are important for understanding how to organize for innovation, play and experimentation (Rushkoff 1999). Experimentation opens up for organizational exploration, thus enabling new participants to contribute with ideas and resources (March 1994). Collaboration implies the inclusion of partners, but also simultaneously excluding others. We wish to analyze on a theoretical or a generic level what collaboration is, and what it can be. The findings of our research may be applicable in practice, specifically when it comes to collaborative software engineering efforts within a large corporation and between its multiple third party collaborators⁷. Third party collaboration enables large companies to focus on developing certain parts of a product or a service while leaving less strategic components to be completed by an outside vendor or partner. A collaborative strategy in that sense facilitates and serves deliberate outsourcing efforts (Quinn 1992). The goal is to free resources and focus on developing that which gives the company a competitive advantage. A collaborative strategy supposedly allows partners in an ongoing trusting relationship to view processes or products that otherwise would be withheld due to Intellectual Property reasons. It is a controlled openness targeted to strategic partners in an ongoing valuable business relationship, allowing the partners to utilize and leverage from each others existing inventions. It is assumed that some Intellectual Property may actually become more valuable when acknowledged and utilized by a third partner – it depends on the resources of the third partner. When trying to understand how openness effect the travel of ideas, and the progress of innovative products and services, we compare the factors that Open Source projects with Progressive Open Source, i.e. controlled openness between partners or actors in a secured corporate environment.

⁵ Other commercial entities offering similar solutions are SourceForge provided by VA Linux.

⁶ <http://sanfrancisco.bcentral.com/sanfrancisco/stories/2001/05/21/newscolumn8.html>

⁷ Third party collaborators are typically outside vendors, contractors, customers, and/or academics.

Organizational Collaborations

Business leaders and CTOs worldwide believe that implementing a collaboration strategy potentially can ensure greater revenue streams, e.g., through increased communication among employees, customers and between companies, facilitated by technology (Biggs 2001). Breaking down walls and building a collaborative culture shall supposedly encourage and improve communication, enable sharing of knowledge, empower people, facilitate organizational learning and bonding, and most importantly improve the quality of life at work (Siviter, Petre, Klein 1997). Software is believed to be the key to maintaining control while at the same time increasing productivity (Toupin 2001). In fact many large companies expect their partners to have a collaboration capability. They perceive of themselves as system integrators putting together things other people or organizations have created in to products that suit the individual customers.

Hewlett-Packard CEO Ms Carly Fiorina recently expressed it like this when talking about Linux:

“The Linux movement is based on openness, on the idea — and the evidence — that the achievement of our collective work is greater than the sum of individual efforts. That everyone benefits when everyone else advances”... “Like the Web itself, Linux is an open source technology that spawned an entire industry — and that continues to be improved by smart minds collaborating all over the world.”⁸

Increasingly, new batteries of communication technologies enable project participants to connect to a dispersed social network (Castells 1996). And in the wake of new technologies people on a regular basis wield their personal social networks to accomplish their every day work (Nardi, Whittaker & Schwarz 2000). Corollary, there is an emergent need to enable people to work together effectively through computers (Mills 1999). For example, teams must be able to establish overarching work plans, divide tasks and subsequently coordinate those tasks.

In this paper we wish to analyze on a theoretical or a generic level what collaboration is, and what it can be. We also want to grasp what it means to actors in practice, specifically when it comes to software collaborative engineering efforts within a large corporation and between its multiple collaborators. We are starting to ask questions such as – why should organizations and/or individuals collaborate? i.e., what may be the potential benefits? When trying to understand and define what collaboration is all about it is perhaps useful to derive its meaning. Etymological, to collaborate means *to labor together, especially in an intellectual endeavor*.⁹ Furthermore it also means *‘to cooperate with or willingly assist an enemy of one’s country and especially an occupying force.’* In addition it may also entail ‘to cooperate with an agency or instrumentality with which one is not immediately connected. Collaboration embraces the spectrum between colleagues working together in a trusting relationship, through instances of different stakeholders trying to accomplish their individual and separate goals, and even situations when adversaries or competitors are compelled to pursue acts of collaboration (Doheny-Farina 1986, Matusov 1993, McMaster, Jones & Wood-Harper 1997, Newman & Newman 1993, Self 1992, Cohen, Cash & Muller 2000). Previous research by Cohen, Cash & Muller indicate that collaboration between stakeholders, or adversaries, “rely on the strategic manipulation of awareness of the existence and availability of information, including documents, people, and processes. Their ability to control access to these resources by selection is one key to their success” (2000:34). The deliberate attempts to curtail information impacts collaboration, especially when parties outside the organization are participating. In this paper we want to address this ambiguity embedded in collaboration, since involving third parties in collaborative activities potentially has an impact on e.g., the system design i.e.,

⁸ Fueling Innovation, Opportunity with Linux LinuxWorld Conference 2002 New York, New York — January 30, 2002

⁹ Websters dictionary.

membership and level of access to information and knowledge perhaps has to be controlled. People from different organizations or from different parts of an organization obviously need to come together and collaborate, however it may at the same time be necessary to control membership and visibility while maintaining as much flexibility as possible (Clement & Wagner 1995). In practice, it is perceived as important to allow flexibility in regard to different layers of access¹⁰, e.g., users may need to share documents and content selectively and on ad hoc basis.

The two-sidedness of collaboration in the Webster dictionary meaning can be interpreted as the act of working together with somebody, i.e., meaning collaboration in the positive sense (maybe with characters added as stated out below: problems of having own ideas being stolen, etc.) but at the same time excluding other people from the collaboration (thus being a collaborator in the negative sense). This situation is possible when an idea is owned, or otherwise important to different organizations, or part of organizations, or individuals at the same time – and you collaborate only with some, others being excluded. The collaborator, then, may be seen as the traitor (or its synonyms, as double agent, deserter, defector, etc.). This is not a quality of the individual collaborator, neither of the collaborative work, it is a consequence of the structure itself, the structure that describes the typical situation: an interest, an idea etc. is not owned or known by a certain agent¹¹ it is mostly at least to some part a general goods, or of general interest. When collaboration, then means not only including people in work process, but also means excluding other, these other, then feel they have the same (or more) right to be included, perhaps even exclusively being the ones accepted. There are, of course, possible to variegate on this theme: A third party may be seen as traitor by another (now totally excluded) organization that is a competitor to the focal organization.

Opening up and sharing ideas at first seems rather appealing. It is a kind of courtship to collaborate. Courtship, to be of any interest means not only invitation to something, but also excluding others from this. The risk we face when opening up and courting others to collaborate, is that we turn in to 'a fellow traveler or a sympathizer', in the negative sense. - Why is that perhaps negative? For one thing it may mean that we sell out our own ideas to others, e.g., individual actors or organizations. It may also mean that we become less creative, through the eagerness of pleasing the other partners. We are less prone to pursue our individual ideas. Ideas are negotiated and power relations sometimes have an impact on what actually gets highlighted. Sometimes ideas are concealed until political issues are settled (Newman & Newman 1993). Other well-known negative aspects of collaboration are a general unwieldiness of cooperation and cooperative endeavors between intellectuals. This is often due to the fact that individuals are afraid to get their ideas taken away from them. Only very seldom can ideas on an immature and early phase, be protected by Intellectual Property rights such as patents, and copyright (Lessig 1999). Individuals may also be afraid of jeopardizing their careers and their reputations for revealing premature ideas. Collaboration then becomes an issue of weakening or strengthening a competitive relationship between actors in a collaborative setting.

Research by Tankjær on the importance of an 'open house strategy' indicates that the process of inviting actors to participate in projects is a powerful way of increasing the perceived importance of the project, its scale and scope. The illusion of openness and visibility, thus creating a transparent process spurred individuals, organizations and institutions to participate. It was perceived as risky to be excluded. However, the openness and transparency in fact were illusory in the sense that only an exclusive selection of organizations actually exerted influence on the definition and execution of the project. What actually motivated actors, and especially institutions to participate, was the fact that they did not want to be left out. They'd rather be participating and collaborating than left outside. This

¹⁰ I.e. selective dissemination of information.

¹¹ Organization and /or individual

indicates that the degree of Openness somehow effects collaboration. Our research aims at trying to grasp this particular relationship, i.e., the relationship between openness and the amount, extent and intensity of collaboration.

To sum up, collaboration denotes something positive in that it embraces the notion of working together, and to possibly conquer an intellectual endeavor. But it also denotes something less positive and constructive, i.e., control, that what is not legitimate, to omit or to leave out, to be in collusion or perhaps in cahoots with the enemy. The ambiguity embedded in the idea of collaboration and collaborative efforts are important to explore when bringing third parties in to the process of inventing new products and services. In its positive sense, collaboration embraces, includes and encourages creative intellectual processes. But clearly it is equally important to acknowledge and emphasize aspects of control, visibility and trust, and how it affects the way people go about doing their every day work.

In the following section we discuss Progressive Open Source and its implications for a large corporation. We proceed with an exposition of some of the organizational challenges that have been acknowledged so far. Thereafter we give a description of an ongoing case study at Hewlett-Packard that seeks to address issues of fostering global networks of software communities. The article is concluded with some tentative results and usage statistics.

2. Progressive Open Source

Dinckelacker et al. (2001) define “Progressive Open Source” as a strategy for large corporations to adopt Open Source software development methods. In essence the concept encompasses coordinated resource sharing and problem solving in a dynamic, multi-actor virtual network. The hypothesis is that by adopting Open Source development methods within a corporation, the corporation can gain from the collaboration styles of the Open Source software methodology resulting in robust code quality, features that are well-tuned to user’s requirements, strong, well-established networks of communities of practice, and so forth (Brown & Duguid 1991, Chaiklin & Lave 1993, Lave & Wenger 1991). Progressive Open Source (**POS**) advocates the progressive adoption of Open Source practices by a corporation in primarily three stages:

- **Inner Source:** open the software source code to only employees of the company,
- **Controlled Source:** selectively open source code to third parties and partners, and
- **Open Source:** open source code to the entire Internet community.

Each one of these stages results in it’s own collaboration networks. Any given software project within the corporation can choose to participate in one or more of the three collaboration practices. Given the common starting point of the corporation, however, each will eventually benefit from, and utilize, the networks of the other two communities through cross-linkages and common members.

The three identifiable collaboration styles of POS are: (1) the internal collaboration of software developers within the company, (2) the collaboration of people mixed from the company and its partners, and (3) the Open Source software development collaboration. The third kind of collaboration, the Open Source method, has recently received much attention from researchers (cf. Mockus et al., 2001; Moon and Sproul, 2000). In this work, we are more interested in understanding the first two kinds of collaboration practices i.e., collaboration around software systems within a corporation and the corporate partners and third party relationships. In HP, we have defined two

programs to leverage such novel collaborative practices: (1) *Corporate Source*, and (2) *Collaborative Development Program (CDP)*.

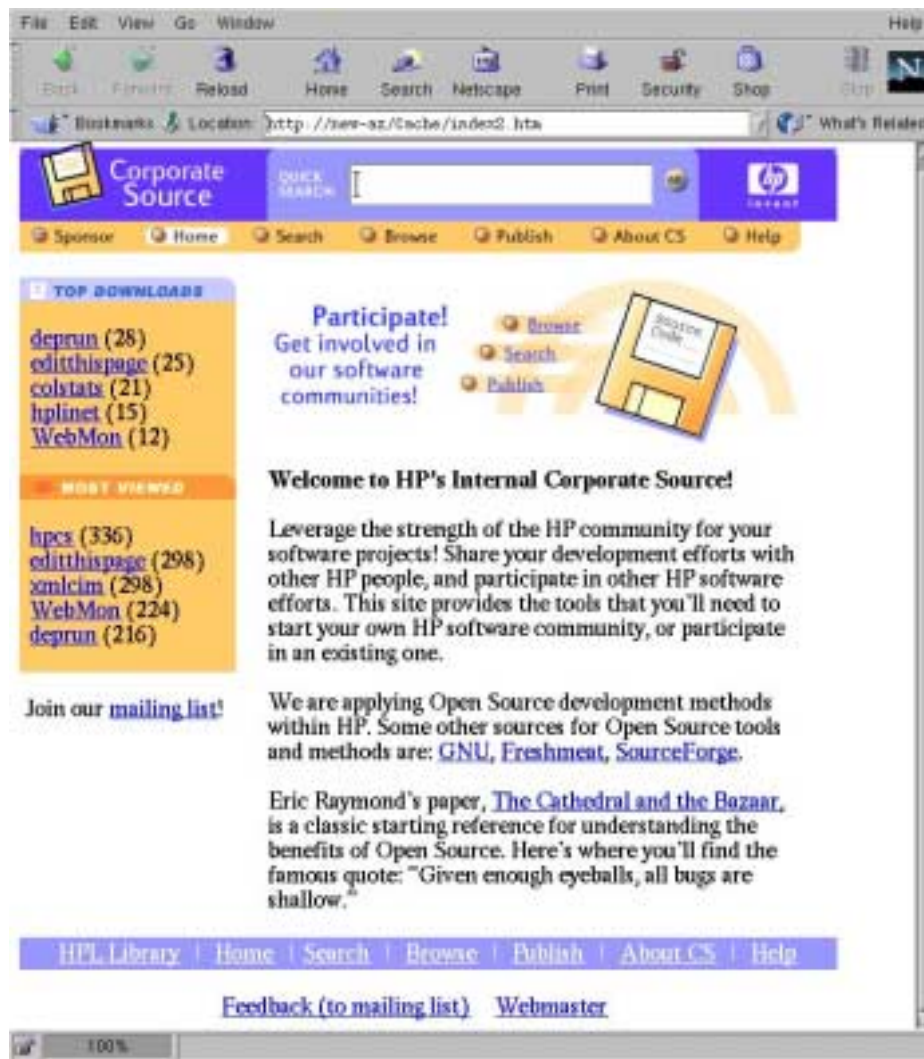
Corporate Source advocates the use of flat, networked organizations for software development in large corporations. To explain the idea, we contrast it with the current, hierarchical form of organization for most software projects. In companies like HP, software products are organized in hierarchies, either functionally or market-driven. Hence, a product group in the printer division writes all printer software, and the operating system group in the computer systems organizations writes the operating system software. The only connection between these two groups is through the Chief Executive Officer, who is often up to ten or twenty levels higher than the engineering groups working on the product. The source code of the software from one group is rarely available to the other group. Hence, if there is a problem in the interface between the printer driver and the spooler on the operating system, several layers and channels of communication and coordination have to be crossed to address the issues. In contrast, Corporate Source advocates that the two groups (and all other groups in the corporation) should freely make available their source code among themselves. In this manner, the printer group should be able to make changes to the operating system spooler source code, and the operating system group should be able to modify the printer driver source code. The ownership and control of what ultimately goes in the product still rests with the original owners.

Corporate Source borrows heavily from the Open Source development paradigm (DiBona et al., 1999) and from the methods of scientific research. From the *Open Source development paradigm*, it borrows the notion of making source code available freely (openly) for all members as a basis for collaboration; advocates the use of open email discussions for feature addition, implementation, review and testing; and provides a persistence base for the source code and email discussions to be available long after their creation date. In this manner, a new person can quickly join a project by understanding the rationale behind some feature selection and implementation (Raymond, 2001). We utilize the World Wide Web (WWW) infrastructure to make Corporate Source projects freely available for all employees to browse through and participate, through a familiar employee portal. From the *scientific methods*, Corporate Source borrows the notion of “publishing” work for peer-review and criticism, and archival storage of important experimental results for future review. Hence, the primary responsibility of facilitating the use of Corporate Source rests with HP’s research library, which is also the primarily responsible for maintaining and disseminating HP’s scientific knowledge as technical reports.

The HP Research Library hosts the Corporate Source service, along with some other knowledge management services, such as a database of skill set of employees, an Idea forum, technical reports, and so forth. Figure 1 shows a typical screen of the Corporate Source service that provides a collaboration hub for the users of Corporate Source. HP employees can publish their own software, update an existing software, search or browse through existing software, or comment, criticize or review existing software or discussions. Traditional hierarchical organizational boundaries are minimized by only exposing relevant information about a user’s network identity and skill set. Any given user’s hierarchical position can, of course, be determined through some of the attributes of the network identity. Hence, one cannot truly achieve a virtual network identity as in the case of the Open Source development, where any given user can completely hide behind a network identity. The nature of the discussions and the corresponding contributions to the software, ideas, and thoughts, therefore, may be different in Corporate Source than in the Open Source communities. We are looking at ways of studying such behavior in order to effectively mitigate the loss of value from artificial hierarchical boundaries of current organizations. We can then truly realize the potential of Corporate Source by empowering even the junior-most member of the organization (who may not be professionally trained as a software engineer) to make far-reaching and wide contributions to the corporation’s software, similar to what we see in the Open Source communities when high-school graduates are able to shake

up the software and media industry, for example with the work of peer-to-peer computing of Napster¹² (Clarke, Sandberg, Wiley & Hong 1999, Alderman 2001, Rose & Buchanan 2001).

Figure 1: Main screen for Corporate Source



The **Collaborative Development Program (CDP)** is advocated by HP's Imaging and Printing group that develops the printers and related products for HP. Several printer products have relationships with each other and third party products, e.g., the all-in-one office jet has some features that are in common with a printer, while other features that are in common with a scanner, and so forth. The source code for the firmware and software for these features, therefore, must be shared between the groups that provide these products. Similarly, software for some of the foundational features, like networking, is common to all the groups. The goal of CDP is to foster appropriate networking, collaboration, and community spirit among the various groups that participate in such development, including third party individuals who may not be employees of HP.

While the Corporate Source program is a grass-roots program from HP Labs, CDP is a group funded and organized program that has executive champions, sponsors, R&D change leaders, and information

¹² Napster is a protocol for sharing files between users. See e.g., <http://opennap.sourceforge.net/>

technology staff. The executive champions have been critical in establishing the credibility of the program and establishing a need for collaboration among the different groups in the organization. The sponsors are critical in committing resources that enable short-term and long-term nurturing and success of the program. The R&D change leaders enable a dialogue of cultural change and training in the organizations to begin the long process of transition from a hierarchical, product-focused organization to a networked, collaboration-focused organization. The IT staff provides the critical collaboration infrastructure on a 24x7 supported basis. The CDP infrastructure supports email discussions and bulletin boards, source code repository (searchable and browsable), and defect tracking. While the Corporate Source infrastructure is a home-grown combination of some available Open Source tools like CVS (Fogel 1999) and Mailman¹³, CDP relies on a third-party, CollabNet, Inc., to provide the tools infrastructure.

The CDP infrastructure resides on the Internet (as opposed to the Intranet for Corporate Source). Hence, bringing in a new third party on-board the CDP infrastructure is a relatively straightforward task. Indeed, the time for establishing a new collaboration project can be reduced to a matter of few days from what it used to be a few months or weeks at best. If all the project participants are from HP, then the setup takes a few hours. Each user in CDP is given a network identity, which is based on their corporate identity. The hierarchical organizational identity, however, is not that easily visible although it can be deduced quite easily. CDP organizes people by projects and by default any HP employee is given read-access to any “open” project in CDP. CDP promotes the sharing of knowledge and information to build a community that will deliver on the priorities across the company. Ultimately the goal is to do away with any organizational boundaries to allow engineers to apply their expertise to provide the greatest return for the company by enabling project teams to deliver innovative solutions faster and with greater reliability.

Both Corporate Source and CDP have been operational in HP for a several months. Corporate Source was officially launched in June 2000. CDP was launched in April 2001. In early 2002 Corporate Source has about 1500 registered users while CDP has 3000 users (10% of whom are non-HP). Forty-five external companies are developing projects with HP using CDP. Corporate Source has about thirty projects, all of which are research projects that are not tied to any HP product. CDP has about 350 projects, most of which are tied to specific HP products. Corporate Source has users in forty-five countries; CDP has users from at least eighteen countries. Both community hubs have active users, although we need to work on increasing the awareness, adoption, and use of Progressive Open Source within HP. Along with providing the right set of tools, we must provide the appropriate organizational structures, rewards, and motivations to transition HP software development into a more collaborative, Open Source style of development.

2.1 Organizational Challenges

The traditional hierarchical organizations reward and promote cohesive project or product related behavior. For example, in HP, individuals are evaluated every year on their contributions to their assigned project. While community-help and visibility is encouraged, it is not the main factor when considering the yearly progress of an employee or their managers. Helping out another person in a different group can sometimes be detrimental to an individual's career. The other group or individual may be more successful based on that help, and eventually in the fight for limited resources within the corporation, may indeed go against the original good Samaritan. It has been stated that a multi-party collaboration, therefore, benefits all parties concerned must therefore be established, motivated, and

¹³ <http://www.gnu.org/software/mailman/mailman.html>

rewarded. Individuals must be able to understand where its in their and their group's best interest to collaborate and what actions to avoid that can potentially lead to giving strength to their corporate competition.

3.0 Research Approach

In this section we describe our approach for understanding the adoption and use of Progressive Open Source within HP. We describe what we seek to explore, and how we intend to collect the empirical data.

The research focuses on collaboration as a mean to improve and speed up innovation in software engineering. We seek to explore what collaboration is all about in practice, i.e., how individual software developers perceive collaboration and collaborative efforts, what it is and what the results are. Organizing for innovation is crucial for companies that are forced to quickly adapt to changing competition, markets and technologies (Dougherty 1996, Hage 1988, Jelinek and Schoonhoven 1990, Zahra and Covin 1995). Despite this awareness, activities that are directed towards changing established structures in the organization and encourage innovation continues to be challenging for the organizations. Collaboration in itself has an ambiguous connotation. It embraces positive and encouraging aspects such as the willingness to work jointly on intellectual endeavors. But also less positive aspects such as control, visibility and giving things away, possibly to an enemy (translated into e.g. a business setting: a competitor, a competing colleague, or a workplace setting). This inherent tension must be mitigated when organizing communities of developers within a corporate setting. Other tensions are obviously the market-technology linking, i.e., keeping operations efficient inside while maintaining the relationship with third parties and customers. Organizing for creative problem solving highlights the balance between effective reuse of already invented ideas while at the same time making room for new ideas. Collaboration also needs to balance between the individual and the collective, e.g., in regard to developing a commitment to innovation, shared responsibilities and accountability. We believe that ethnographic studies of technology-mediated collaborative work can provide important insights in the social interaction, e.g., how work is coordinated and how unexpected events are handled and co-managed (Engeström & Middleton 1998, Heath & Luff 1991, Goodwin & Goodwin 1996, Suchman & Trigg 1991, Bentley et al 1992). We want to know more about what the users of CDP and Corporate Source actually are doing, and also what they think about what they are doing (Gantt & Nardi 1992). Ethnography is suitable for studies of collaborative work, especially when studying large communities of users.

*Our nature lies in movement: complete calm is death.*¹⁴

Czarniawska and Sevón advocate that researchers on the field have to understand ongoing changes through observing the events as they are unraveling, e.g., by listening to the testimonies of those affected (1996). It is important to study changes since “only what moves is visible” (1996:2). In times of changes old things and habits are put to sleep and new ways are brought to life, they are *constructed*. We try to collect ‘little narratives’ in search of understanding, striving to capture the richness of the meaning of collaborative action, evoking the associations with movement as well as association (Czarniawska & Sevón 1996). At the end of our project we wish to be able to say something about how local action emerges and how it becomes institutionalized within a large corporation. The only way to do this is to talk to those who know, i.e., the organizational actors. This means that we are

¹⁴ Blaise Pascal, cited in Bruce Chatwin 1988:183

looking at how action (behavior and talk, e.g.) is repeated to, in the end, become something taken for granted. And to look at what this 'taken for granted' includes.

The goal of the research is to create a multi-viewpoint understanding of the new software paradigm, Progressive Open Source. We set out to do so by combining different research methodologies and theories. The research design encompasses participatory observations, semi-structured interviews with users of the Progressive Open Source, as well as quantitative analysis of data collected on the use of the collaborative systems, i.e., how the technology is utilized for a variety of collaboration, and how the community grows and proliferates. We will conduct in-depth interviews with selected users of Corporate Source and the Collaborative Development Program. The interviews will cover the spectrum of managers, Human Resource Officers and engineers. For this part of the investigation, we have developed a framework of questions. The investigation aims at providing insights on the behavior of the adopters of the new methodologies offered by Progressive Open Source within Hewlett-Packard. The user statistics are reviewed in order to define and categorize the adopters. We will focus on particular projects including third party collaborators. In order to reduce biases and ensure confidentiality, we protect the integrity of the interviewees, i.e., they participate on an anonymous basis.

By way of introduction it is important to understand what the developers perceive as innovative about the projects facilitated by Progressive Open Source. The general and perhaps most important aspect to be studied within this particular design is: to get peoples' ideas about what constitutes an innovative process of collaboration aimed at innovative products. The study is expected to contribute to a comprehensive understanding of if and how the Progressive Open Source paradigm is understood as a useful collaboration opportunity for the HP development community. In addition, we address similar questions to managers and Human Resource officers to grasp how they perceive Corporate Source and the Collaborative Development Program and its importance for innovation. The study addresses eight main areas, even though the interview design allows deviations in terms of addressing the issues that are perceived as important by the users. We now briefly describe the areas and what we want to explore.

Furthermore, what the impact is on their day-to-day work. We wish to know more about the advantages respective disadvantages from an individual as well as a team perspective.

We are keen to collect the success stories that have already occurred as a result of using Progressive Open Source. We hope to know more about the characteristics of innovation e.g., the relative advantage, the compatibility, complexity, trialability and observability (Rogers 1962, 1995). Another area of interest is motivation and how collaborative environments may be encouraged and how individuals and teams are to receive sufficient rewards and recognitions for their contributions. We draw on the work of Mauss (1924/1990) to try to understand more about gift-cultures and how people are motivated to make contributions to a community. Existing reward structures within the company may conflict with the notion of the gift economy. Finally, we address issues directly related to third party interaction e.g., communication, support, security and trust, and most importantly strategy and issues of corporate philosophy.

In addition to the above-mentioned interviews, we collect stories from the field in order to grasp the context of Progressive Open Source. We want to understand from where the idea has arrived and where it has been materialized, and why it is perceived as important by a large corporation to adopt methodologies and ideas from the Open Source model of developing Software. The research is best described as being inter-disciplinary, since the methodologies stem out of the software-engineering paradigm as well as from organizational studies. The research project is designed and undertaken as

part of a doctoral thesis in business studies in close conjunction with researchers on software engineering within a corporate entity, Hewlett-Packard Labs in California.

4.0 Expected Results

Openness is part of HP culture. Early on, the founders of HP, Bill Hewlett and David Packard, understood and promoted an “open-door” policy in HP. The basic theme was that there are no doors on anyone's office, and anyone can communicate with anyone, regardless of hierarchical positioning (Packard 1995). Clearly, collaboration is just as much an issue of culture and tradition as it is technology. The Progressive Open Source has evolved out of a context rich of precursors and prototypes of collaborative development models utilized within the company. We wish to highlight one of them that exhibit most of the traits acknowledged in the collaborative development model applied today, the Owen Firmware Cooperative model (Toft, Coleman and Ohta, 2000).

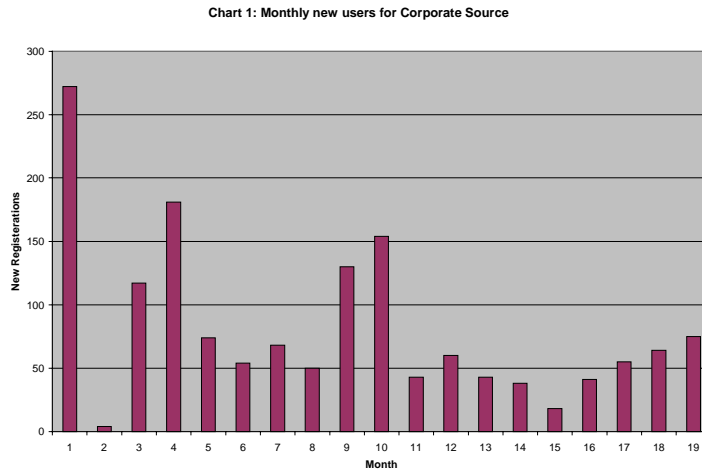
The Owen Firmware Cooperative project stems out of an incessant business constraint, leading to an emergent need to do more, preferably better, with the same resources and investments. Before the Owen project, a few related products were developed serially. In order to meet business goals (time-to-market) several related Owen products were developed in parallel through enhanced reuse and development efficiency. A technical approach was adopted in order to create a firmware-architecture and to define and adopt common tools in order to facilitate code sharing and leverage. In fact, it was acknowledged that the code in itself is valuable and important, not only for the product being developed, but for future products too. In order for other developers to be able to pick up the code, they must be able to understand and make use of it with ease. This was feasible through well-defined interfaces, i.e., interface descriptions represented in a standard form. Moreover, it was stressed that the components must be loosely coupled, and the establishment of clear principles and guidelines for using and extending the architecture. Above all, cooperation and collaboration was guiding the work.

The Owen project also indicates that it is of great importance to establish and build strong working relationships and trust, especially when the work teams are globally distributed. However, it was also acknowledged that contribution to the community is far more important than distribution of work within a global community. A set of operating principles were confirmed in order to stipulate the important features e.g., leverage of existing knowledge and progress, critical but simple guidelines, the importance of contributions, project team autonomy and empowerment, and a focus on maximizing utility of the contributions.

The Owen project resembles Open Source Projects in several ways: its emphasis on empowerment, and stress on the importance of maximizing the utility brought back to the community of users. The management structure differs from the traditional, e.g., power and influence in the development process, is guided by competence, rather than formal status (meritocracy). The code in itself is regarded as valuable, not only the end product. And finally, the projects carefully monitor and manage the evolution of interfaces.

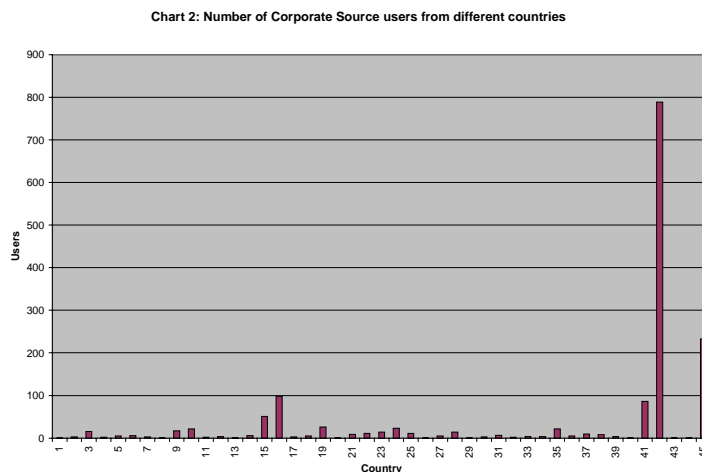
The Owen project was a big success: prior to the Owen cooperative, All-in-One devices were introduced 18 months after the equivalent Single Function Printer, today All-in-One devices are co-introduced with the equivalent Single Function printer. The success of Owen gives us assurance that given the right organization and supporting infrastructure, Progressive Open Source will be successful in HP. The work reported in this paper is a step towards addressing the core organizational issues to be addressed for a successive implementation of Progressive Open Source.

Tentative results indicate that Progressive Open Source and its precursors facilitate collaborative efforts leading to improved conditions for software development, re-use and innovation within Hewlett-Packard. At this point the efforts are gaining global reach. User statistics indicate that Progressive Open source operates in at least 55 countries, in spite of lacking central project headquarters and national origins. New users and projects are being added to both the Corporate Source and CDP programs at a steady rate. For example, the following graph shows the addition of new users to the Corporate Source service on a monthly basis. However, past experiences indicates



that complex tasks and distributed work have been dependent upon planning, coordination as well as supervision. Moreover, common repositories of information, in this case source code and other related information, tends to become obsolete over time, and hard to make use of as content and users increase. The challenge is obviously related to how to stay focused on goals, schedules and quality by organizing the information in such a way that increased usage and an incrementally growing community not stall the benefits of sharing a common repository and a common tool-set (e.g., Brooks 1975, Moon & Sproull 2000). There are huge potential benefits of sharing and opening up, since knowledge doesn't disappear when used, i.e., in theory we may expect a net gain in knowledge through re-use. The pooling of resources enables sharing and leveraging existing knowledge facilitated by common standards, such as work processes and tools (Arthur 1994). However, existing technology favors knowledge transfer of established ideas, i.e., implementation of mature ideas rather than spurring the idea process that needs to be in place first. This indicates that projects most likely have to have reached a certain level of maturity before opening up to a wider community, whether internally within the traditional organizational boundaries, or to an external group of third party developers.

Chart 2 shows the number of registered users from different countries:



5.0 Summary

In this paper we described the use of Open Source collaboration style within a large company. We described an innovative program, called Progressive Open Source, that has been defined at HP to leverage the collaboration benefits of Open Source development and use methods. The program currently has a global reach, with no project headquarters, no national origin, and operates in at least 45 countries. Early experiences indicate increased location independence, i.e., developers can remain physically close to customers, while working wherever is convenient. Developers also are easily re-deployed between different projects. Progressive Open Source is also an electronic Knowledge Network that facilitates the decentralized organization. The computer networks support and supply the actors with corporate information, that is available online, worldwide, for immediate access. Early statistics indicate time compression, i.e., development can be maintained and supported on a 24 by 7 basis, i.e., towards a 24-hour workday. Potentially, software projects never sleep – they follow the sun. Programmers working in different time zones ship code back and forth to keep the development process moving. The research reported in this paper constructs a theoretical framework to understand and document this new collaboration paradigm at HP.

References

- Alderman, Johan, (2001), *Sonic boom: Napster, MP3 and the new pioneers of music*, London: Fourth Estate
- Arthur, W. Brian, (1994), *Increasing returns and path dependence in the economy*, Ann Arbor University of Michigan Press
- Barbrook, Richard, (1998), *The High-Tech Gift Economy*, First Monday 3.12, http://www.firstmonday.dk/issues/issue3_12/barbrook/index.html
- Bentley, R., Hughes, J.,A., Randall, D., Rodden, T., Sawyer, P., Sommerville, I., Shapiro, D., (1992) *Ethnographically-informed systems design for air traffic control*. In proceeding of the Conference on Computer Supported Cooperative Work, CSCW 92, ACM, New York, pp. 123-129.
- Brooks, Frederick, P., (1975), *The Mythical Man-Month – Essays on Software Engineering*, Addison-Wesley Publishing Company, California.
- Brown, J., S., & Duguid, P., (1991), *Organizational Learning and Communities-of-Practice: Towards a Unified View of Working, Learning, and Innovation*, *Organizational Science* 2 (1): 40-570
- Castells, Manuel, (2000), *Materials for an exploratory theory of the network society*, *British Journal of Sociology*, Vol. No. 51, January/March 2000, pp. 5-24, ISSN 0007 1315, London School of Economics.
- Chaiklin, S., & Lave, J. (eds), (1993), *Understanding Practice: Perspectives on activity and context*, Cambridge: Cambridge University Press
- Chatwin, B, (1988), *The Songlines*, London: Picador
- Christensen, Clayton, M., (1997), *The innovator's dilemma when new technologies cause great firms to fail*, Boston, Mass. Harvard Business School
- Clarke, I., Sandberg, O., Wiley, B., & Hong, T.,W., (1999), *Freenet: A Distributed Anonymous Information Storage and Retrieval System*. In ICSI Workshop on Design Issues in Anonymity and Unobservability
- Czarniawska, Barbara & Sevón, Guje, (1996), *Translating Organizational Change*, Berlin de Gruyter
- DiBona, Chris, Ockman, Sam & Stone, Mark, (1999), *Open Sources: Voices from the Open Source Revolution*, O'Reilly
- Dinkelacker, Jamie, Garg, Pankaj, K., Miller, Rob, Nelson, Dean, (2001), *Progressive Open Source*, HP Laboratories Palo Alto, HPL-2001-233, Hewlett-Packard Company. To appear in the *Proceedings of the 24th International Conference on Software Engineering* Buenos Aires, Argentina, May 2002.

Dougherty, Deborah, (1996), Organizing for Innovation, in Handbook of Organization Studies, edited by Clegg, Stewart, R., Hardy, Cynthia & Nord, Walter, R., Sage Publications

Engeström, Yrjö & Middleton, David, (1998), Cognition and Communication at Work, Cambridge University Press

Fogel, Karl, 1999, Open Source Development with CVS, Cariolis, The United States

Gantt, Michelle & Nardi, Bonnie, A., (1992), Gardeners and Gurus: Patterns of Cooperation among CAD users, Hewlett-Packard Laboratories, Human-Computer Interaction Department, Palo Alto, CA, ACM 0-89791-513-5/92/0005-0107

Goewin, C., & Goodwin, M., J., (1996) Formulating Planes: Seeing as a Situated Activity. In Middleton, D & Engeström, Y (eds.) Communication and Cognition at Work: Cambridge University Press, Cambridge pp. 61-95

Hage, J. (ed), (1988), Futures of Organizations, Lexington, MA: Lexington Books

Heath, C., & Luff, P., (1991), Collaborative Activity and Technological Design: Task Coordination in London Underground Control Rooms, In Proceedings of the Second European Conference on Computer-Supported Cooperative Work, Kluwer, Dordrecht, pp: 65-80

Jelinek, M., & Schoonhoven, C., (1990), The Innovation Marathon: Lessons from High Technology Firms, Oxford: Basil Blackwell

Kuhn, Thomas (1970), The Structure of Scientific Revolutions, Second Edition, Enlarged, The University of Chicago Press, Chicago, IL.

LaPlante, Alice & Seidner, Rich. (1999), Playing for Profit: How Digital Entertainment Is Making Big Business Out of Child's play, New York: Wiley

Latour, Bruno, (1987), Science in Action, Cambridge, MA: Harvard University Press

Latour, Bruno, (1999), Pandora's hope: essays on the reality of science studies, Cambridge: Harvard University Press

Lave, J. & Wenger, E., (1991), Situated learning: Legitimate peripheral participation, Cambridge: Cambridge University Press

Lessig, Lawrence, (1999), Code and other Laws of Cyberspace, Basic Books

March, J. G., (1994), A Primer in Decision Making - How Decisions Happen, New York: The Free Press

Mauss, Marcel, (1924/1990), The Gift: The Form and Reason for Exchange in Archaic Societies, (transl. W.D. Halls), New York: W.W. Norton

Mockus, Audris, Fielding, Roy, T., & Herbleb, James, (2000) A Case Study of Open Source Software Development: The Apache Server, Bell Labs, 263 Shuman Blvd., Naperville, IL 60566 USA, Proceeding of the 22nd International Conference on Software Engineering, Limerick, Ireland, June 2000

Moody, Glyn, (2001), Rebel Code, Inside Linux and the Open Source revolution, Perseus Publishing, Cambridge, Massachusetts

Moon, Jae Yun & Sproull, Lee, (2000) Essence of Distributed Work: The Case of the Linux Kernel, <http://firstmonday.org/issues/issue5-11/moon/index.html>

Mukerji, Chandra, (1998), The Collective Construction of Scientific Genius, In Cognition and Communication at Work, Edited by Yrjö Engeström and David Middleton, Cambridge University Press

Neumann, Peter, g., (2000), Robust Nonproprietary Software, IEEE Symposium on Security and Privacy, Oakland, CA May 15-17

Nonaka, Ikujiro, (1994), A dynamic theory of organizational knowledge creation, Organization Science, 5(1):14-37

Packard, David, (1995), The HP Way, How Bill Hewlett and I Built Our Company, Harper Collins, New York

Quinn, James Brian, (1992), Intelligent Enterprise, The Free Press

Raymond, Eric, S., (2001), The Cathedral and the Bazaar: Revised Edition, O'Reilly, CA

Rehn, Alf, (2001), *Electronic Potlatch, A study on new technologies and primitive economic behaviors*, KTH INDEK, Doctoral Dissertation

Rogers, Everett, M., (1962, 1995), *Diffusion of Innovations*, New York, Free Press

Rose, Nick & Buchanan, Nichola, (2001), *A&M Records v Napster: the case and its consequences for copyright owners in the music industry*, *Copyright world* 2001:112, 13-15

Rushkoff, Douglas, (1999), *Playing the Future: What we Can Learn from the Digital Kids*, New York: Riverhead Books

Sevón, Guje and Kreiner, (1998), *Constructing R&D Collaboration*, Copenhagen Business School Press

Smith, M., A. & Kollock, P., (eds) *Communities in Cyberspace*, London: Routledge, 1999

Suchman, L. & Trigg, R., (1991), *Understanding practice: video as a medium for reflection and design*, In Greenbaum, J. & Kyng, M. (eds.), *Design at Work: Lawrence Erlbaum Associates*, Hillsdale, NJ, pp. 65-90

Tangkjaer, Christian, *Oresund as an Open House Strategy by Invitation*,

Toft, Peter, Coleman, Derek, and Ohta, Joni, (2000), "A Cooperative Model for Cross-Divisional Product Development for a Software Product Line," in Patrick Donohoe, Ed., *Software Product Lines: Experiences and Research Directions*, Kluwer Academic Publishers

Tuomi, Illka, (2001) *Internet, Innovation, and Open Source: Actors in the Network*, <http://firstmonday.org/issues/issue6-1/-tuomi/index/html>

Victor, B. and Boynton, A., C., (1998) *Invented here: Maximizing your organization's internal growth and profitability*, Boston: Harvard Business School Press

Wayner, P., (2001), *Free for All: How Linux and the Free Software Movement Undercut the High-Tech Titans*, New York, Harper Business

Zahra, S. & Covin, J., (1995), *Contextual influences on the corporate entrepreneurship-performance relationship: a longitudinal analysis*, *Journal of Business Venturing*, 10: 43-58

Zeleny, Milan, (1989), *Knowledge as a new form of capital: Division and reintegration of Knowledge*, *Human Systems Management*, 8 (1): 45-58