Joel E. Fischer

Identifying, Visualizing and Supporting Social Networks for Collaborative Work in a CSCW-System

December 17, 2007

Selbststaendigkeitserklaerung

Hiermit erkläre ich, dass ich die vorliegende Masterarbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Ort, Datum und Unterschrift

Danksagung

Ich möchte meine Dankbarkeit all denen ausdrücken, die diese Masterarbeit möglich gemacht haben. Zu allererst möchte ich meinem Betreuer Peter Mambrey am Fraunhofer Institut für Angewandte Informationstechnik danken, der mich in mehr als drei Jahren gemeinsamer Arbeit an seinen Erfahrungen hat teilhaben lassen, durch den mir Einblicke in dieses interessante Forschungsfeld möglich waren, der mich immer wieder ermutigt hat, mich in Projekarbeit zu beteiligen, die das übliche Maß studentischer Hilfskräfte überschreitet. Ohne seinen Horizont, seine Erfahrungen und reichhaltigen Gespräche und Anmerkungen zu meinem Thema wäre diese Arbeit so nicht möglich gewesen.

Kein geringerer Dank gilt meinem Betreuer Wolfgang Prinz, der mir immer wieder wertvolle Anmerkungen zu meiner Arbeit, insbesondere zu der technischen Realisierung gab, der mich immer wieder auf verwandte aktuelle Arbeiten im Forschungsbereich aufmerksam machte und mich stets motivierte und ermutigte. Ich danke auch den Kollegen der Forschungsgruppe CSCW am Fraunhofer FIT für fruchtbare Diskussionen, Hinweise und aufmunternde Worte. Insbesondere danke ich Rudolf Ruland, dessen Expertise und Hilfe mir bei der Integration in BSCW sehr geholfen hat.

Dankbar bin ich Alistair Hampshire für das Korrekturlesen der Arbeit.

Großer Dank gilt meinen Eltern, Christine Ewert und Wolfram Fischer und meiner restlichen Familie für die Unterstützung jeglicher Art in meiner Studienzeit. Ohne ihre Unterstützung hätte ich mich nicht so auf Studium und die Abschlussarbeit konzentrieren können.

Ich bin meiner Freundin Daniela sehr dankbar für ihre Geduld, Verständnis und Rücksichtnahme in der Zeit des Schreibens. Sie hat mich auch in schweren Zeiten immer wieder motiviert, stets zu mir gehalten und mir Kraft geschenkt.

Joel Ewert Fischer
Duisburg, im Dezember 2007

Contents

List of Figures

Part I

Foundations

1

Introduction

Throughout the past decades, the diminishment of the primary and secondary sector of the economy in the so-called developed economies went hand in hand with the growth of the tertiary sector[1], often alongside with the localization of this development in a post-industrial or information society. The activities in tertiary sector such as finance, technology and services are characterized by knowledge-intense and information dependent processes. With the advent of technical communication networks such as the internet, the possibilities for transmitting information to support knowledge-intense work arose. As today's work practices become increasingly fragmented and specialized, as organizations grow to span geographic distances, the support of work by means of Information- and Communication Technologies (ICT) has become a crucial and inseperable practice.

Due to these developments, a field that aims at supporting cooperative work by means of ICT emerged in the early 80's called Computer-Supported Cooperative Work (CSCW). It is essentially a research field that investigates the nature of cooperative work and chances of supporting work with ICT, but also a pragmatic supportive practice that designs, implements, tests and sells supportive systems. The systems that support cooperative work have many names, one of the widespread names is groupware, by virtue of its goal to support cooperating groups of people. For the support of cooperative work with ICT to be successful, several concepts of assisting the work's essential processes have been researched. The support of the individual's awareness of people, artefacts and activities is considered to be a crucial concept, sometimes even the essential process [Neale et al., 2004] that must be supported in cooperative work.

Members of cooperating groups can be seen as being interconnected in a social network, alongside with the employed artefacts of work. The management research literature has long discovered the impact of social networks

[1] see for example the "Key Indicators of the Labour Market Programm" by the ILO, available at: `http://www.ilo.org/kilm`

on effectiveness and efficiency of productivity [Cross and Parker, 2004]. The technique of Social Network Analysis (SNA) both visualizes social networks and studies the network's topology to discover the centrality and interconnectedness of its elements and other traits of networks. This thesis claims that awareness supporting functionality of a groupware can be enhanced by SNA techniques to visualize networks that underlie cooperative work and to support everyday activities of cooperative work.

On the basis of an examination of relevant background of CSCW research, related work and an empirical requirements analysis, add-on awareness supporting functionality for an existing groupware (BSCW) is developed in form of a visualization tool. The approach to concept and design of the visualization tool is clarified, its implementation is depicted and the tool is evaluated in a qualitative, ethnographic approach.

1.1 Problem Definition and Objectives

The groupware that the visualization tool is developed for falls into the category of a "shared application[s] supporting the participants' interaction with shared work objects" [Dix et al., 2003, pg. 690], whereby the participants work in a locally distributed, asynchronous manner. The individual awareness of the other member's work therefore is mediated by the artefacts of work and the events in the shared workspace. Other than face-to-face or eMail communication, this type of communication is indirect and therefore it is referred to as *communication through the artefact* [ibid., pg. 690]. This type of indirect, mediated communication is often the only source of awareness of the locally distributed co-workers. Thus, the functions supporting awareness need to be carefully designed. Until now, there are several functions supporting the awareness of the work done by others in the workspace, which will be introduced further on in this thesis.

Awareness support does not only aim at identifying activities by a member's colleague [Pankoke-Babatz et al., 2004], but also the member's own position toward the colleagues in the network. The generic notification of the actors' activities in workspaces does not rate the importance of an activity by a certain actor for the individual. The relevance of an activity for an individual is often appointed by the interrelation of that activity with the individual's activity. The identification of relevant information is crucial to evade an overflow of information, this is also true for awareness information. However, the contextual interrelatedness of the workspace members' activities with the individual activity is mostly unaccounted for in awareness supporting functionality to this date. It is the nature of interrelatedness that affects the relevance of an activity for an individual. In order to assess the individual relevance of an actor's activity, it is essential for the individual to be able to rate that activity's relation to the individual's work. A network view onto the activity space affords insights on the interrelatedness by allowing the address

of questions such as: Who else is contributing to the shared workspace and how does this relate to my activities? Who are the actors that I cooperate with more closely and more frequently than with others? Who has common interests with me, who shares the same expertise? How did the workspace evolve chronologically over time? Being aware of what the others do and when they do it and the interpersonal relationships that emerge out of the activities in the network help to become aware of one's own position in the working context.

However, there is no function to this day that provides the individual member of a workspace with a comprehensive overview of the characteristics of his network: the co-workers and their interrelating activities on the shared artefacts of work. It is hypothesised that the individual awareness described above cannot be achieved easily with the current awareness functionality implemented in the shared workspace system, because the current view onto BSCW with its hierarchical workspace structure rather emphasizes the docuements than the people that cooperate within the workspace by using the documents. The hypothesises will be tested in an empirical requirements analysis.

The aim of this work is to enhance the awareness of the individual's working network by providing the member with user-definable, visually explorable network-perspectives onto the activities of cooperation in the shared workspace and the interpersonal working relationships that emerge from the activities.

This thesis will investigate the impact of the provision of a dynamic visualization on the individual's awareness of his working network. Does the approach of visualization mirror the individual's subjective notion of his personal working network? Do the afforded means of interaction foster meaning-making and the application of the tool to support the individual's tasks that arise in daily cooperative work? Instead of prescribing the tasks to be supported by the tool as part of the design, it will be shown that use cases the tool can be employed in and tasks that are supported by the tool emerge out of the ethnographic evaluation of the tool.

1.2 Structure of the Thesis

The overall necessity for the development of a visualization tool of the co-workers network is justified and motivated by the exploration of the background research (see chapter 2) and related work (see chapter 3). Here, knowledge-intense work will be introduced as a basic characteristic of modern society and the notion of socio-technical systems will help to set the scene for exploring actors and activities of cooperative work. CSCW and its basic concepts, especially groupware and awareness are englightened to provide a basis for the design of an awareness supporting tool.

The development approach (chapter 4) introduces the used technology to realize the design and highlights the structure of the reflective-creative procedure of requirements analysis, design, implementation and evaluation of the visualization tool and thereby structures Part II of this thesis.

A requirements analysis (chapter 5) among two distinct user groups will help to arrive at an understanding how users perceive the current view on BSCW to test if the author is correct about the starting assumptions. The analysis also aims at finding out about the character of cooperative work in BSCW and to reveal hypothetic chances and challenges of the visualization tool.

The concept and design of the visualization tool (chapter 6) describes the design rationale: possible scenarios of usage of the tool, the visual and interactive elements of the tool and the approach to using SNA techniques to extract mediated interpersonal networks and to deduct common interest for the individual.

The implementation of the prototypical visualization tool is specified in chapter 7. The integration approach of the developed tool into the existing groupware BSCW is depicted, and the implementation of some selected, important functionality of the tool is detailed.

In addition to evaluating the tool's usability, the evaluation (chapter 8) discovers use cases and tasks the tool can be used to support that emerged out of user interaction with the tool during the evaluation. The findings shed light onto research questions and reveal promising scenarios where the tool can be successfully applied.

Implications of the insights of the evaluation are of general interest for awareness support in CSCW and are discussed in the conclusions of this thesis (chapter 9). This thesis is concluded with an outlook on future work that builds on the insights of this thesis in chapter 10.

2

Background

This chapter aims at stating the theoretical background of this thesis upon which the development depicted in Part II resides. It introduces knowledge-intense work as a basic characteristic of modern society. The concepts of knowledge and learning are elicited to arrive at an understanding of its actors and processes. On this basis, the relevance of Computer-Supported Cooperative Work (CSCW) as a supportive practice is developed. Basic concepts of CSCW, overview and concepts of systems that support cooperative work, awareness as a crucial concept of CSCW and principles of information visualization further motivate the contribution of this thesis. As the development integrates functionality based on Social Network Analysis, this research field and relevant notions for this thesis are introduced in the final section of this chapter.

2.1 Knowledge, Learning and Activities in Socio-Technical Systems

Today, a responsible and efficient handling of knowledge is seen as a prerequesite to deal with this key resource to all sectors of modern society due to "globalization, keener world-wide competition, shortening in development cycles for products, demographic shifts in the world's industrial countries, reduction in the half-life of knowledge" [Meier et al., 2001, pg. 14]. The slogan *Knowledge is Power* is today as pertinent as never before. Scientific and technological innovation rely heavily on knowledge; governments, institutions and organizations largely base their decisions and thus their behavior as a whole on information. The term *Knowledge Society* embraces the value of information and knowledge as well as its inherent challenges that are superimposed on the individuals of the society; namely disorientation through information overflow and the lack to filter out relevant information. Information and Communication Technology (ICT) set out to channel the flow of information so it will find its destination and to structure information and make carriers of knowledge

accessible and thereby facilitate processes of learning, information sharing and knowledge management. To provide a foundation for the development of systems that support cooperation, an understanding of knowledge and learning has to be developed. The subject of investigation is the socio-technical system – according to the notion inherent in this approach, its elements society and technology cannot be considered in isolation, they are mutually constitutive and interdependent [Ropohl, 1999]. The strength of the term socio-technical system lies in its flexibility: A socio-technical system can be an organization that consists of people that use technology to support their work, but it may also be a cross-organizational project team or a loosely coupled social network that employs computer-mediated communication. This section aims at identifying the key concepts of learning and knowledge to clarify the relevance and the scope of Computer-Supported Cooperative Work (CSCW).

2.1.1 Knowledge and Knowledge Management

Knowledge-intense work needs structured and explicit support of its processes. It is common sense that symbols, data and information can be generated, disseminated, duplicated and stored by means of ICT. As individual's perceive and interpret information in situated, contextual activities, bodily bound knowledge [Merleau-Ponty, 1966] is constructed in processes that incorporate negotiation, construction and reconstruction of meaning, association and sensation on the basis of their prior experiences. Knowledge comprises the set of skills that individuals use to solve problems – theoretical insights, practical heuristics and conventional, emotionally rated instuctions on how to act [Mambrey et al., 2003]. Processes of transformation through shared context and shared knowledge construction lead from individual to collective knowledge representation [Nonaka and Takeuchi, 1995]. The process of construction or development of knowledge is considered to be learning [Mambrey et al., 2003]. This process is acknowledged to exist for organizations as well [Argyris and Schön, 1999]. Organizational learning is considered to be the implicit process of change of the organization's *knowledge base*. The knowledge base consists of individual and collective knowledge resources amended by data and information underlying this knowledge; it is the organization's foundation upon which it executes its tasks [Argyris and Schön, 1999].

Knowledge management is understood to be an intervening practice that designs the organizational's *knowledge base* according to the organization's goals, thus making the implicit process of organizational learning explicit and accessible [Probst et al., 1999, pg. 46]. Individuals are crucial to knowledge management since they are capable of transforming information into knowledge and they are the central carriers of the organizational knowledge base. They produce and own immaterial economic resources. Collectives are a central element of the knowledge base, too: a collective's problem-solving abilities exceed the sum of individual skills of its members [Probst et al., 1999].

ICT can serve a key function in the process of knowledge management and knowledge sharing. [Probst et al., 1999] suggest six key processes of knowledge management; namely the identification, acquisition, development, distribution/sharing, usage and storage of knowledge. Information that underlies all knowledge work can be generated, stored, structured and made available for sharing or be disseminated by technological systems. Furthermore, ICT can support knowledge-intense work by identifying human carriers of knowledge and thus enable the usage of expertise that resides in humans. The discipline that employ ICT to support such knowledge and information-sharing, is known as CSCW. But before the basic concepts of CSCW are elicited, the actors and processes of knowledge-intense work will be characterized.

2.1.2 Groups and Social Cognition

As the scope of this section is to identify the actors and their processes that are relevant to CSCW, the individual level of information processing, cognition and behavior is not considered. Individuals do not carry all the knowledge that is required for an organization ro reach a goal, rather it is distributed among the members of the smallest unit of sociological interest: the group. According to Guzzo and Shea [1992, pg. 272] a group is a social system with the following characteristics:

- It is perceived as a whole by its members and by nonmembers that are accustomed with the group,
- its members are mutually interdependent,
- its members have different roles and responsibilities.

The relevance of groups for organizations is rising, a lot of activities that have been conducted by individuals in the past are being carried out by groups today [Guzzo and Shea, 1992]. The perception of the group as a coherent whole is based in the group culture. The approaches that deal with group culture have in common that they assume a common set of thoughts among the group members. This set of thoughts is comprised of knowledge about the group (e.g. rules and norms), knowledge about its members (e.g. on roles and skills) and knowledge about the work of the group [Meier et al., 2001].

The social cognitive approach deals with the processes of collection, storage and recall of information in a group. Larson and Christensen [1993] showed that cognitive processes of encoding, storage and recall can be observed and analyzed not only on the individual level, but on the group level as well. For instance, the process of encoding is related to the identification and interpretation of relevant information within the group. The authors furthermore show that groups that have innovation-oriented norms rather develop discursive group dynamics in which diversive opinions are raised than groups that strive for harmony and elude confrontation. Discursiveness facilitates the negotiation of goals and tasks in a group and raises the information value for the group's members.

A crucial advantage of a group over an individual is that the collective memory is superior to the individual memory. Cohen and Bacdayan [1994] show experimentally that procedural knowledge for a group to fulfil a task is stored between the group's members and argue that these "(...) organizational routines – multi-actor, interlocking, reciprocally-triggered sequences of action – are a major source of the reliability and speed of organizational performance" [Cohen and Bacdayan, 1994, pg. 554]. This form of memory has been identified as transactive memory: "A transactive memory system is a set of individual memory systems in combination with the communication that takes place between individuals" [Wegner, 1986, pg. 186].

The superiority of collective memory is often elicited by the process of information recall in the group: when a group recalls its knowledge, the members can correct each other and provide own experiences in the process of recall – this is a strong argument for a discursive group culture.

The usage of decentralized knowledge can only be successful if the members know about each other's expertise. A potential problem then is that the members must be able and willing to share their knowledge, a prerequesite for successful knowledge management is the existence of an organizational culture of trust and mutual respect. However, it is a shortcoming of social cognition that influence of culture, identity and situatedness are not considered. The socio-cognitive approach treats groups as closed systems. The fact that crucial organizational knowledge is not only stored between its members, but that the information grounding the knowledge is often formalized and distributed across a technical support system is skipped, as well as the fact that knowledge is often acquired from outside the organization.

2.1.3 Theory of Social Practice and Communities

Situated approaches to learning and knowledge emphasize the relational interdependence of agent, environment, activity, meaning, cognition, learning and knowledge [Lave and Wenger, 1991]. According to the creators of *theory of social practice* cognition and communication in and with the environment is embedded in the chronology of the continuous interaction. Correlation and meaning of situations are constantly produced, reproduced and changed. Participation in communities of practice is always based on the negotiation of meaning in the world. This implicates that understanding and experience are in constant interaction – they mutually constitute and determine each other.

The concept of communities of practice (CoP) has thankfully been adopted by those that try to find solutions for the support of communities. It has been introduced by Lave and Wenger [1991] as a concept of learning, where learning is characterized by a process that ranges from *legitimate peripheral participation* in a community of practice as a novice and evolves toward the center as engagement and complexity are rising. Learning becomes an integral and inseperable constituent of social practice. In situated learning, the learner no longer is seen as a recipient of factual knowledge but as an actor interacting

with his environment. The work is central in the paradigm shift away from the conception of learning as only information processing to a holistic view that includes situatedness, corporality and emotion.

Communities of Practice

Wenger [1998] introduces CoP as being anywhere, where people act. Every person participates in CoP, be it at home, at work, in the clique, in a party, in church or in backyards, be it directly or virtually. CoP do not exist through themselves, they develop in a larger context. Characteristics for a CoP are:

- *Mutual engagement.* Practice is not abstract. It exists because people engage in activities that are under constant negotiation.
- *Joint enterprise.* The joint enterprise keeps the community together. It is the result of a collective process of negotiation which reflects the complexity of the mutual engagement. It is defined by its participants' process to reach it. It is the negotiated reaction on the situation and is constituent of every participant. It is not only a set goal, it creates relationships of mutual responsibility among the participants that become and integral part of practice.
- *Shared repertoire.* Mutually created or adopted routines, words, tools, conventions, signs, symbols, artefacts, acts or concepts of the community. Meaning is assigned to the repertoire by negotiating within a constituating process of social interaction and reification.

Wenger sees the concept of identity as intertwined with the concept of CoP, mereley shifting the attention to the individual: "Focusing on identity, however, is not a change in topic but rather a shift in focus within the same general topic" [Wenger, 1998, pg. 114]. Negotiaton and acceptance of each other's roles is a prerequesite for participating in a CoP, and incidentally the identity of the member is negotiated. The identity of the participants is constantly negotiated, just as the meaning of a situation, by the interaction of participation and reification, identity is a continuous becoming, a neverending process. Considering teams, the creation of identity is not just an individualistic trait, the sense of belonging together, of perceiving the group as a whole largely depends on the individual's identification with her role.

Communities of Interest

Online communities are often much less closely-knit than educational or organizational communities. "A very basic definition of an online community could be a set of users who communicate using computer-mediated communication and have common interests, shared goals, and shared resources" [Lazar and Preece, 2002, pg. 129]. The main difference to CoP is here, that the members

of a community of interest do not pursue shared tasks and do not collaborate in the sense of CoP. As this thesis emphasizes computer-supported cooperative work that is characterized by teamwork as indicated above, loosely structured communities such as *Del.icio.us* or *MetaCafe* are not within this work's scope. However, Preece has contributed to the design of community support in a substantial way by stressing the importance of both usability and socialbility. Where usability is concerned with the front-end matters of community software, sociablility focuses on social interaction. Since computers mediate in human communication, human-computer interaction becomes an integral part of human-human interaction. Therefore, a close interrelationship between usability and sociability exists. Preece recommends that a successful online community caters for a blend of usable software and carefully crafted sociability as in social policies that support the community's purpose and are understandable, socially acceptable and practicable [Lazar and Preece, 2002, pg. 144].

2.1.4 Activities and Challenges of Cooperative Teamwork

Neale, Rosson and Carroll [2004] state that groups are not the same as teams. In their terminology, teams are more specialized, task-oriented groups that are characterized by working together to achieve common goals whereby the tasks and their coordination are interdependent. With an increasing complexity of work, the demand for more tightly coupled groups of people who carryout a task with common goals has arisen, along with an often locally distributed participation.

Goal-oriented teams work in collaboration to carry out "joint projects that are characterized by the need for communication, planning, coordianting tasks, monitoring project progress and cooperation" [Neale et al., 2004, pg. 113]. Project work is typically long-term and demands the participants to "establish and maintain an ongoing awareness of the other's actions, plans, goals and activities" [ibid., pg. 113]. The work involves an iterative process based on changing objectives and circumstances of planning, acting and assessing the project's state. The teams that work in such projects need to be supported in information sharing, scheduling, role taking, synchronization, and allocation of resource [ibid.].

A team's activities can be considered on different levels of abstractions. The notion of *activity space* describes a bounded space that consists of people, artefacts, processes and activities which are undertaken as a result of events [Benyon, 1998, pg. 708]. People in this space undertake activities by using artefacts, thus, activity in this notion is seen to always be mediated by artefacts. Processes are the activities that make use of the artefacts within the system and that are triggered by a person's physical activity [Benyon, 1998]. This system-theoretical view has some parallel to the perspectives than can be taken on to describe activities in groupware, even though activity here does not distinguish between the user's physical actions and the processes that are

triggered on the artefacts of work. For simplicity, activity is subsumed to be an action that changes the state of an artefact or adds a new one to the shared workspace. In BSCW, the technical description of activities consists of actors that perform certain activities at a certain time (events) with shared artefacts [Seeling et al., 2007].

The design of groupware integrates the analysis of tasks to be accomplished in the collaborative environment and different types of task analysis methods are prevalent in the literature. However, such methods as *hierarchichal task analysis*, which decomposes a user's goal into the granular steps neccessary to reach the goal are appropriate mostly for well-defined single-user interface activities. Mutli-user activities that groupware aims to support are rather ill-structured [Neale et al., 2004], and involve communication and collaboration across different media channels [Budweg et al., 2006]. Nevertheless, with respect to the characteristics of groups, communities of practice and teams, some frequently occuring activities can be identified.

As a team's structure evolves and changes during time, the team deals with fluctuation of its members. While members leave the team or the organization, a groupware can support the process of "handover" of the former member's work. New members are challenged to "catch up" with the substantive work of the group. The role of a member may also evolve; studies of joint authoring showed that the roles such as author, co-author and commentator change frequently throughout the lifetime of a document [Dix et al., 2003, pg. 505]. This implicates that if a groupware employs a role concept to manage access rights it must be flexible toward this kind of work pattern. Those tasks arise frequently in group work and should be considered by designers of groupware.

From the ideas of distributed cognition it can be concluded that any artefact that is mutually created (e.g. drawings on a whiteboard in a meeting) does not only serve the purpose of communication, it can be seen as a concrete embodiment of group knowledge [Dix et al., 2003, pg. 507]. A groupware that is designed with insight of this fact, needs to provide a structure where mutually created artefacts can be stored and retrieved easily at a later time, which demands the groupware to provide flexible structure and efficient search methods.

There are many more examples of activities like the above that teams need to be supported in, but instead of providing an exhaustive list of granular activities, I will now attempt to identify the main processes within cooperation. Many attempts have been made to depict the activities that underlie cooperative work. Unfortunately, hitherto there is no consensus in the literature about the main activities and especially their interrelatedness. Here, I will follow the categorization of teamwork according to the level of interaction between group members as suggested by Borghoff and Schlichter [2000] and elaborated by Neale et al. [2004]. Borghoff and Schlichter [2000, pg. 110] identify an increasing degree of group communication from informaion to coordination to collaboration to cooperation. Neale et al. [2004] elaborate this notion to a multifaceted concept that includes aspects of work coupling and the demand

for communication needed. The strength of their model is that it includes contextual factors, which caters for the situated nature of human activity described above. In addition, the authors state "Perhaps the core challenge for CSCW systems is providing effective support for activity awareness" and furthermore argue that "If the proper levels of communication and coordination are supported, groups achieve common ground and acquire activity awareness critical for effective group functioning" [Neale et al., 2004, pg. 115]. Figure 2.1 shows the authors' model with levels of work coupling in relation to a rising demand of communication and coordination based on contextual factors. The overall goals are to achieve common ground and activity awareness. The elements of the model will be discussed in the following.

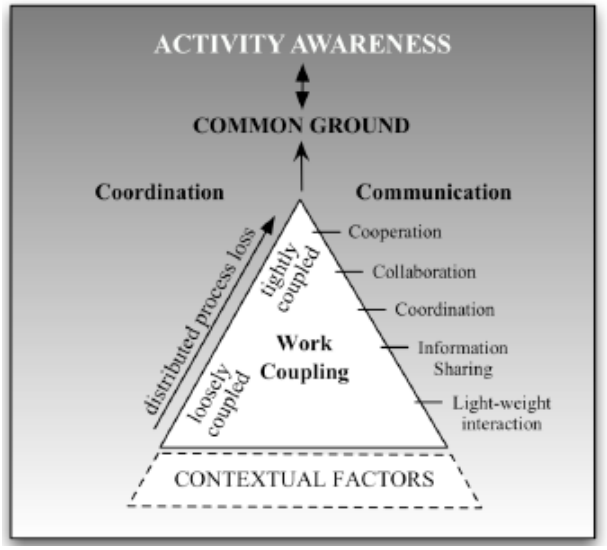

Fig. 2.1. Activities of cooperative work with the goals to achieve common ground and activity awarenes. Adopted from Neale et al. [2004, pg. 115]

.

Contextual Factors

Contextual factors situate ongoing activites, and their understanding is crucial to make sense of the activities. As we have already learned, participation in communities of practice is an ongoing process of the negotiation of meaning and identity (see section 2.1.3). Obviously, co-located groups that rely on face-to-face communication have advantages over distributed groups in this process as they can make meaning based on rich interaction including nonverbal and paraverbal communication and the subtleties of speech acts. In everyday work

of distributed teams, it is neccessary to have contextual information on other people, their roles, tasks and backgrounds. Interaction is situated and always involves not only syntactic and semantic information exchange on the content level, but a pragmatic relational level [Watzlawick et al., 1969]. This implies that besides exchanging information, each communicative partner interprets and assesses his social relationship with his opposite. A system may provide contextual information and thereby may inhibit the creation of an unshared understanding of the work and the people. For instance, systems may support such questions that arise in the context of work: Who is present? What are they doing? What are the artifacts of interest? What is the people's educational background? Whom can I ask? What can I do? Questions of this kind are often addressed by developments that support awareness. Awareness and concepts of its support in shared virtual environments is discussed in section 2.2.2.

Work Coupling

Neale, Carroll and Rosson suggest that the degree to which members must communicate relates to the degree of work coupling. In addition to categorizing activities by degree of communication, their framework includes the gradation of work coupling from loosely coupled work that requires few interactions to tightly coupled work that is dependent on frequent and qualitative demanding communication to achieve highly interdependent tasks. Neale et al. [2004] identify five levels of work coupling, in increasing order:

- Light-weight interactions. Involves casual social interaction and communication about the work.
- Information sharing. Can be in a direct fashion such as by e-mail or as store-and-retrieve from a shared repository.
- Coordination. Requires members to coordinate both activities and communication, the content of work and the plan to carry it out. Involves activites such as planning, scheduling, workflow management and synchronization.
- Collaboration. This level of work involve group members who work toward a common goal. Members may be performing seperate tasks that are interdependent, but the work is still done individually. They share goals, tasks and try to maintain a high state of shared knowledge.
- Cooperation. Group members who cooperate have the highest level of work coupling. They share all of the above and common plans how to achieve the goals. In addition, many tasks are persued concurrently in face-to-face activities with a high degree of coordination.

Awareness and Common Ground

The concept of awareness and its support in groupware is considered to be a crucial concept of CSCW in its own right, therefore, it will be elaborated as a basic concept of CSCW in section 2.2.2.

While awareness in a psychological concept of how aware someone is of an individual, a group, a tasks or recent events, the concept of *common ground* extends awareness to include a qualitative dimension: common ground is the idea of a shared or joint awareness.

> "Common ground is the product of joint awareness or mutual knowledge, and grounding behaviors is the process of maintaining joint awareness" [Neale et al., 2004, pg. 117].

The notion of common ground has been adapted from Clark's work on verbal behavior, who states that common ground is "the mutual knowledge, beliefs, and assumptions shared by the speaker and addressees" [Clark et al., 1983, pg. 247]. Cooperative work involves the negotiation of a shared repertoire (see 2.1.3), groups have to update their common ground frequently and this process is referred to as *grounding*. The goal of grounding is to achieve a shared understanding, for people to find out what they have in common with the others.

To conclude the model depicted in figure 2.1, the level of work coupling identifies the degree of communication and coordination required. With rising complexity of the work, also the amount of grounding required to achieve common ground increases. Activity awareness is seen as a product, a process and a goal of cooperative work on all levels. The activities are situated and therefore influenced by the amount of shared context factors.

2.2 Basic Concepts of CSCW

As computer networks became widespread, the possibilities of computer technologies multiplied – in addition to single-user applications, it was now possible that users on different workstations connect to each other by means of their computer. As Dix et al. [2003] pinpoint, "one result was the emergence of collaboration between individuals via the computer – called computer-supported cooperative work, or CSCW" [ibid., pg. 177]. The term CSCW was coined by Greif and Cashman as a slogan for a workshop[1] and is generally used to describe the research area in which activities take place to support cooperation or as Borghoff and Schlichter [2000, pg. 92] phrase it, "CSCW refers to the theoretical foundations and methodology for teamwork and its computer support." Following this notion, CSCW has become an interdisciplinary research area, as its aim is to understand human interaction and problem solving and to design the best way to support interacting humans by means of computer technology.

Thus, CSCW incorporates insights of numerous disciplines such as industrial, organisational and cognitive psychology, e.g. ergonomics of workplace,

[1] according to Borghoff and Schlichter [2000, pg. 92] the workshop took place in MA, USA in 1984

individual and organisational learning and knowledge, perception, emotion and interaction of individuals with an application or with other individuals mediated by the computer. Social sciences look into group processes, intra- or intercultural differences and interdependencies of society and technology. Engineering includes several research areas that build systems on different levels to support CSCW, including mathematics, information sciences but also hardware-oriented disciplines. In addition, philosophical questions arise more often than expected during research. As this panopticum suggests, CSCW is a wide research area and as it is an applied field, it has many perspectives. This introduction concludes by following the definition of Wilson [1991]:

> "CSCW is a generic term which combines the understanding of the way people workd in groups with the enabling technologies of computer networking, and associated hardware, software, services and techniques."

This sections aims at identifying the basic concepts of CSCW. Since CSCW is a pragmatic discipline that aims at building beneficial, usable and supportive systems for groups that engage in common tasks with mutual goals, these systems get looked into in the following section on groupware. Classification of groupware, example systems and their application domains are attended to.

2.2.1 Groupware

The advent of groupware marks a change in emphasis in software from single-user interfaces that support problem solving to interfaces that enable or support human interaction [Ellis et al., 1991]. Many names for groupware are present throughout research literature which often mean the same, while some of them emphasize on certain aspects. For instance, collaborative learning environments (CLEs) aim explicitly at the support of computer-mediated learning and teaching. Shared workspaces or collaborative platforms are more general terms, wheras collaborative virtual/working environments (CVEs/CWEs) emphasize the aspect of providing an environment for user action; this term accomodates notions of situated action (see 2.1.3).

Here, the generally accepted and early used [Johansen, 1988] term groupware will be followed, but we keep in mind that groupware comprises systems that support different aspects or even single tasks that arise in cooperation. In a common generic definition groupware is defined as:

> "computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment" [Ellis et al., 1991, pg. 40].

Groupware refers to software systems supporting activities as pinpointed in section 2.1.4 on cooperative teamwork. To classify these systems, the classical time/space matrix [Ellis et al., 1991] that defines systems as to where and

Table 2.1. Time/Space Matrix classification of groupware according to Ellis et al. [1991]

Time/Space	**Co-located**	**Distributed**
Synchronous	Meeting room support	Video conferencing
Asynchronous	Electronic blackboard	Shared workspaces

when participants perform their cooperative work is introduced. According to the time/space matrix (see table 2.1) groupware can be conceived to assist co-located teams and groups that interact face-to-face or distributed teams whose members interact across space from geographically remote sites. Furthermore, a groupware can be conceived to support synchronous interaction within real-time or to assist asynchronous non-real-time interaction [Ellis et al., 1991].

The table shows the matrix and in its quadrants popular applications of the combinations of time and space. In a meeting room members of a team interact synchronously at the same place. They may use a plethora of applications and hardware devices to support their work. A meeting room equipped with workstations for each participants and a shared large screen or electronic whiteboard adheres to the principle of WYSIWIS[2].

Video conferencing is a popular application where participants interact synchronously but are locally distributed. Collaborative tools that support distributed synchronous authoring of texts (GROVE, [Ellis et al., 1991]) or visual languages (*FreeStyler*, [Hoppe and Gassner, 2002]) are examples of sophisticated tools of this type. In the popular domain, Instant Messaging systems, of which the latest applications such as Skype, MSN or ICQ support both audio and text and often even video, fall into this category as well.

Electronic support for asynchronous and co-located interaction is rarer than the other types, the sticky note is a popular "non-techie" example. Electronic blackboards or advertising columns represent groupware of this type.

Document repositories that enable asynchronous interaction via document exchange for locally remote participants are a classical example of this type of groupware. Shared workspaces and collaborative working environments belong to them, but they often feature functionality that support synchronous interaction and awareness as well. The most popular application of the internet supports asynchronous and distributed interaction: electronic mail. Despite one-to-many communication is possible, it is not seen as groupware by defintion because it lacks the interface to a shared environment.

[2] This acronym stands for "What you see is what I see" and refers to interfaces for which the shared context is guaranteed.

Classification of groupware according to the cooperative work framework

Different ways of classifying and grouping groupware comprehensively have been proposed. While Borghoff and Schlichter [2000] follow the classification according to the system's functionality as proposed by Ellis, Gibbs and Rein [1991], the approach to classify groupware according to the human's activity that is supported by the software [Dix et al., 2003] is preferred here, as it reflects the approach of this thesis.

Dix et al. [2003] suggest a cooperative work framework that is based on the entities involved in cooperative work, that is the participants themselves and the artefacts upon which they work. The possible relations that exist between the entities serve to classify the systems that support these relations. Figure 2.2 depicts these relations in the framework. The authors identify

- direct communication between the participants as a process that is facilitated by systems that support *computer-mediated communication*.
- The goal of communication is to achieve a common understanding similar to the ideas of common ground that have been introduced in 2.1.4. *Meeting and decision support systems* capture this common understanding.
- As the co-workers interact with shared work objects, the artefacts of work, *shared applications and artefacts* denote the systems that support the member's work with the artefacts.

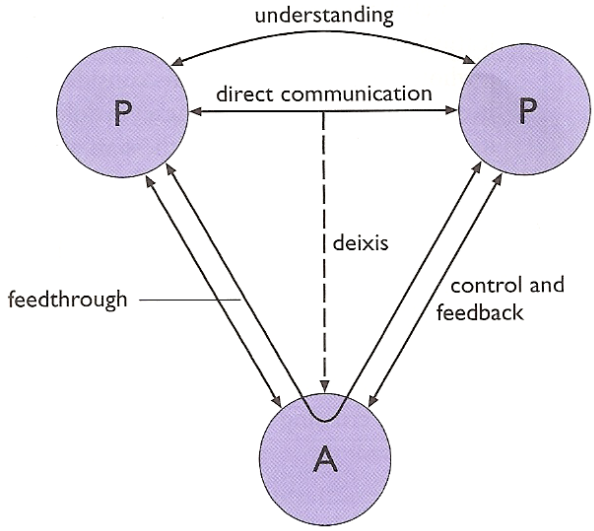

Fig. 2.2. Cooperative framework with communication through the artefact. Adopted from Dix et al. [2003, pg. 699]

For the purpose of this thesis, another relation that arises out of this framework is important to recognize. When people interact with shared artefacts, the direct communication between two participants involves deictic references to the artefacts of work. As a result, the co-workers become aware of one another's actions. Note however, that an explicit reference is not required to achieve awareness. The observations of the other co-workers actions often suffice to create awareness. Dix et al. [2003, pg. 690] refer to this process as *communication through the artefact*.

Application domains and example systems of groupware

Lotus Notes [Press, 1992] is one of the first successful commercial groupware systems. It resembles a bulletin board system based on email and database applications supporting information transfer between users. It allows concurrent revisions and modifications of shared data and for consistent updating of shared information.

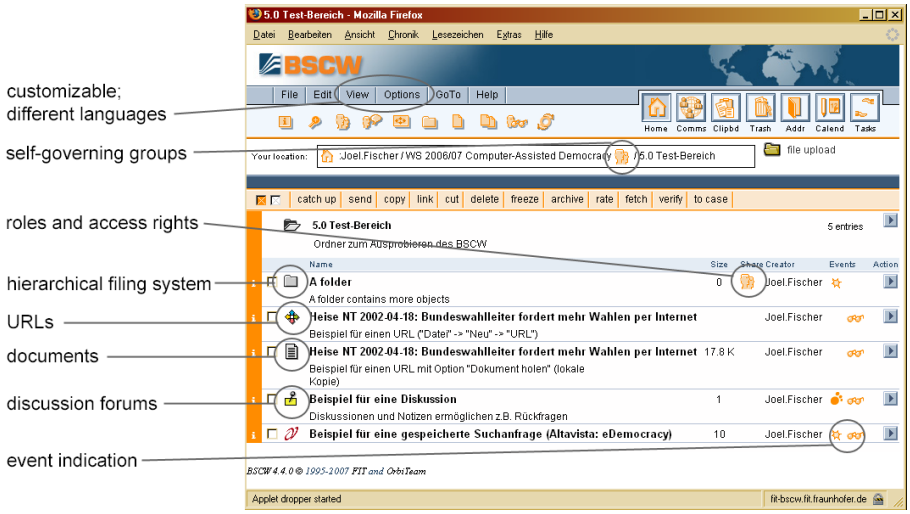

Fig. 2.3. The BSCW user inteface

.

BSCW (Basic Support for Cooperative Work) was the first groupware accessible on the web without need for client software [Bentley et al., 1995; Appelt, 1999] and has been devloped further ever since. It is a web-based shared workspace system, where authenticated members own and manage hierarchically structured document repositories stored on a central server. Figure 2.3 shows the standard user interface of BSCW as it displayed in a browser with its main features indicated. BSCW's most used feature is distributed document sharing (up- and download) [Appelt, 2001], but it also features group

discussions, workflow management, round mails, polls, and awareness mechanisms such as a daily report, information on currently online members or history of workspace events (events are user actions in the workspaces such as reading, modifying or creating documents). BSCW's sophisticated member role management, which defines the type of allowed actions for a user, enables BSCW to be used in teaching environments where the restriction of rights is required.

BSCW is mostly used in the educational sector [Appelt and Mambrey, 1999; Appelt, 2001], where it is popular because of its ideal functionality to support lectures and classroom sessions with additional material. Furthermore, BSCW supports a domain that Borghoff and Schlichter [2000] refer to very general as telecooperation and that is described as "media-supported, cooperative work executed between individuals employees, organizational units and organizations distributed across multiple locations" [ibid., pg. 105]. Therefore, BSCW is a groupware that supports cooperative work as it has been depicted earlier in this thesis. For instance, BSCW has been used to support distributed administration, e.g. as the German capital was transferred from Bonn to Berlin, the project POLIKom supported the administrative cooperation of the transfer [Fuchs et al., 1995]. BSCW is especially used as a shared workspace for distributed project work of multi-party projects, such as the EU research project IPerG[3]. Version management and locking are important features to support such text- and software-production centered projects. To facilitate orientation, users canbe annotate, rate and search for the artefacts of work by query. It also features a calendar and address book to coordinate work.

BSCW extends a standard web server by means of its Common Gateway Interface (CGI) with its kernel functionality provided in the interpreted language Python [Appelt, 1999]. With its modular client-server architecture, it can easily be amended with additional functionalitiy in the form of packages installed to the server directory or by means of methods that access the server functionality through an XML-API.

Much research in and around BSCW has been devoted to providing context awareness information for the user. The generic Event and Notification Infrastructure (ENI) [Gross and Prinz, 2004] has been tightly integrated into the BSCW architecture and enables efficient augmentation of awareness functionality. The development described in this thesis takes advantage of this infrastructure.

2.2.2 Awareness

CSCW systems can only be successfully applied if the cooperation partners are made aware of the state of the cooperation process, and its activities that are currently undertaken or have been undertaken in retrospective [Prinz, 2001].

[3] `http://www.iperg.org`

This is essential to avoid misunderstanding, coordination and synchronization problems.

Dourish and Bellotti [1992, pg. 107] stress the importance of information sharing, knowledge of group and individual activity and coordination for successful collaboration. Information relating to these factors fosters the user's awareness. According to the authors, "awareness is an understanding of the activities of others, which provides a context for your own activity" (ibid., pg. 107). The context's function is to ensure that the individual's contributions are relevant to the group's activities, goals and progress. As a result, awareness information enables coordination of the collaborative work. The authors emphasize that context is comprised of both the *content* of the shared working object and the *character* of the process of producing the object. They consider awareness of both aspects as crucial for the individual to assess the sense of the others' activities and tailor his own work accordingly.

For instance, translating this early concept of awareness to BSCW, which realizes awareness support through an event-based model, the *character* of the contribution is provided through meta information on the type and date and executer of an action, while the *content* is provided through the artefacts of work.

Dourish and Bellotti deal with a concern that shows the immaturity of awareness support at that time: the problem of how to collect awareness information. They propose the explicit generation of awareness information by the user, in separation from the actual work space object as an option. However, they did acknowledge the difficulty of user acceptance of this approach, as this means an extra work load for the user with no direct benefits. This circumstance has been identified earlier as a cause for the failure of CSCW application. Grudin [1988, pg. 86] says that

> "The application fails because it requires some people do additional work, while those people are not the ones who perceive a direct benefit from the use of the application."

However, the other option to collect awareness information Dourish and Bellotti suggest is widely employed in awareness support today: the (from the user's perspective) passive collection and subsequent presentation of awareness information in the same shared workspace that the object of collaboration is located in. It is widely acknowledged today (e.g. in [Grudin, 1994]), that an overhead of work to provide meta information on the process of work is rather unacceptable and should be avoided where possible. BSCW adheres to this principle by means of its automated event-based model – every user action is logged and saved permanently and presented as awareness information in different ways to the other users.

As stated above, groupware can be classified according to modality and locality of the nature of cooperative work it supports. In analogy, awareness support for synchronous, locally distributed work has different requirements than the support for asynchronous distributed work. An example for the for-

mer case is that of a collaborative text editor such as GROVE [Ellis et al., 1991, pg. 46]. Naturally for a synchronous collaboration tool, WYSIWIS interface issues stand out – such as shared and private view, the visual congruence of displayed information, windowing and concurrency control. Since this work conceives additional functionality for a groupware that supports asynchronous distributed work, the concept of awareness is elicited from this perspective.

With reference to Pankoke-Babatz [2003] the reason of awareness support in cooperative systems will be developed on the basis of the notions of time and place in communication and cooperation. Furthermore, different types of awareness functionality that support different scopes of cooperative work will be introduced and exemplified by pointing to the implementation of awareness support in different groupware systems.

The Role of Time and Place in Communication and Cooperation

Every environment has its own time, rhythm and its own way of documenting the past [Pankoke-Babatz, 2003, pg. 89]. This is inherent in the physical traits of the environment. The earth documents its history in its geology, ancient civilizations left proof of their existence through their modifications to the environment. Archaeology is concerned with discovering traces of long forgotten civilizations and species. Even the invisible climate left its imprint throughout history, whose traces scientists read from ice kernels like a hunter reads footprints of his prospective prey.

Human action is situated in time and place [ibid., pg. 87]. The interaction with other humans requires perception of the opposite, as it is inherent in a face-to-face scenario. As Watzlawick et al. [1969, pg. 57] suggest in one of their axioms of communication, a nature of a relationship is dependent on the punctuation of the partners' communication procedures. According to this notion, the individuals involved structure the communication flow differently; each partner rather sees his action as a reaction to the opposite's action. With respect to time, this means that an action precedes the resulting action, even though this does not imply that communication is causal. Cause and reaction is determined individually by the process of subjective punctuation, and the communicating partners might or might not agree on the communicative cause and its reaction. A typical scenario is that the speaker sees himself as mereley reacting, whereas the receiver understands just that reaction as a cause for a subsequent reaction.

Whereas perception of the opposite is inherent in face-to-face communication, decoupling the interaction from (synchronous) time and (same) place, as virtual collaborative spaces that support distributed asynchronous work intend to, necessitates designing artificial support of time in interaction. For a communication to be functioning, the receiver of information must be able to reconstruct the chronological flow of preceding bits of information, e.g. which uploaded document is a pre-requisite for the actual document of interest. As workspaces evolve during time, the timely footprints of the actions in the

workspace become essential means to comprehend the structure of information and to facilitate orientation within the workspace.

[Pankoke-Babatz, 2003, pg. 88] applies her findings on time to electronic behaviour settings and suggests the integration of the following time-related phenomena:

- Event time: this enables a common timely reference for the interaction partners to synchronize their work.
- Rhythm and tempo: As the author points out, these concepts are tightly interwoven with cultural norms and conventions. To become aware of the emerging rhythm of a workspace can be helpful to adapt one's own work flexibly to emerging conventions.
- Sequence: The shared working environment can sequencialize action.
- Documentation of the past: This enables the actors to pick up their interaction adequately after a time-out, since they can inform themselves about recent, work relevant events in the workspace.

Furthermore, the author argues that in addition to facilitating orientation in the workspace, the consideration of timely aspects enable situated action [ibid., pg. 243]. Taking into account such insights in the design of electronic settings enable mutual observation and make contextual action identifiable post hoc – and thereby support the needs of long time collaboration [ibid., pg. 96]. The perception of environmental states and events are a prerequisite for human action [Lantermann, 1980 as cited by Pankoke-Babatz, 2003, pg. 243]. The concept of awareness addresses these insights by supporting mutual observation, social presence and notification of events.

To conclude, every natural environment documents its history in its own way by storing traces of modifications that happened to it in the past. During their occurrence, events and actions in an environment are directly observable, while at a later point in time, traces in the environment and changes in the state of the environment refer to past events. To incorporate these insights into electronic shared workspaces, they have to be translated into explicitly designed traits of the workspace. To foster awareness support for an asynchronous shared workspace system such as the BSCW it is necessary to capture and store occuring activity to make it available to the user at a later point in time and thereby create awareness information. Awareness information should be supplied in an easily perceivable way to the observer. Furthermore, awareness information should be presented in a way to help reconstructing past action including event time and sequence.

Translating the lessons learned in this section into my concept of the visualization tool, orientation in a workspace can be enhanced not only by expressing the *current state* of the workspace – as this would only be a snapshot of a workspace in time that disregards the chronological evolution of activity that led to the current state in the first place. Conceptualizing this insight for an interactive visualization tool, the tool is required to make the history

of the workspace identifyable post hoc, thus visualizing the *dynamic*, *evolving nature* of a shared electronic workspace.

Types of Awareness

Prinz [2001] describes types of awareness that are independent of the time/space classification of groupware. The author distinguishes between *task-oriented* and *social awareness*. The main difference between the two is its scope. While task-oriented awareness information is centered on an object that is part (or product) of a cooperative process, social awareness focuses on the shared virtual or real environment of the users. Task-oriented awareness relates to the perception of activities within a common task on a shared object. BSCW is in essence a system that provides task-oriented awareness, other systems that support this kind of awareness according to Prinz are *GroupDesk* [Fuchs et al., 1995] and *Interlocus* [Nomura et al., 1998].

Social awareness embodies information on the social presence of others, be it in a shared virtual environment or in a real office. Classic applications that concentrate on fostering social awareness are Instant Messaging Systems such as ICQ, Skype or MSN Messenger. These tools focus on the support of synchronous communication by showing other connected peers if a user is online and allow synchronous text and voice chat. Recent approaches try to integrate location-specific awareness into such systems, as e.g. in the system *PRIMInality* [Gross and Oemig, 2005]. The system uses information of four hardware sensors to determine a more precise online state automatically; based on the possible binary combinations of the four sensors, 16 different online states are detected and displayed to the online peers, e.g. if the user is in the office at all, at the computer but with company, or in the office but away from the computer.

A somewhat more narrow and technology-centered type of awareness is *workspace awareness*. Gutwin and Greenberg [2002] define this concepts as an "up-to-the-moment understanding of another person's interaction with the shared workspace." [ibid., pg. 417] The authors clarify that the concept is bound in two ways:

> "First, workspace awareness is awareness of people and how they interact with the workspace, rather than just awareness of the workspace itself. Second, workspace awareness is limited to events happening in the workspace – inside the temporal and physical bounds of the task that the group is carrying out" [Gutwin and Greenberg, 2002, pg. 417].

In the context of this thesis it is important to realize that awareness support of users in the shared workspace is only able to base its awareness information on technologically bound activities – the system has no notion of activities outside of the system. This is especially true for the interpretation of user-artefact activities to infer mediated social relationships. To cater for this circumstance, the term *inference mechanism* is used, and inference can

only be based upon assumptions and does not neccessarily represent reality. The process of inferring mediated working relationships on the basis of user-artefact activities is described in detail in section 6.3.

However, systems that support awareness do not always just focus on supporting one type of awareness. Prinz [1999, pg. 337] recommends that CSCW-systems support both kinds of awareness, since an explicit distinction between social and task-oriented awareness cannot be made in cooperative practice. For instance, systems that are rather task-oriented may include explicit social awareness functionality. For example, BSCW has been augmented with such functionality as MetaWeb [Trevor et al., 1997] or a more recent extension, the *JMonitor* applet which gives instant information not only on recent events, but also on who's online. In addition to functionality that supports both types of awareness explicitly, the boundaries between these two types of awareness may blur in the perception of the users, e.g. the frequent activity of a user in a shared workspace is task-related but other observing users may attribute a strong social presence in the environment to that frequent user. Supporting this notion, Pankoke-Babatz states about workspace-awareness:

> "Implizit lässt sich aus diesen Informationen auch *Social Awareness* ableiten. Das heißt die erhaltenen Gewärtigkeitsinformationen können nicht nur hinsichtlich ihrer Aussagen bezüglich des Handlungsgeschehens interpretiert werden, sondern auch unter sozialen Aspekten, z.B. wer ist sehr aktiv, wer arbeitet mit denselben Gegenständen etc. Dies hat Einfluss auf die Beziehungsqualität" [Pankoke-Babatz, 2003, pg. 248].

As this work attempts to develop an inference mechanism of implicit social structure in cooperating peer groups, this implicit relation of tasks and its social embedding is a crucial notion of this thesis.

Pankoke-Babatz' distinction between types of awareness is similar to that of Prinz, she uses the term *presence-related awareness* to refer to social awareness and *activity-related awareness* to refer to task-oriented awareness. The term *activity awareness* can also be found in [Nomura et al., 1998; Neale et al., 2004] and can be understood as a broader type of task-oriented awareness, as it includes a user's actions *outside* of the workspace to fulfil a task.

Awareness Functionality: Challenges and Solutions

Awareness support in a shared environment is always entangled with the chance of transparency and its inherent problem of control. On the one hand, awareness support provides transparency to the groupware's users by giving information on the users' activities, on the other hand, therein lays danger of unwanted control of the individual's work. Several solutions have been presented to cope with this problem. Personalization of the application can include means for the user to filter out which type of actions should not be

made public to other users. Prinz [2001] suggests event-based access rights to control the distribution of event information. In addition, a subscription model requires the user to subscribe to the events he wants to be noticed of, this in turn avoids the flooding with unwanted information. This approach has been realized in the NESSIE infrastructure [Prinz, 1999].

The principle of *reciprocity* is another way of guiding awareness. User A is informed when user B gets information about user A's activities and user A can only request information on user B if B is allowed to request information on A's activities respectively. Video based media-spaces often apply this principle by transmitting the picture only when the opposite has agreed to share his picture as well.

The distribution of information carries an inherent problem for the receiver of information: she has to filter out the relevant information. In times where one often has the impression to be flooded with irrelevant information, relevance of a sender's information is crucial to the general acceptance of the sender. The relevance of a message for the receiver has been postulated as an inherent aspect of the *cooperative principle* of communication by the philosoper Grice [1989]. This is also true for cooperative systems that inform the member of activities. The worker may be a member of several workspaces with lots of activities of which only a few are relevant to the member. The notification of all events may therefore result in a "drowning" of the relevant information. A subscription model as pinpointed above is one approach to counteract flooding with undesired awareness information.

Another approach has been presented by Benford and Fahlén [1993]; Benford et al. [1995]: The *spatial awareness model* is especially appliccable for collaborative virtual environments that include user embodiment because the employed metaphors of *aura*, *focus* and *nimbus* are easily associated with a notion of embodied situated activity. The model that incorporates notions of actor's physical locations and distance between them is applied in a multi-user 3D-environment DIVE, to describe the mutual perception that two objects in this environment have. The objects can be, for example avatars or furniture, e.g. a blackboard. The avatars have an aura, a space surrounding them that describes the presence of an avatar. Avatars can only communicate if their auras overlap. The focus of an avatar is its line of vision resp. its direction of action, its nimbus describes the possible distance of perception through others. For the case of communication between two avatars, this implies that an avatar's focus must be on the other avatar and within the other avatar's nimbus. For the case of an avatar interacting with a blackboard, an avatar gets a pen to write on the blackboard if he sets his focus on the blackboard and is within the blackboard's nimbus [Benford and Fahlén, 1993 as cited by Prinz, 2001, pg. 341].

This concept can be interpreted as offering a solution for the problem of being flooded with unwanted awareness information. As it has been stated, a subscription model provides another solution to this problem. However, as I argued in the introduction to this thesis, the individual relevance of aware-

ness information is influenced by the interrelatedness of an activity with the awareness information requesting user's own activity in the workspace. This interrelation is not considered by common awareness support functionality, as it only presents information on other user's activity to the individual user. A network perspective on the activity space as it is incorporated in the visualization tool developed here however, offers a view on interrelated activities. Similar to the spatial awareness model, it expresses distance between actors, that may help the observing user to qualify other co-workers' activities as more or less related and thus more or less relevant to the user's own contributions to the workspace.

2.2.3 Information Visualization

The graphical representation of awareness information that is developed as an applet in this thesis is generally speaking a tool that visualizes information. Therefore, it is neccessary to revisit the literature about this scientific branch, to open up a general horizon what purpose the visualization of information serves.

Card et al. [1999] state in their comprehensive overview "Information Visualization" that graphical inventions of all sorts serve two distinct purposes. One is for communicating an idea. The second – and the one that information visualization is concerned with – is the use of "graphical means to create or discover the idea itself" [Card et al., 1999, pg. 1].

External cognition describes the way in which internal and external representations and processing weave together in thought. The concept tries to grasp the notion that the external world plays an important role in thought and reason. Norman [1993] suggests that the use of cognitive artefacts or physical inventions to enhance cognition is all around us.

> "A good external representation (representing perceptions, experiences, and thoughts in some other medium other than that in which they have occurred) captures the essential elements of the event (...)" [Norman, 1993][4].

Card et al. [1999] point to examples of external cognition such as navigation charts, diagrams and multiplication aids. The task of multiplying a pair of two-digit numbers can be done 5 times faster in longhand using paper and pencil than in the head. They argue that it is not really the mental multiplication that is difficult – what is difficult is to hold the partial results in memory until they can be used.

They conclude their examination by saying that the progress of civilization can be read in the invention of visual artefacts: from writing to mathematics, to maps, to printing, to diagrams to visual computing. Norman says: "the

[4] as read in Norman's summary of his book at `http://carbon.cudenver.edu/~lsherry/cognition/smart.html`, accessed Nov 9, 2007

real powers come from devising external aids that enhance cognitive abilities" [Norman, 1993, pg. 43]. This quote refers to the essence of information visualization: to exploit the interactive visual medium of graphical computers to devise external aids enhancing cognitive abilities. Card et al. reason that visual artefacts have profound effects on peoples' abilities to assimilate information, to compute with it, to understand it, to create new knowledge. "Visual artefacts and computers do for the mind what cars do for the feet or shovels do for the hands" [Card et al., 1999, pg. 5].

The authors define visualization as "the use of computer-supported, interactive visual representations of data to amplify cognition" [ibid., pg. 6], whereby cognition is understood by the authors as the acquisition or use of knowledge. This clarifies that "the purpose of visualization is insight, not pictures" [Hamming, 1973 as cited by Card et al., 1999, pg. 6]. The authors identify the main goals of insights as discovery, decision making and explanation. The ability to perform these and other cognitive activities can be increased by information visualization.

Whereas scientific visualization is understood as the visualization of physical data, information visualization faces the problem that the data does not have any obvious spatial mapping. This is also true for the visualization attempted in this thesis. The problem of mapping "nonspatial abstractions into effective visual form" [Card et al., 1999, pg. 7], has largely been solved by the research in network visualization as applied graph theory. The definition of information visualization augments the above definition in this form:

"Information Visualization is the use of computer-supported, interactive, visual representation of abstract data to amplify cognition" [ibid., pg. 7].

According to Card et al., Turkey [1977] introduced the approach of *Exploratory Data Analysis*. The approach focuses on the use of pictures to give rapid insight into data, not on the quality of graphics. It is that sort of rapid insight that is aimed at by the visualization tool developed in this thesis. Also, it will be part of the visualization to compare the external picture of the co-workers network with the internal notion of the individual's cooperative network, and the individually perceived value of the external cognitive artefact will be subject to the evaluation as well.

Cognitive Perspective on Information Visualization

A study of *InfoCanvas*, a peripheral display that shows awareness information graphically showed an advantage over a portal style and a text-based display of the same information [Plaue et al., 2004]. Participants recalled significantly more information of the graphical presentation. This study shows that graphical information leads to better recall and deeper encoding of information than purely text-based information.

The effects of textual vs. graphical information presentation on recognition and recall have long been studied in cognitive psychology. This has lead to insights such as the *dual coding theory*, which states that graphical annotation

of text or verbal speech lead to a dual coding of information in associative networks of verbal and imaginal representations [Clark and Paivio, 1991]. In general, a deeper (because dual) encoding of information leads to better memory, since schematas that represent associative knowledge are richer with information.

It can be concluded that multi-faceted, graphically enriched information visualization in general lead to deeper encoding of information and better recall of that information at a later time.

Visualizing Awareness Information

Awareness information can be visualized in different ways. Dix et al. [2003, pg.702] stress the importance of well-designed infrastructures to plug in alternative visualizations easily. Generic infrastructures such as BSCW's *Event Notification Infrastructure* (ENI) allow easily amenable visualizations of the data, as the data model is independent of the way in which it is presented – similar to the separation of concerns inherent in a *model-view-controller architecture*. With a generic infrastructure at hand, BSCW has been a research platform for different awareness services in the past.

In the TOWER system awareness information has been visualized in a 3D-world that depicts the shared artefacts as a landscape and users' actions as animations of avatars in the landscape [Prinz et al., 2003]. Figure 2.4 shows the actions of an avatar in the virtual world of TOWER's *DocuDrama* and the visualization of members reading a document at the same time [Schäfer et al., 2003].

In another, less technical view, a MVC architecture can be described as providing a *concpetual* and a *perceptual level* of description of the information it represents [Benyon, 1998]. The conceptual level provides an abstraction of the experienced world, the model; and the perceptual level provides a view or viewpoint onto that structure. The model contains the information and the view is the way in which the information is presented to the user. Benyon describes the resulting system of the two concepts generically as an information artefact. An information artefact consists of a conceptualization of objects in the experienced world and a viewport that makes that conceptualization accessible. There may be several different viewports onto the same underlying structure. Different viewports may emphasize different aspects of the same structure by providing distinct perspectives. It is important to realize that the form of the viewport denotes the function. Designers can create viewports with an intention to denote functions, but the user may interpret not just the primary function, a user will make other connotations about the culture, history and ideology of the environment in that the function is embedded [Benyon, 1998, pg. 715]. Thus, dfferent viewports lead to different interpretations of the same underlying structure.

Designers create viewports with a goal in mind of how the perspective provided by the viewport is interpreted by the user. The ENI contains infor-

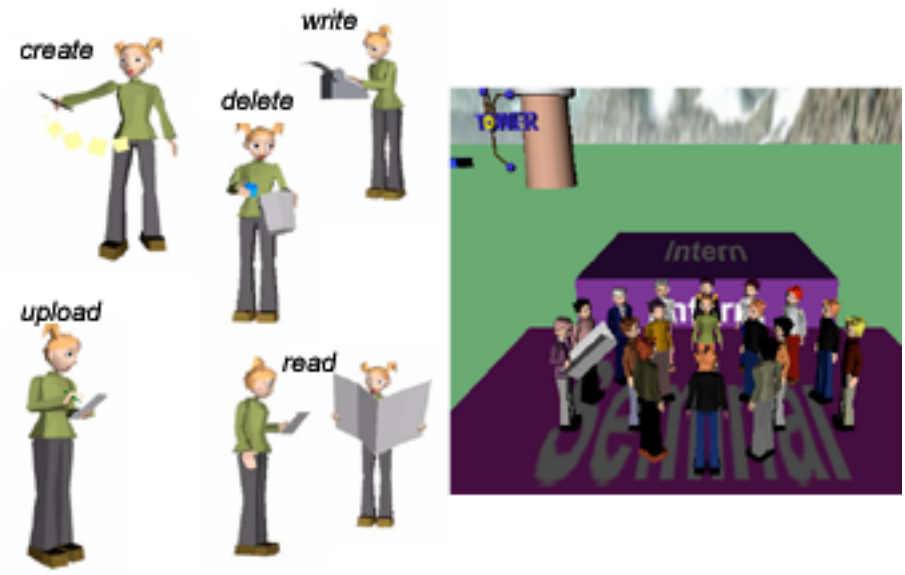

Fig. 2.4. Awareness visualization in a 3D-world in *DocuDrama*, an extension to BSCW. Adopted from [Schäfer et al., 2003]

.

mation on actor's activities on shared artefacts at a certain time and subsumes the information elements as *events*. Viewports may provide perspectices with the intention to highlight specific relations between the elements. *DocuDrama* can be seen as focussing on the cooperation of actors in the shared cooperative environment (several workspaces): In the landscape that represents the cooperative environment the actors' embodiments are grouped together on the spaces that they currently cooperate upon (see right part of fiure 2.4).

Another viewport onto the events of BSCW workspaces is *SmartMaps* [Gräther and Prinz, 2003]. SmartMaps is an applet that visualizes activities in shared workspaces in a treemap. Recent activities are emphasized in colour and tool tips show information on the events and the location of the artefact in the workspace hierarchy (see figure 2.5). Its static size enables SmartMaps to be integrated into the banner of a BSCW workspace.

SmartMaps provide a more strict hierarchical view and employs a focus on recency of events in the visualized hierarchy of workspaces. Such a view on actors and their collaborative relationships as *DocuDrama* provides, lies beyond its scope.

This comparison of different viewports shows how different foci of interpretation and meaning making can be facilitated by explicitly designing different

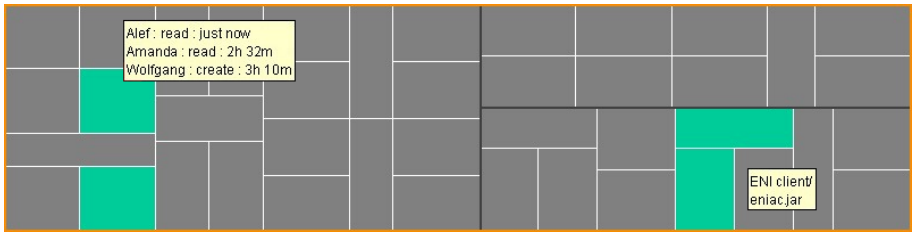

Fig. 2.5. SmartMaps visualize awareness information. Adopted from [Gräther and Prinz, 2003]

.

perspectives onto the same underlying model. However, this section is concluded by an important insight that qualifies the designer's intent:

> "The full meaning of, and activities undertaken with, information artefacts are not determined by the designer: they are produced by the users" [Benyon, 1998, pg. 715].

2.3 Social Network Analysis

The discipline of Social Network Analysis (SNA) applies techniques of mathematical graph/network theory to social aggregates. The visualization of social networks has always played a significant role for the analysis of networks [Freeman, 2000]. Freeman points out that images are critical in helping to understand network data and in helping to communicate that understanding to others. SNA often studies large, densely populated networks that emerged over a long period of time [Newman, 2004] and whose entities are connected by nonvolitional links [Fisher, 2005]. Nonvolitional in this context means that the relations between the entities are unintentional and implicit, e.g. in studies of who sat next to each other on an airplane. The analysis of large networks such as the co-authorships in the ACM library [Erten et al., 2003] describe these networks by means of statistical measures from graph theory, such as the centrality of a node, the betweenness of a node or the average number of links that connects any two nodes of the network (average distance) [de Nooy et al., 2005].

SNA has lead to important discoveries of social network properties. For instance, Milgram [Milgram, 1967] originated the *six degrees of separation* that he found out to separate the acquaintance of any two individuals in the US on average in an experiment. The application of Pareto's 80/20 rule to networks has identified the existence of hubs in networks [Barabási, 2003], whose human parallel in studies of information flow in social networks has been dubbed *bottleneck* [Cross and Parker, 2004] or *gatekeeper* [Fisher, 2005]. In an empirical study of media use and interpersonal communication we have conducted in a

seminar at university, we found that the lecturer and his assistants had the role of a communicative hub between two groups of students [Budweg et al., 2006].

The goal of SNA is to describe characteristics of networks as it is an empirical discipline; in its essence, it does not recommend interventions to improve social networks, nor can its findings easily be interpreted to do so. The transfer of insights of SNA to technological interventions to support social networks cannot be done by simply applying SNA to CSCW. A major disadvantage of traditional SNA is that it assumes a static network, that can be described by statistical measures. In contrast, the social networks that underlie cooperative work are highly dynamic in their nature, as they are context- and task-dependent [Cross and Parker, 2004], volitional and evolve over time. As members of cooperative teams alternate between indvidual and group work within several specialized teams [Gutwin and Greenberg, 2002], so do their perspectives onto their social networks change and alternate. A comprehensive analysis of the social networks of cooperative work would be cumbersome as these networks are in a state of constant change. Furthermore, what communicative channels to include? Exchange of work artefacts and information may be analyzed, as electronic systems usually allow the logging of information transfer conveniently, even though an exhaustive analysis would involve the collection and analysis of data from several systems such as e-mail, shared workspaces and instant messengers, to name a few. But what about informal social networks that arise out of face-to-face communication in coffee rooms and cafeterias? Informal networks have been shown to have tremendous influence on the performance of organizations [Cross and Parker, 2004].

The social network visualization tool developed here, does not attempt to be an application of SNA, it is not a complete tool for quantitative analysis of identified networks. Rather, it attempts to offer its user alternating perspectives onto his social networks of cooperative work. The tool visualizes structures that emerge from the usage of the groupware. Similar to contrasts between a group's formal and informal structure [Cross and Parker, 2004], the formal structure of a workspace (its folder hierarchy) differs from the structure that emerges from usage of the workspace. The tool offers a perspective onto the actors and their activities with the artefacts of work in the shared workspace and the inferred interpersonal relations that arise out of these activities. Those perspectives may be explored visually, and thereby the user can reflect on his networks and improve the awareness of relations that arise out of user-artefact interaction in the workspace. It is for this visual orientation, that statistical measures of graph theory are applied. An algorithm that places nodes and edges according to their network topology is used to facilitate visual orientation of the user (see section 6.2.1 for the design of the layout). The degree of a node (the number of edges it is connected to) is used to emphasize the connectedness of a node through size. The number of artefacts that a user is connected to can be compared to the artefacts of other users to identify people that work with similar content and thereby estimate

shared interests and expertise. If two nodes that represent users are connected via an intermediate artefact, a mediated working relationship can be inferred (see section 6.3).

The approach in this work can be summarized to enhance awareness functionality in groupware by using some statistical measures of SNA in a pragmatic way to facilitate visual orientation and reflection on activities of actors in workspaces and the arising interpersonal relationships.

3

Related Work

Two broad fields of related work can be identified for this thesis. Firstly, graph visualization of social networks from such diverse domains as literature evolution and co-authorships of scientific papers [Erten et al., 2003] or the online dating service *Friendster* [Heer and Boyd, 2005] shed light on general approaches to the visualization of social networks.

Secondly, visualization of awareness information for the domain of CSCW [Pallot et al., 2006; Seeling et al., 2007] is more specific and similar with regards to the purpose of the development exerted in this thesis, but the approaches may differ significantly w.r.t. the chosen visualization.

3.1 Social Network Visualization

Two different approaches to the application of network theory can be detected in the research literature. One approach is to employ the algorithms not only for visualization but also for analysis of the networks according to statistical measures. Another one is to use the algorithms provided by graph and network theory to visualize networks to explore the emerging views. The approach in this thesis is to follow the latter, pragmatic approach of visualizing and thereby offering means to explore the network. Thorough statistical analysis exceeds the scope of this thesis.

An explorative approach to network visualization is also followed by Heer and Boyd [2005]. They describe the design and implementation of *Vizster*, a visualization system for end-user exploration and navigation of a large-scale online social network called *Friendster*, an online dating service. The visualization integrates non-standard visual search and analysis support, e.g. connectivity highlighting, linkage views, focus and context and visualized search results. It also automatically identifies and visualizes community structures based on an algorithm that identifies group structures based on on link analysis.

Perer [2006] introduces a tool for the improved analysis of social networks. The tool focuses on the visualization of quantitative network analysis to foster an intuitive visual analysis. The author stresses the importance of user interaction with a visualization to maintain legibility of large networks. The user can select ranking criteria and the network is visualized accordingly. Thereby, lists of ranked nodes according to betweenness centrality or a matrix based on the shortest path between two nodes are visualized. The author shows how coding of network elements can be used to foster meaning-making. For instance, color and opacity is used to encode centrality or the shortest path. The distance of nodes from the center node is encoded by decreasing color saturation. Furthermore, cohesive subgroups can be visualized on the basis of link structure or attributes of the nodes to e.g. show communities of interest.

Newman [2004] presents a study of the structure of networks of scientific collaborations deducted from the pattern of coauthorship of papers for the fields of biomedicine, mathematics and physics. Newman's approach is a classical SNA approach, as he computes statistics for the three different networks such as number of authors, number of papers, papers per author, average collaoborators, average distance and clustering coefficient. The average distance between any two authors for biology, physics and mathematics is 4,6; 5,9 and 7,6 respectively. Those numbers are surprisingly close to Milgram's *six degrees of separation*.

In a similar approach, Erten et al. [2003] analyze the ACM library of 1981-2000 which contains more than 100.000 research papers and roughly the same number of distinct authors. The authors show that such questions as "What were the hottest topics in computing the 1990's?" and "How many co-authors are typical in a research paper today?" [Erten et al., 2003, pg. 1] can be answered easily by using SNA. However, the conclusions that can be drawn from those decontextualized data are minimal, and the collaborators cannot be supported in any way by the insights of purely statistical data.

The authors Fisher and Dourish [2004] present an approach to support everyday collaboration. Their starting point is that everyday collaboration is largely based on single-user interface work, that is compiled to mutually created artifacts of work. However, the tools used for this are usually not multi-user tools, e.g. multi-user text editors. The authors assume the existence of recurrent social and temporal structures, that they exploit to (also) visualize them on the basis of email traffic as an indicator for everyday collaboration. The authors use the Java Universal Network/Graph Framework[1] to build the web-client applet that dynamically reads in and visualizes networks from a database. O'Madadhain et al. [2005] stress the advantage of using a toolkit such as JUNG rather than an existing tool because it can be scripted in a much more general and flexible way.

[1] available at `http://jung.sourceforge.net`

3.2 Visualizing Awareness Information

Pallot et al. [2006] report on the integration of several technologies to accomodate current business network dynamics. They suggest an approach to integrate shared workspaces, wiki and blog into a collaborative environment where formalised concepts and members' profiles are key components to support *people-concept networking*. The authors value the navigation and exploration of people-concept networks represented in visual hypergraph structures highly and argue that this could lead to the design of new collaborative platforms to stimulate creativity and innovation.

The authors present a visualization tool implemented in a HyperGraph Java applet that shows the relationships among people and documents by rendering the *read* and *write events*, which the employed shared workspace system BSCW logs for each user, as edges between the actors of the network which are represented by nodes: the artifacts and the users (see figure 3.1. BSCW generates an XML file that dynamically calls the visualization of the selected user(s). This implementation is a development of Fraunhofer FIT employee Rudolf Ruland and provides a starting point for the implementation done within this thesis. The implementation will be referred to as the package "Readers" in the course of this work, as this is the name given to the package on the BSCW server.

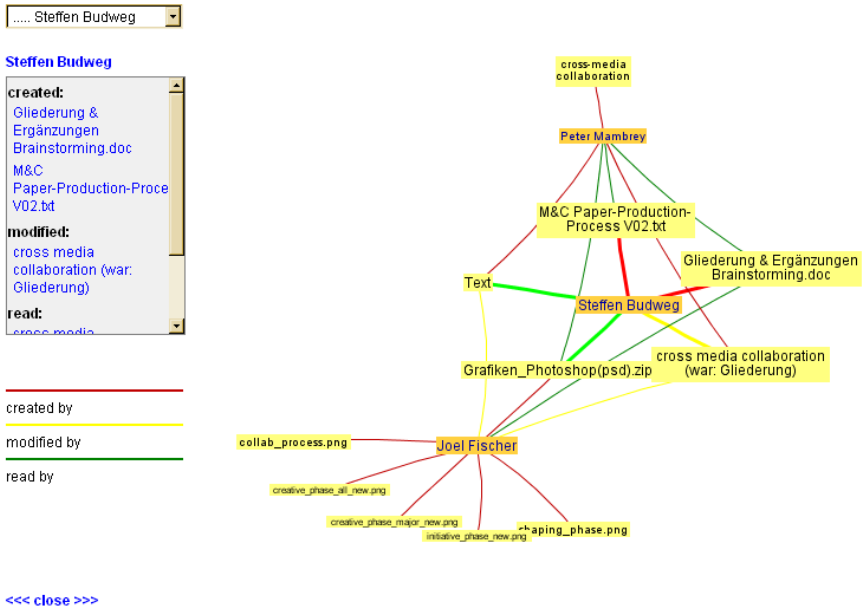

Fig. 3.1. The interface of the EventMap visualized in a HyperGraph applet.

A sophisticated approach to the visualization of workspace events to enhance awareness is proposed by Seeling et al. [2007]. The system *SWAPit* that has been integrated into BSCW combines an anlysis of unstructured text documents and related structured and hierarchical data. In a combined graphical view (see figure 3.2) a topicmap visualizes similar documents close to each other, while the other views describe activity facts such as actor, artefact, event type and event time; categories of the structured workspaces; and an interactive tool enables text and data analysis.

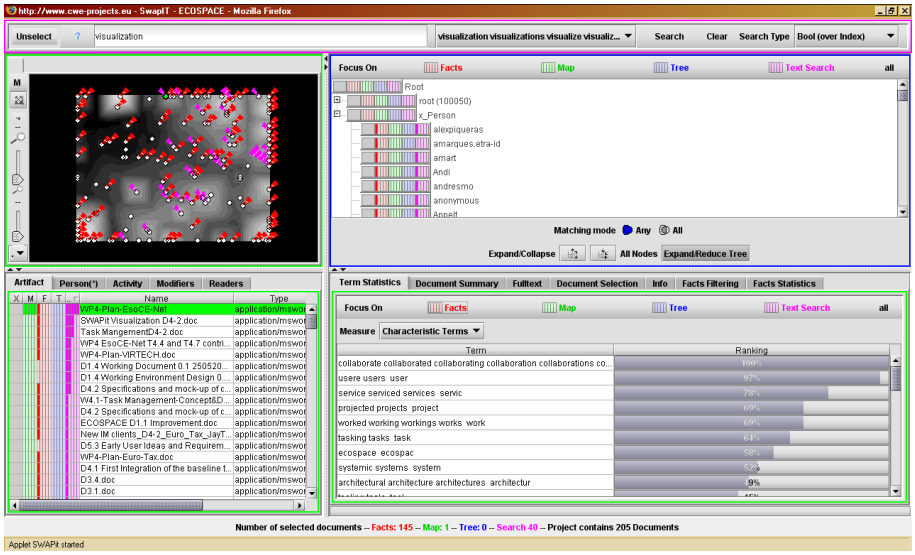

Fig. 3.2. The *SWAPit* System view of the EU research project EcoSpace workspace.

SWAPit enables the addressing of question concerning the nature of cooperation and common interest and the identification of experts in a certain topic, as it offers means to analyze workspaces according to the similiarity of documents as well as activities in the workspace. Therefore, it is possible to identify social working networks of people that share interests because they are aware of similar documents or collaborations of people that have mutually created documents. The integration of text mining techniques as suggested in Seeling et al. [2007] is especially interesting, as it can refine the identification of social working networks and allows for recommendations of cooperation. For instance, a service may be integrated that recommends to the users to contact a certain colleague because he was identified as an expert of a certain topic that the user has shown interest in. Similarly, artefacts of interest may be recommended.

Part II

Development of a BSCW Add-On to Visualize Underlying Social Networks

4

The Development Approach

It has been acknowledged in the development of information technology systems for a long time, that a system cannot be designed without considering the user [Eason, 1982], this has led to user-centered design techniques, where the user's goals and tasks are prevalent in the design process or even to participatory design where the user is present throughout the design process as an expert consultant to the designers [Floyd and Keil, 1983; Mambrey et al., 1986].

Floyd and Keil [1983] developed an evolutionary participatory system development process (STEPS), where users participate in each cycle of the iterative process from design and specification to usage, evaluation and redesign. The system is evaluated according to the users' everyday context of work in the real world and not under artificial laboratory setting. The user is considered an expert of his daily work and therefore a valuable consultant to the designers and implementers of the system. Throughout a system's lifecycle, interactive methods such as expert discussions or workshops are conducted to unite users and designers as partners in the design (and redesign) process. STEPS has for instance been adopted by PoliTeam [Fuchs et al., 1995].

The development process in this work employs a user-centered development approach, meaning that the user's benefit is the prevalent goal of the design process. However, for reasons of feasibility, during the design and specification users do not participate as consultants of design. Nevertheless, an empirical requirements analysis collects the user's opinions on the tool's desired functionality. The analysis serves the purpose of sufficiently motivating the development. It will reveal the characteristics of the design of the visualization tool that actual users find appealing. Furthermore, explicit design decisions that underlie the functionality can be underpinned empirically with the results of the requirements analysis. In addition, the analysis will try to found the hypothesis empirically (see chapter 5 for a more thorough discussion).

Figure 4.1 shows the development process. As suggested by Floyd and Keil [1983] and taken on by [Fuchs et al., 1995; Pankoke-Babatz, 2003] the devel-

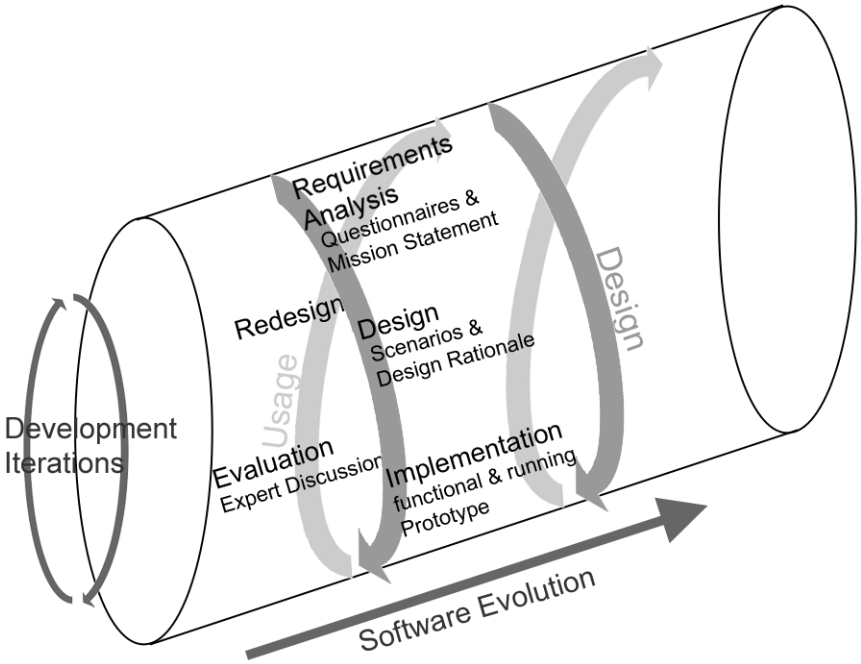

Fig. 4.1. The Development Process. Phases of Design and Methods used.

opment process does not end with the implementation of the first version of software. As the authors suggest, the evaluation of software should continue when it is in use, for only *technology-in-use* enlightens strengths and shortcomings as well as the whole spectrum of effects caused by the software. Therefore, figure 4.1 situates the evaluation in the usage phase of the evolutionary process of software. Thereby, the evaluation becomes formative for the next (re-) design iterations. Within this work, one cycle in the iterative development process is run through. The first phase, a user-centered requirements analysis will add empirical support to the foundational justification of the development (see chapter 5). In the second phase, the design of the visualization tool will be conceptualized (see chapter 6). Scenarios [Carroll, 2000] will exemplify real world contexts of usage of the tool and design rationale techniques are employed to make design decisions transparent. Principles and heuristics of user interface design [Shneiderman, 1998; Norman, 1998] furthermore guide the decisions in the design process.

In phase three, the concept will be transformed into a running prototype. The specification of the prototype is depicted in chapter 7. The programmed visualization tool is an add-on to an existing system (BSCW). The current version 4.3.4 of BSCW offers two distinct possibilities to develop add-on func-

tionality. Client applications can access the BSCW server's functionality by means of an existing RPC-API. Yet, the more robust and user-friendly variant is to implement the functionality as an additional package to the directory of the server, since users do not have to install additional software to their system. Therefore, the visualization add-on will be implemented on the server side. The prototype is developed on a locally installed BSCW server.

An initial evaluation of the visualization is the scheduled final step of the development process. The evaluation (see chapter 8) will provide feedback on the design and will be conducted according to insights of ethnographic research: the tool will be tested in the real world context of everyday work of the user, where the user is seen as an expert of his work and partner in the evaluation. In an expert interview, the tool will be evaluated according to the ways in which the tool supports the user's everyday goals and tasks of using the system. Expert interviews provide a feasible way to explore the benefits and shortcomings of the resulting views. The tool is tested on a BSCW server that the eProfessionals that contributed to the requirements analysis have access to.

4.1 Used Technology

The visualization tool is implemented as add-on functionality to the BSCW system. Instead of building the visualization tool from scratch, an existing graph visualization framework will be used. Two graph visualization frameworks will be visited and their potential for usage in this development will be estimated. Due to the potentially complex behavior applications provide, modern programming languages offer the possibility to encapsulate complexity in programming libraries. This reduces the amount of code to be written by the programmer and enables the implementation of complex behavior within a short amount of time.

For web-based technologies, Java provides state-of-the-art functionality and support of interactivity. A Java Abstract Programming Interface (Java API) provides complex behavior in a flexible way that can be adapted to the specific needs of an implementation for BSCW. Java offers portability of code, therefore the implemented functionality can easily be moved and employed to different BSCW servers. In addition, Java guarantees platform-independentness, thus, the user's operating system is transparent to the tool, meaning that it is not affected by the type of operating system run on the user's computer.

The visualization tool is implemented as a Java Applet. Java Applets run in regular web browsers and provide interactive features to web applications that cannot be provided by HTML. BSCW functionality has been augmented with Java Applets in the past. For example, *JMonitor* provides awareness information by showing who's online and which events happened recently in the workspaces of the user.

Since the BSCW is implemented in Python and the visualization tool in Java, the data that the tool uses to visualize the peer networks has to be provided through an interface. A straigtforward way to pass data is by means of XML files. On the the user's call, responsible BSCW methods generate an XML file at runtime. This XML file is a description of the basic properties of the graph to be visualized by the Java Applet, including the user nodes, the document nodes and the events that exist as relations between users and documents and additional properties of the relations such as date and type. The XML file is parsed by a Java XML file handler and transformed into the desired objects of the visualization. The process is described in detail in chapter 7.

4.2 Graph Visualization Frameworks

In the following, existing graph visualization frameworks will be visited and assessed according to the requirements of the intended development.

4.2.1 HyperGraph

HyperGraph[1] is an open source project that has the potential to provide visualization functionality to be implemented in this work. A Java Applet that makes use of the HyperGraph API is already being tested on BSCW within the visualization tool provided by the functionality "Readers" [Pallot et al., 2006] (see also fig. 3.1). HyperGraph provides a Java library that contains the necessary classes and some methods to visualize graphs and especially hyperbolic geometry. Via JavaScript, Java methods can be called from the UI in the browser to dynamically interact with the visualization. Figure 4.2 gives an example of a network visualization of the Java Applet that implements HyperGraph functionality. The visualization shows the readers (but also modifiers and creators) of the documents in the current workspace.

HyperGraph is successfully being tested on a BSCW server already - therefore, to chose HyperGraph would mean to be able to fall back on existing experiences with the implementation of this framework. However, HyperGraph today does not provide algorithms that lay out the graph in a meaningful way, e.g. the distance of the nodes cannot be encoded meaningful, so as to depict similarity of interest or awareness of the same artefacts between users. Thus, algorithms that add meaning to the topology of the graph would have to be implemented from scratch. Also, HyperGraph does not include interactivity for the user such as filtering or highlighting of nodes and edges - neccessary features for human sense-making [Perer, 2006] of complex graphical pictures.

HyperGraph has mostly been designed to lay out hyperbolic geometry, prominent examples for the application of this are hyperbolic browsers, e.g.

[1] available at `http://hypergraph.sourceforge.net`

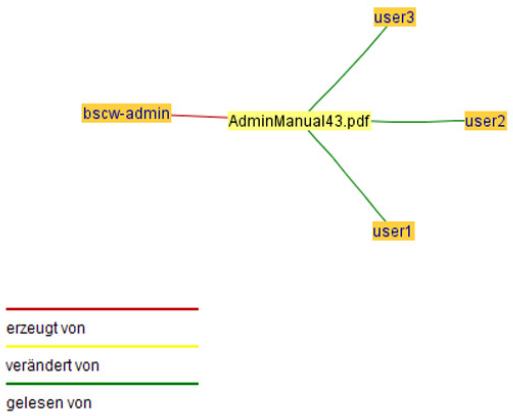

Fig. 4.2. An example of a visualization created by the Java Applet "Readers" on the test server. The picture shows the creator of the document and its readers.

HyperGraph is used on `http://www.mfirst.de/wi/` to visualize the german *Wirtschaftsinformatikindex*. For the purpose of visualizing networks the framework introduced in the next section is better suited.

4.2.2 Java Universal Network/Graph Framework

An alternative to HyperGraph is the Java Universal Network/Graph Framework[2] (JUNG).

> "JUNG is an open-source software library that provides a common and extendible language for the modeling, analysis, and visualization of data that can be represented as a graph or network. It is written in Java, which allows JUNG-based applications to make use of the extensive built-in capabilities of the Java API, as well as those of other existing third-party Java libraries." [O'Madadhain et al., 2005, pg. 1].

Figure 4.3 shows an example of JUNG's capabilities to visualize a network and, in this case, a demonstration of the shortest-path algorithm. The user can select two nodes and the shortest path between them will be highlighted.

JUNG has been developed at the School of Information and Computer Sciences at the University of California, Irvine. It is distributed under the BSD open-source license which allows anyone to create derived work from JUNG. The developers of JUNG set out to provide network data analysis with an "extendible language for the manipulation, analysis and visualization

[2] available at `http://jung.sf.net`

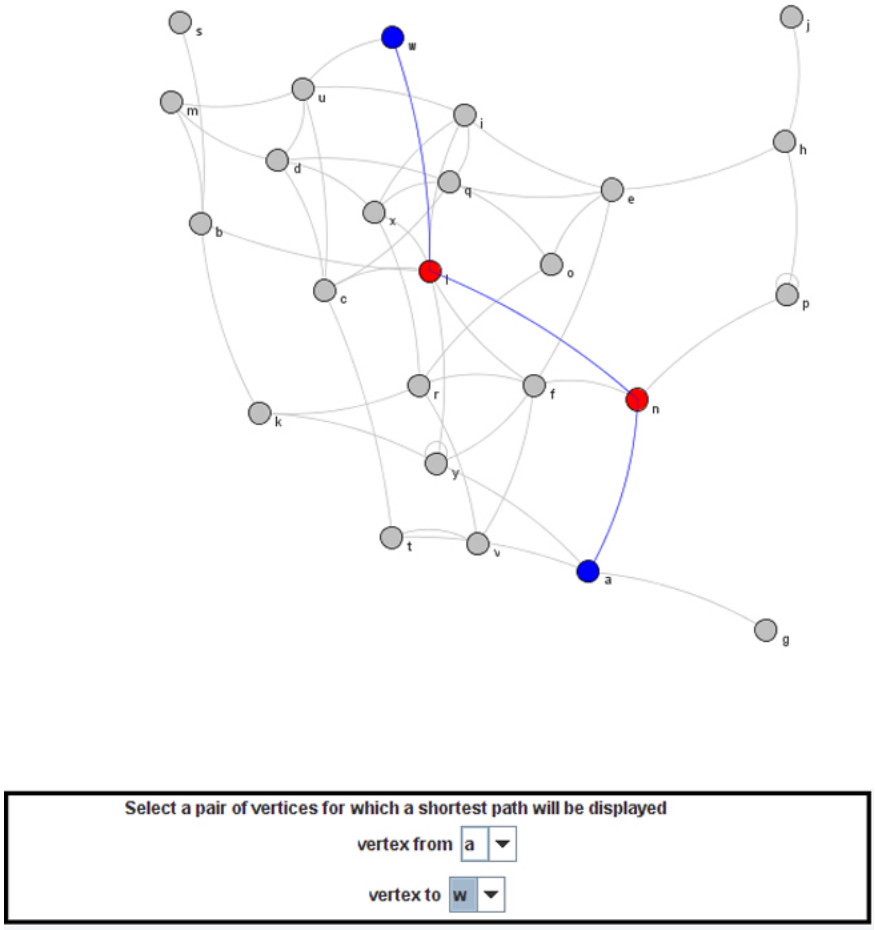

Fig. 4.3. A network visualized with JUNG. The shortest path between the nodes w and a are is highlighted. Screenshot produced from examples available at `http://jung.sourceforge.net/applet`.

of data that can be represented as a graph or network" [O'Madadhain et al., 2005, pg. 2]. Since the add-on developed within this thesis will be based on JUNG, the framework's most important features will be described here.

The major features of JUNG according to O'Madadhain et al. [2005] include:

- Support for a variety of representations of entities and their relations, such as directed, undirected or multigraphs (graphs with parallel edges) and bipartite graphs. The latter of which will be particularly relevant for the design of our visualization tool.

- Facilities for annotating graphs, entities and relations with metadata. Those mechanisms will be used to label the nodes and edges of our graph with names and also with data that will be computed for the dynamic visualization of the graph.
- Implementations of a number of algorithms from graph theory, exploratory data analysis, social network analysis, such as routines for clustering, statistical analysis, network distances and ranking measures.
- A visualization framework that makes it easy to construct tools for the interactive exploration of network data.
- Filtering mechanisms that extract subsets of a network; this allows the user to adjust the visualization according to the parameters that he's interested in.

To make JUNG a truly extendible and flexible language, O'Madadhain et al. [2005] outline some design principles of their framework. JUNG uses combinations of layers of abstractions that Java provides by employing Interfaces, abstract classes and implementation classes. This separates specifications (e.g. interfaces) from their implementations (e.g. implementing classes). Also, the object-oriented idea of inheritance can be found throughout the framework. JUNG makes heavy use of expressions called predicates, that, when evaluated on a specific argument, return'"true" or "false" to constrain e.g. the vertices or edges that may be added to a graph. Predicates are also used as flexible specifications for filters, for example, an edge can only be drawn when it passes a certain argument. To cater for this, JUNG makes extensive use of the Commons Collection library [ApacheJakartaProject, 2007], especially of the `Predicate` interface. The library Colt [CERN, 2007] is also employed regularly throughout JUNG - it provides functionality for high-performance scientific computing that JUNG applies to its graphs.

Basic Concepts

On an architectural level, JUNG tries to incorporate a model-view-controller model. The graph represents the model, the view is represented by the layout, the renderer and the visualization viewer – the controller is implementation dependent. The user can tweak the appearance of the visualization by means of the UI, then the implemented event and change listeners update the view, whereas the model remains unchanged.

The graphs, vertices and edges in JUNG have properties that can be extracted and operations that they can perform. Those properties and operations are defined by the interfaces, where each type of implementation of graph, vertex or edge may have different operations allowed on its objects.

Figure 4.4 provides the type hierarchy for graphs. The interface `ArchetypeGraph` defines a graph to be a container of vertices and edges, with methods for accessing and modifying them, defining constraints for vertices and edges and for specifying listeners, so as to handle changes the user

makes at the UI-representation of the graph. The interface `Graph` extends `ArchetypeGraph` and is specialized for graphs whose edges connect exactly two vertices. `Graph` in turn has subinterfaces, of which `KPartiteGraph` and its implementation `KPartiteSparseGraph` is especially interesting for this work.

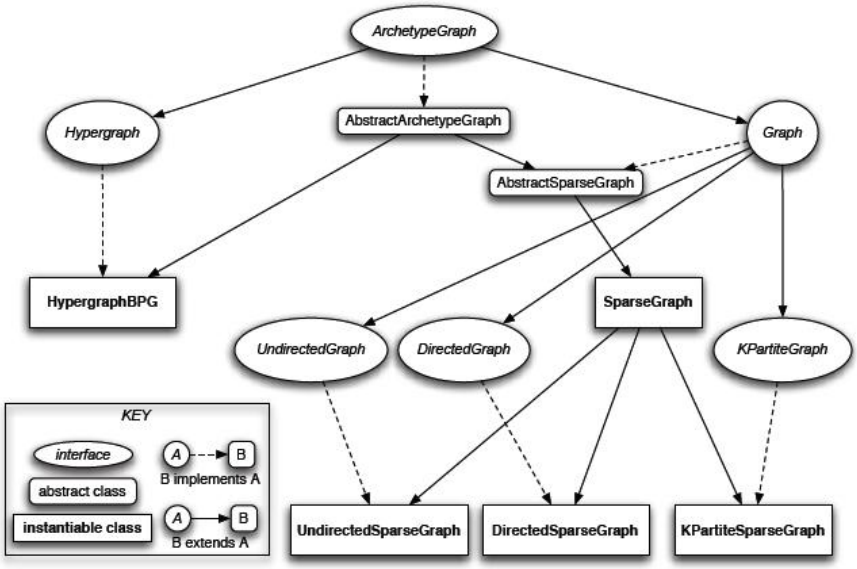

Fig. 4.4. The type hierarchy for graphs (adopted from [O'Madadhain et al., 2005, pg. 9]).

The other two most important elements of the framework are vertices and edges, both of which are defined in a similar way as the graph element in several interfaces, abstract and implementing classes. Here, I will not go into any more detail about the type hierarchies.

Creating and Adding

JUNG provides three different ways of creating a graph. You can create a graph explicitly by calling the constructor of the desired graph, e.g.

```
KPartiteGraph g = new KPartiteSparseGraph();
```

The alternative that is chosen in this work, you can read the graph from a file. Currently, JUNG supports only simple Pajek [Batagelj and Mrvar, 2007] and GraphML files [Brandes et al., 2007]. Pajek is a proprietary format designed for the network analysis tool Pajek. GraphML is a more general format.

However, the functionality of parsing GraphML files is very limited and requires additional programming to cater for the specific needs of visualization in this work. O'Madadhain et al. [2005, pg. 24] argue that they cannot provide data input interfaces to cater for the needs of the applications build with their framework due to the many possible formats data can be stored in. Developers will have to write their own parsers.

GraphML is similar to the GraphXML file format [Herman and Marshall, 2000], that is used in the "Readers" package as an interface between the native BSCW operations written in the programming language Python [PythonSoftwareFoundation, 2007] and the Java applet code to visualize the graph. Both formats are XML-based representations of graphs. Since the tool developed here will also read in data from the BSCW, the graph has to be read in from a file (see chapter 7).

Once an empty graph has been generated, you have to create and add vertices to the graph by calling, e.g.

```
Vertex v1 = g.addVertex(new UndirectedSparseVertex());
Vertex v2 = g.addVertex(new UndirectedSparseVertex());
```

Finally, and edge is added in between the two vertices by calling:

```
Edge e = new UndirectedSparseEdge(v1, v2);
g.addEdge(e);
```

Extending Graphs with User Data

JUNG provides a built-in mechanism, the `UserData` class that has a number of useful operations to annotate JUNG objects with meta information, e.g. vertices and edges. Each object has an associated user data repository that is useful for storing additional information to vertices and edges. For instance, all the metadata that is necessary for our add-on is stored here, such as user names, event types and dates. For example, a user's name, which is represented by a vertex, can be stored like this:

```
String user_key = "user name";
v.addUserDatum(user_key, "My Name", UserData.SHARED);
```

where `UserData.SHARED` is the type of copy-action employed by the copy method. That kind of data can be used to check the vertex against a constraint where it will be filtered out if it doesn't pass it, or simply to display the data at the UI.

Decorating and Labelling Graphs

JUNG comes with a subpackage that allows customizing the way the graph elements look – it allows setting the size, colour, shape and stroke of the

vertices and edges and to attach labels to both of them. Of course, these can be altered to accommodate certain operational logic, so that certain types of vertices or edges are displayed in a certain way.

Algorithms

The framework comes with a number of algorithms that can be applied to a graph. Those algorithms assign values to certain elements of the graph according to the graph's structural properties or lay out the graph in a certain way to reflect the topology of the graph. It includes ranking algorithms that compute the "influence", "authority" or "centrality" of a given vertex, algorithms that derive clusters from a graph and algorithms that measure paths between certain vertices. JUNG also includes random graph generation from certain probabilistic models and classes that calculate statistical measures on graphs, e.g. degree distributions. Furthermore, functionality is provided that transform graphs, one of which is of utter importance for the development in this work. JUNG offers functionality to transform a bipartite graph – a graph in which the vertices belong to two distinct partitions – into a unipartite graph. For instance, the bipartite graph would be made up of vertices that represent BSCW users and BSCW artefacts. The edges would represent *read*, *write* or *modify*-relations. The class `KPartiteFolder` can transform this graph into a unipartite graph where the vertices are the users and the edges represent the mediated relations between the users. That is, if the edges (a, b) and (b, c) exist in the original graph, then and edge (a, c) will be generated in the transformed graph [O'Madadhain et al., 2005, pg. 23].

Visualization

The functionality JUNG provides to lay out and render graphs requires the Java Swing API. There are three main components responsible for visualizing the graph. The `VisualizationViewer` extends a Java Swing component, a `JPanel`, and represents the drawing area upon which the graph is rendered. The `Layout` takes the graph and determines the location at which each vertex is drawn, depending on the implementation of `Layout`. Finally, the implementation of `Renderer` takes the data provided by the `Layout` and paints the vertices and edges into the provided `VisualizationViewer`.

5

Empirical Requirements Analysis

5.1 Research Design

According to principles of user-centered design, the requirements will be empirically polled. Whereas the overall necessity for the development of a visualization tool of the co-workers network is justified and motivated by the exploration of the background research (see chapter 2) and related work (see chapter 3), a requirements analysis will sufficiently motivate the development.

The analysis among two distinct user groups will help to arrive at an understanding how users perceive the current view on BSCW to test if the author is correct about the starting assumptions. The analysis also aims at finding out about the character of cooperative work in BSCW and to reveal hypothetic chances and challenges of the visualization tool. It will reveal the characteristics of the design of the visualization tool that actual users find appealing.

5.1.1 Research Questions

In the introduction, it was hypothesised that the current view on the BSCW (that is, its UI) places the artefacts of work in the focus, rather than the people creating, modifying and reading. It is argued that the metaphor of a hierarchical filing system does not contribute to an awareness of the individual member's position in the co-workers network. A personal network is hard to identify because the member would have to remember the creators of the documents he read and modified, as well as he would have to know the readers and modifiers of his artefacts of work. The package "readers" supports that and is an initial step, but it still places the artefacts into the center of focus (also see chapter 6). The requirements analysis tests this hypothesis against the opinion of other BSCW users.

Furthermore, the requirements analysis will reveal important characteristics of a network visualisation in the user's eye. For instance, different views

on network exist. The users might as well be asked about the desired view. Do they prefer an egocentric view of the network or a holistic view, which depicts the network according to network topology rules? For example, with the most connected node in the central place. As this view can be used to control the member's individual position, this view is assumed to be protested against. Fisher and Dourish [2004] warn that "gathering and analyzing traces of individual activity pose[s] significant potential for invasion of privacy". The type of functionality that a user desires will be looked into.

Some demographic questions will also be asked, e.g. gender, age and level of BSCW expertise.

5.1.2 Methods

A randomly chosen student group and a randomly chosen group of eProfessionals are questioned by means of a short online questionnaire. The tool used is a BSCW poll. To cater for the possibility of comparing the two groups, two identical versions of the questionnaire have been developed: to one the students will be invited and to the other one the eProfessionals. The survey of two distinct user groups enables an analysis of the differences and the similarities between the two groups.

Questionnaire

At the beginning of the questionnaire, the subjects are introduced to the underlying assumption that by means of the BSCW people cooperate that are embedded in a social network. The subject is informed that the questions relate to the subject's social network in which he works and uses BSCW to support his work. The term "working network" is introduced to cater for this context.

The questionnaire seeks to test several assumptions of this work empirically. If not stated differently, the questions are statements the subject is asked to agree or disagree to on a five-point Likert scale. First, questions about the current view onto BSCW are posed to test if the author is correct about the starting assumptions that motivated the development of an add-on in the first place. The subject was asked to correspond to the statement that the current view onto BSCW focuses on the artefacts of work and not onto the persons that work with the artefacts. Additionally, they were asked if the current view allows identifying the type and the degree of working relations to other persons in their working network, e.g. to whom they have a close co-working relation. Then the subject's opinion was asked on the statement that the "hierarchical folder structure" of BSCW makes it cumbersome to overview whose documents the subject reads and modifies and who reads and modifies his documents. Finally, the subject was asked to correspond to the statement "The current view onto BSCW lets me feel as part of a working network".

The second scope of the questionnaire was to test hypothesises about the character of cooperative work in BSCW. In particular concerning the distinction between the two most common actions a user generates on an existing document: read and modify events. To see if an assumption of the kind "if two people read the same document, a social relation exists between them" is correct the subject was asked to correspond to the statement. "If I read documents of another person in BSCW, then a direct relation to this person exists in my working network." To compare differences between read and modify events, the same statement was made with "read" exchanged by "modify". To see if the closeness of a relation increases with the number of documents commonly read or modified, the subject is asked if "The more documents I read of one person, the closer this person is to me in my working network.", again, the same statement was formulated for modify events. To give the subject opportunity to compare common read and modify actions and their impact on their working networks explicitly, the subject is finally asked to give his opinion on the statement "I work closer with people whose documents I modify than with people whose documents I read."

Thirdly, the questionnaire's goal is to collect an initial opinion about the chances of an add-on that visualizes the working network and on some design decisions that subjects can contribute to in a user-centered manner. In absence of any graphic means, two different network visualizations are described textually to the user and then he is asked to correspond to statements relating to the description of the visualization. At first, a bipartite graph is described that shows users and documents as nodes and events such as read-, create-, and modify-events as edges between the nodes. The subject is then asked to correspond to the statement that such a view onto BSCW would improve his overview of the working network. Then he is asked if such a view has the potential to improve the perception of the type and the degree of working relations. The second description of a graph is one where all the nodes are persons and the edges are the relations between the nodes. The subject is asked if such a view would place the node that represents him into the center and group the other persons around him depending on how close they work together (egocentric view) or if such a view would show the entire selected workspace (holistic view) and not center the current user but rather a different node, e.g. the one with the highest centrality, the node with the most edges. As this view can be used to control the individual's position in the working network, the subject is asked if because of the possibility of control they decline this view. About both of the views (the egocentric and the holistic view) the subject is asked to agree or disagree on the statement that such a view would have the potential to improve the perception of the type and degree of working relations.

At the end of the questionnaire, the subject is asked to provide demographic information. First, he is asked to rate his user level, if he sees himself as a beginner, advanced or expert user. The subject's gender and age is levied.

Before the subject can provide a final comment, he is asked to rate the comprehensibility of the questions of the survey.

5.1.3 Population/Sample

The potential population of this requirement analysis are all the BSCW users. Since BSCW is heavily used in (mostly universital) learning and teaching (Kerres et al. [2004], Deussen et al. [2004]) students are a well-represented user group among all BSCW users. Therefore, one group of subjects is recruited from students of the University of Duisburg-Essen – the only requirement is that they are familiar with BSCW and have used it before. The sample is incidental. The student group is made up of a seminar that Peter Mambrey conducted in summer 2007 at the University of Duisburg-Essen. Students of different faculty memberships attend the interdisciplinary seminar. Some are from the faculty of engineering, others are from the social sciences. 27 students participated in the survey.

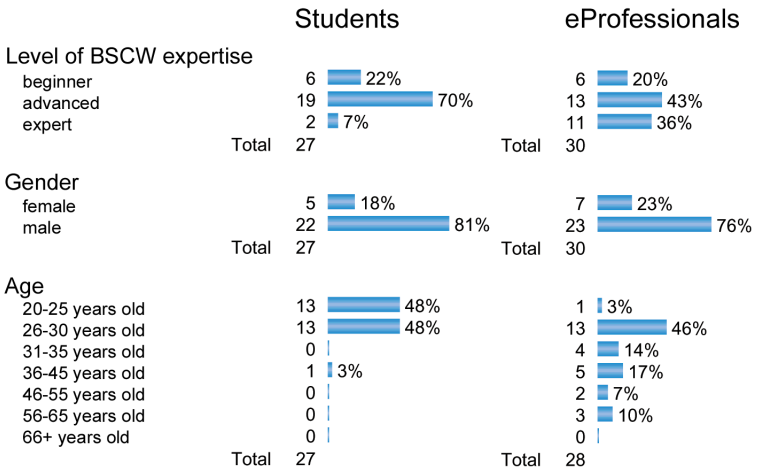

Fig. 5.1. The demography of the sample of the requirements analysis.

Another well-represented user group are members of organizations that employ groupware to support their asynchronous, locally distributed work, as described in [Mambrey et al., 2003]. These organizations can be from any sector, e.g. industry, governmental or from the scientific research community. For reasons of practicality, the sample of eProfessionals is recruited from the employees of the Fraunhofer Institute for Applied Information Technology (FIT). This sample is promising as probably most of the employees of Fraunhofer FIT work with BSCW on a daily basis to support their work, since

BSCW has been developed by the predecessor of FIT, the GMD [Appelt, 1999]. 30 eProfessionals participated in the survey.

The demography of the two groups is summarized in figure 5.1.

5.2 Results

Figure 5.2 shows the interpolated means of the likert-scale items (questions 1-4) with a confidence interval of 95% for each of the groups. Thus, a probability of error of $\alpha = 5\%$ is assumed. As it can be seen at once, the intervals of the standard error of the mean clearly overlap for all items; therefore, the differences between the two groups are not significant ($p > .05$).

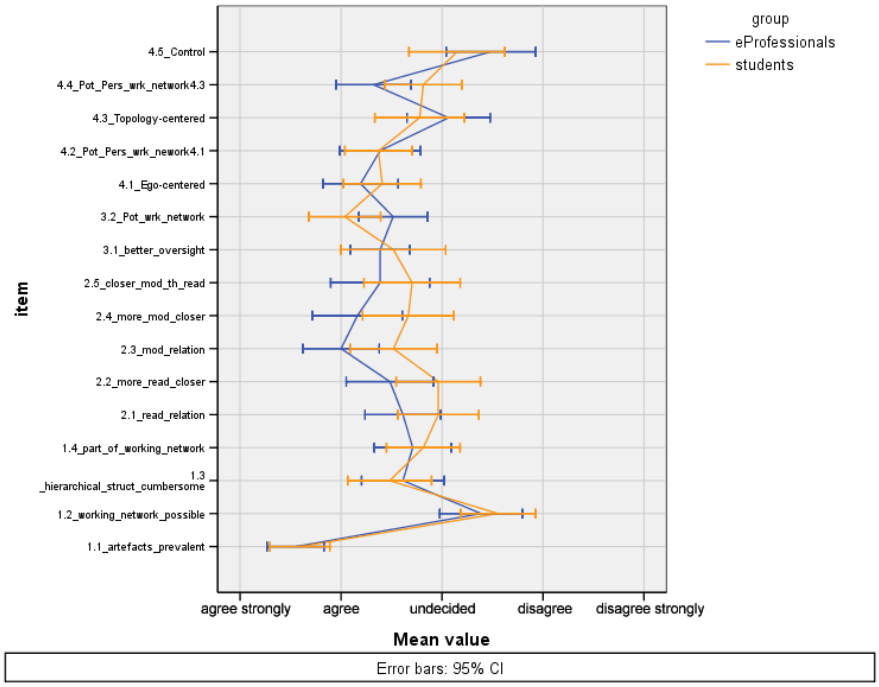

Fig. 5.2. Interpolated means with a confidence interval of 95% for each group.

The overall trend of the two groups is quite similar, whereas the eProfessionals tend to agree slightly more than the students, except for the items 3.2, 4.3 and 4.5. The items are numbered from bottom to top in ascending order with respect to the scopes of the questionnaire delineated in the section above. The items 1.1-1.4 concern the current view on the BSCW, the items 2.1.-2.5

inquire about the character of cooperative work and the questions 3.1-4.5 try to grasp a potential impact of a visualization tool to enhance awareness.

The first statement of the questionnaire was the most agreed upon. The statement "the current view on BSCW rather focuses on the artefacts of work than on the persons that work with the artefacts" experienced the most acceptance with 89% of the eProfessionals agreeing (33%) and agreeing strongly (56%), and 91% of the students agreeing (40%) and agreeing strongly (51%). Whereas the majority of the subjects disagreed with the statement "the current view onto BSCW allows the identification of the type and the degree of working relations to other persons in their working network" (see item 1.2 on fig. 5.2). W.r.t. item 1.3 subjects were more careful; their answers intermit between undecidenness and agreement. However, still a majority of 53% of the subjects agree or agree strongly with the statement "the metaphor 'hierarchichal folder structure' makes it cumbersome to overview whose documents I read and modify and who reads and modifies my documents". The agreement with the statement "the current view onto BSCW makes me feel as part of a working network" is even less, even though it is still positive.

The second set of items of the questionnaire refers to the character of cooperative work that can be inferred from relations of events users commit commonly on an artefact. Whereas the interpolated means run almost congruent to each other for the two groups for the first set of question, the eProfessionals tend to answer roughly half a category more positive in this section than the students. The statement "if I read documents from another person in BSCW, there exists a direct connection to this person in my working network" was agreed upon by 46% of the eProfessionals and only 32% of the students. The agreement is far greater for modify relations (see item 2.3 in fig. 5.2), where 58% of the students agree and 77% of the eProfessionals (40% even agree strongly). To see if the closeness of a relation increases with the number of documents commonly read or modified, the subject is asked if "The more documents I read of one person, the closer this person is to me in my working network.", again, the same statement was formulated for modify events. Interestingly, the agreement for increasing common read events leading to a closer relationship rises as compared to the answers in 2.1, whereas the agreement for the statement for modify events drops as compared to the non-causal statement in 3.3 (see fig. 5.2). The statement "I work closer with persons whose documents I modify than with persons whose documents I read" was agreed upon by 59% of the eProfessionals and 55% of the students.

eProfessionals and students approach each other again when they assess the potential of a visualization of a working network. First, subjects were asked to correnspond to the statement "A visualization of a user-artefact network would enhance the overview of my working network". 69% of the eProfessionals agree (6% of them strongly) while 51% of the students agree (29% of them agree strongly). When it comes to estimate the potential of an enhanced perception of the type and degree of their working relations (item 3.2), 50% of the eProfessionals agree and additionally, 10% of them agree strongly, and

even 76% of the students agree (25% of them strongly). The same statement was posed for a ego-centered and a topology-oriented user network (without artefacts). Comparing these values on a potential user-artefact network with the estimated potential impact that a network comprised only of users has (items 3.2, 4.2 and 4.4 respectively), tne differences are virtually nonexistent for eProfessionals, whereas students value the enriching potential of a user-artefact network higher than an ego-centered user network and this, higher than a topology-oriented user network.

The subjects are questionned on the desired network topology of the personal network inherent in the statements "the view would center the node representing myself" (item 4.1) and "the view would not place the node representing myself into the center but the node representing the user with the most relations into the center". Here the eProfessionals clearly favor an ego-centered view (20% agree strongly, another 60 % agree) over a topology-oriented view (30% disagree and 36% remain undecided). The students arrive at similar answers and it is especially interesting, that with 40% undecided, subjects do not seem to have a clear opinion on this statement. The opinion on the potential danger on control was inherent in item 4.5, where subjects were asked to correspond to the statement "I decline a topology-centered view because a third party could control my position, type and degree of working relations". The majority (56%) of eProfessionals disagrees with this statement, whereas the students do not agree among each other. 39% disagree, 32% agree and another 25% are undecided.

The detailed responses of the two groups to the questionnaires can be found in appendix A.

5.3 Interpretation of the Results

With respect to the overall positive and agreeing answering behavior of the subjects, the disagreement that the current view onto BSCW allows the identification of the type and degree of working relations to other persons in their working network is a sign for the absence of means that support the individual's awareness of his working network.

The rather indifferent answers concerning common read events (items 3.1 and 3.2) suggest that common read activities on a document should not be overestimated when inferring cooperation between two co-workers. In contrast, the overwhelming agreement concerning common modify events (3.3) and the resulting cooperative relation between two individuals emphasize the existence of a cooperative relationship between individuals that modify each other's artefacts. This is supported by the agreement of the majority (58%) that they work closer with people whose documents they modify than whose documents they read. However, an increase of the intensity of the relationship by number of commonly modified documents cannot be interpreted without doubt, as the mean of this item (3.4) drops compared to the previous

item (3.3). Assumptions about causal relations between number of commonly read or modified artefacts and a resulting intensified relationship cannot be supported. What can be concluded is that common modify events lead to a stronger cooperative relationship than common read events, whereas the amount of commonly modified documents does not neccessarily lead to an intensified cooperation. Those findings will be taken into account in the design of the visualization of mediated working relationships (see section 6.3).

Questions 3 and 4 were concerned with discovering the potential that a cooperative work network visualization would have. The majority of both the groups agrees that a visualization of a user-artefact network would improve the overview of the workspace. An even greater majority (60% of the eProfessionals and 75% of the students) imagine that a visualization would enhance the perception of the type and degree of their working relations. Therefore, it is interpreted that the visualization has the potential to enhance the workspace awareness (see section 2.2.2) of the members of the workspace.

Another goal of the analysis was to find out about the requirements of the desired layout of the visualization. Do the subjects prefer an ego-centered view or a topology-oriented view that is layed out according to topology rules of social network analysis, e.g. that places the node with the highest degree into the center of the workspace? Here, the results are rather ambiguous and the number of subjects that are undecided is rather high. A preference of one or the other layout therefore cannot be concluded. The design of the visualization tool will therefore feature both types of networks, a topology-oriented user-artefact network and an ego-centered network of mediated working relationships between peers.

5.4 Critical Assessment of the Requirements Analysis

The problem of capturing requirements by means of a questionnaire is that you can only capture what you ask. The Pygmalion effect or Rosenthal effect [Rosenthal and Jacobson, 1968] describes the way in which subjects internalise the expectations of the observer. Rosenthal posited that the observer's biased expectancies can create self-fulfilling prophecies. This general observer-expectancy effect that is inherent in any survey, observation or experiment is furthermore complicated by the problem of asking subjects to use their imagination to picture a visualization and then ask them about the potential of change. Of course, this is will lead to subjectively biased answers, as there is no common stimulus material. Subjects seemed to also be irritated by being asked to imagine a visualization. This is supported by their answers on the comprehensiveness of the questions in the questionnaire (see fig. 5.3). Half of the eProfessionals find the questions either moderate or difficult to comprehend. Exceeding this by far, 78% of students thought the questions were moderate or difficult to understand.

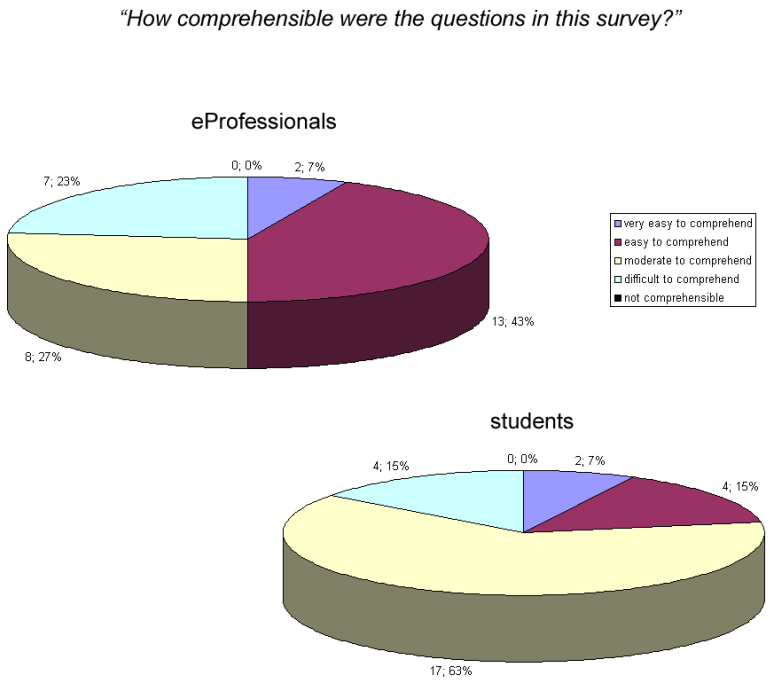

Fig. 5.3. Comprehensiveness of the questions asked in the requirements analysis.

This is supported by free comments that the subjects were able to give at the end of the questionnaire. Four students and six eProfessionals explicitly stated that questions three and four (the "imagine a visualization..."-questions) were difficult to understand, two persons complained that the questions were suggestive.

For future requirements analysis, it is advisable to use participatory requirements analysis techniques, where the users consult with the designers to capture the requirements in cooperation. Thereby, the focus should be shifted from technichal requirements gathering as a mechanistic approach to the organisational and social factors in design in a process called requirements engineering [Thomas, 1996].

> "For CSCW this means that requirements analysis is focused on precisely those factors that are central to the support of cooperative work through technology, with the additional possibility that cooperation itself can be made a part of the requirements analysis process" [Thomas, 1996, pg. 1].

Ethnographic techniques of requirements analysis [Dix et al., 2003, pg. 470] are an alternative (or an enriching amendment) to the participatory approach, with the essential difference that the analysis takes place in the real work context of the users. This approach caters for the situatedness of activity and instead of decontextualized sessions with the users in the designer's venue, the desingers visit the users during their everyday work and infer requirements from their everyday tasks and activities.

6

Concept and Design of the Visualization Tool

The idea of designing and implementing a visualization tool to support networks of collaborative work arose out of thoughts about a problem space. The current viewport onto BSCW emphasizes the artefacts of work in a static folder hierarchy, rather than the people that cooperate in the workspace. As the members of shared workspaces conduct their activities in the workspace, a hitherto invisible network of interrelated activites emerge that spans across the hierarchical boundaries of the folder structure. In its essence, cooperation is characterized by people that engage in a joint enterprise to achieve common goals, their patterns of interrelated activity can easily be conceived as constructing a social network. The study of networks that emerge from cooperation within a shared workspace is assumed to give important insights on chronologic workspace evolution, work patterns and actors' interrelated activity. That interrelated nature will give insights about the quality of cooperation by depicting e.g., closeness of two actors because they jointly contributed to documents. Mediated interpersonal networks can be deducted from the joint activities on common documents, whereby the type of activity indicates the degree of the mediated working relationship. A network perspective onto BSCW is conceived to provide a rapid overview of the activities in a shared workspace. The visualization tool allows flexible sense-making of a network by providing real-time interaction components that affords dynamic filtering, highlighting and other means of visual meaning-making.

This chapter outlines the concept and design rationale of the visualization tool. Scenarios indicate how the tool could be used in a real world context. The design of a user-artefact network for BSCW will define the network, introduce the visual properties of the network and the interaction with the visualization tool. In addition, the approach to discovering and extracting mediated interpersonal networks by inferring mediated working relationships from joint activites on shared artefacts is elicited.

6.1 Scenario

Figure 6.1 shows the use case of awareness functionality in BSCW. A user that employs the BSCW to support his cooperative work can satisfy his need for information about events on the artefacts of work (*workspace awareness*, see section 2.2.2) by several distinct functions, e.g. by the "Daily Report", that the user can subscribe to and then is emailed to him. Typically, a user which is involved in cooperative projects starts his working day by scimming the "Daily Report" for changes in the folders of his workspaces that he is most concerned with, and if he sees that a new artefact has been created in the folder(s) of his interest in the time of his absence, he follows the link in the Report to the new artefact. An alternative is to simply browse the workspace and check the "Awareness Icons" situated right next to the artefact in the folder structure of BSCW that inform the user of events that happened in the meantime. However, this requires that the user confirms that he noticed the events every time he uses the BSCW, otherwise, the icons remain unchanged and lose their information value. If the user is especially interested in the evolution of a certain artefact, or has not visited the BSCW for a long time, he can call the "History" for each artefact. This is especially useful to get an overview of contributors to or readers of one artefact. The functionality "Readers", which displays a list of what artefacts have been read, created or modified by a certain user supports workspace awareness as well.

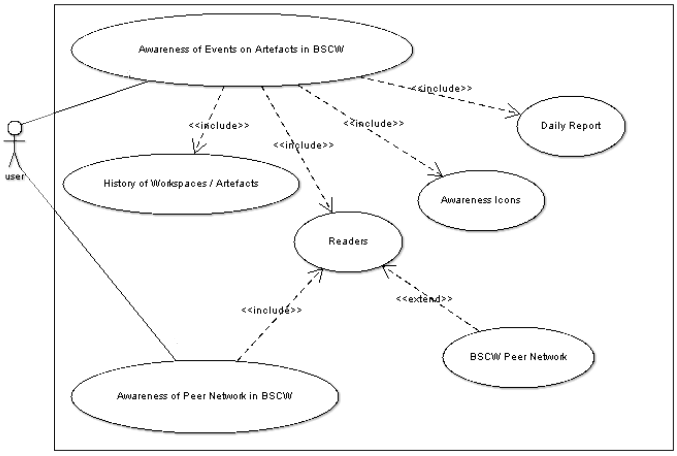

Fig. 6.1. Use of awareness functionality to enhance different types of awareness.

The developed visualization tool "BSCW Peer Network" can be seen as an extension to the existing "Readers" functionality. The visualization can be regarded as another tool providing workspace awareness. However, it is the only functionality that supports the individual's awareness of his peer

network, a type of *social awareness* (see section 2.2.2). As peers are valuable carriers of knowledge, it can be crucial in the everyday context of work to identify people with a similar background or expertise in a different field. A scenario may look like this: The next face-to-face meeting for the team that is concerned with the evaluation of the project's tools is scheduled next week. You have to prepare a presentation on the methodology of evaluation. You have problems to identify a starting point of your presentation, as you did not meet the co-workers personally yet, you do not know what information you can consider known or unknown. A while ago, you have created several documents in several folders of the BSCW that, if all participants were aware of them, had established a common ground of understanding. Instead of browsing through the folders and calling the history for each document, you prefer an overview of who has read your documents at a glance. As an alternative to the list that is presented in the "Readers" view, you call the visualization of the "BSCW Peer Network" for the topmost folder in which you have created a document. You can see at a glance which user nodes are connected to the documents you created and can thus determine whether enough common ground has been established between you and the co-workers in the team.

As it has been indicated above, BSCW is heavily used in teaching and learning. As a lecturer, it is valuable to have an overview of the recent work done in the shared workspace. Again, BSCW caters for that through its awareness functionality that has been introduced above. In addition, it may be neccessary for the lecturer to control the nature of collaboration of the peer group, e.g. in order to assess the contribution of each individual. Figure 6.2 differs from the former use case only by the user's goal. Where in the former use case the user's goal was to collect information to foster workspace or social awareness to support his work, in this case the goal of the user, which may e.g. be a lecturer or a project manager, is to control the collaboration among the members of the workspace (e.g. students). It is commonplace for a lecturer to assess the individual's contribution to a mutually created project space. Often, the lecturer's awareness of the nature of contribution may be hindred by group effects. The visualization tool may also be used to enhance the lecturer's understanding of the groups' individual's contribution to the project in a way that is not mediated by effects of human communication such as sympathy or the perception of one student as a free rider. A result may not only be a negative assessment of an individual but a targeted motivational verbal address that enhances the group's perseverance of its goals in the whole.

6.2 Design of a User-Artefact Network for BSCW

A straightforward way to describe the users' work in BSCW from a graph- or network-oriented perspective is a user-artefact network. The entities of the network are the users and the artefacts of work. The relations between the

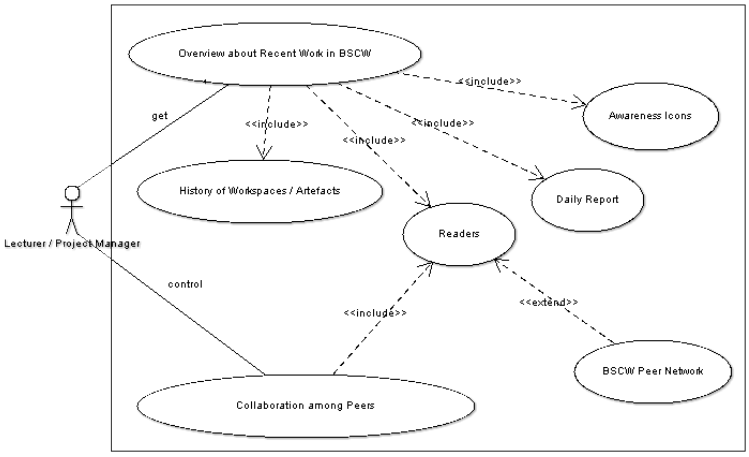

Fig. 6.2. Use of awareness functionality to control collaboration among peers.

entities are the events that the user causes on the artefacts, or to term it more user-oriented, the user's actions on the artefacts. This section outlines the design of the interactive visualization of the user-artefact network beginning with a formal network type definition. The different types of relations between users and artefacts that are supported by the visualization will be characterized. We will then evolve toward user-centered requirements that enable the interactive exploration of the visualized network. What visual aids are necessary for the user to grasp the network's quality? Therefore, the visual possibilities of assigning meaning to a graph will be exploited, such as layout, color, shape, size and labels. Meaning making can be enhanced by the user's interaction with the visual appearance - the means to filter, highlight and extract certain subsets of the graph will be looked into.

6.2.1 Definition of the User-Artefact Network

The visualization is based on the events that are logged by BSCW. Those events are always events that a user causes on an artefact. In the graph, the event is represented by an edge that connects a user and an artefact. If we interpret this from a graph-theoretical perspective, the edges link only vertices from two disjoint sets. In this view, there can never be an edge between two users or between two artefacts. Therefore, the resulting Graph:

$$G = (V, E)$$

where, G is the graph and E a set of edges and V a set of vertices with the restriction of a partition:

$$V = V_1 \cup V_2$$

and every edge in E is of the form v_1v_2 for some v_1 in V_1 and v_2 in V_2, so that the graph is a bipartite graph of the form:

$$G = (V_1 + V_2, E)$$

Weisstein [2002] adds that a bipartite graph is a special form of a k-partite graph, where $k = 2$.

The two partitions are represented by a set of users and a set of artefacts, respectively. The edges that connect the vertices represent three different types of events: *read*, *create* and *modify* events. In more detail, the *read* events are caused by users that browse through the BSCW workspaces and read (or download) documents. The *create* event is caused by uploading documents to the server, the *modify* event comprises operations such as replacing an existing document by a new one, uploading a new version of a document (versioning), renaming documents or revising meta-information of objects (see also: [Appelt, 2001, pg. 340]).

6.2.2 Visual Properties of a User-Artefact Network

Given that a graph consists of only vertices and edges, one has to think carefully about the visual mechanisms of attaching meaning to the picture.

Nodes

Two distinct colors are used for nodes representing artefacts and nodes representing users. If one or more nodes are selected to e.g. change their position or highlight their neighbors, their color changes from red to orange for users and from blue to cyan for documents. Optionally for the user, the size of the nodes will be dependent on the degree of the node. The authors de Nooy et al. [2005, pg. 63] state that "The degree of a vertex is the number of lines incident with it." For each vertex, the degree will also be displayed in a tool tip, when the user hovers over the vertex with the mouse. Also, vertices with higher degree are more likely to be found within dense sections of the network. This relates to the topology (layout) of the graph, which we will look into further down. Each node has a label that either gives the (full) name of the user or the name of the document. When the user decides to highlight all the users that are connected via one intermediate artefact to a certain user, the stroke of the connected user nodes is shown bold.

Edges

The edges have colors that distinguish the event type that they represent, too. Read events are usually by far the most common events (also: see below), therefore an unobtrusive color is picked. Create and modify events are much

less often and therefore are represented by more obbvious color. In addition, each edge carries a label which shows its type. A tool tip displays the date of the event. The length of the edges is a result of the layout algorithm.

Layout

The spring-embedder simulation concept has been successfully applied to graph topology, as in the algorithm of Fruchterman and Reingold [1991]. de Nooy et al. [2005, pg. 16] state that their tool *Pajek* uses spring-embedder influenced algorithms as well to lay out its graphs. It is based on the notion of a mechanical system of springs (the equivalent to the edges in the graph) and steel rings (the eqivalent to nodes), where a balance between attractive and repulsive forces keeps the system stable. A layout based upon a spring-embedder algorithm also places the nodes with the highest degree into the center of the graph. The algorithm used to lay out the graph will be discussed in more detail in section 7.5.1. A layout according to the topology of a network facilitates visual orientation: nodes that are connected by more edges are placed closer to each other, high centrality of a node (by its degree) will place the node into the center of the visualized network, weakly connected nodes are placed in the peripery of the network.

6.2.3 Interaction with the Visualization

The GUI is implemented in Java Swing, the Swing framework offers the state-of-the-art GUI support for Java applications and applets. By creating its leigtweight components by itself, the frawework supplies platform independentness. Thereby, platform dependent specifica of implementing for different operating systems are avoided and the user finds the applet in the same look-and-feel and interaction manner on all possible platforms [Krüger, 2001, pg. 740].

The usability of a graph visualization tool depends on the flexibility it provides, given that its task is to visualize sparse graphs with few elements as well as densely populated graphs with maybe hundreds of elements. Therefore, the user's interaction needs to be supported to:

1. dynamically filter out edges and nodes according to the users goals,
2. dynamically highlight aspects of the graph by changing the appearance of elements,
3. dynamically view a desired part of the graph by magnifying, zooming, panning and centering of the graph.

Filtering

The visualized workspace may have a lot of users and artefacts which results in a densely cluttered picture. For the user to make sense of the picture, he

needs to be supported to concentrate his view according to his goals. Since the edges represent three different types of events, the user will be offered means to filter out either type of events or a combination of them. For instance, if the user decides to filter out the common read events, the initially dense picture will lighten up. If you filter out the edges in the graph, the result will be a lot of vertices that are not connected to edges any more. The amount of nodes could be distracting for the user. Therefore, the GUI will include a check box that hides those orphaned nodes. The layout will position the nodes with the fewest edges at the periphery of the graph. Often, graphs contain a lot of nodes that have only a few edges, especially in the case of a user-artefact network for BSCW. A document that was read only once is connected by two edges, which represent its create event and the read event. To lighten up a densely populated graph radically, the GUI offers a check box to filter out the nodes which are connected by less than three events.

As indicated above, every environment has its way of documenting its history. This insight has also been transferred to electronic settings such as the BSCW as a requirement for situated action [Pankoke-Babatz, 2003]. BSCW workspaces may have a history of several years, of which only a period may be interesting to the user. Also, the history of a workspace watched in fast motion may give the user insights about peaks and valleys of activity. To accomodate these insights on the role of history, the GUI offers a date slider which acts as a filter for the date of an event. An event that happened after the date the user picks on the slider will be filtered out. Here, the user can adjust the slider to view the workspace at a certain date in the past. The reason why a slider was picked and not, for example buttons, is that a slider is a quasi-analogous interaction component - the user decides about the speed he moves the slider at, getting immediate feedback. Sliders offer the best interaction for the user to experience the dynamic character of an evolving workspace. This design trait is also influenced by the notions of dynamic network analysis, which criticises that network visualizations are often only snapshots of a network at a certain point in time, but lack the ability to capture the dynamic nature of networks [Carley, 2003].

Furthermore, the user may not be interested in the events that are too far in the past. Therefore, he can adjust the lower boundary of accepted events by means of a calendar spinner, which accepts dates between the date of the first event of the workspace and today. If an event happened before the picked date, it will be filtered out. As the slider sets the upper boundary of events that are accepted, this date serves as a counterpart. By combining the two, the user can pick any period in the past and see what happenend then in the workspace.

Highlighting

Looking at a visualized graph one often does not see at once which nodes are the most active (in case of a user) or most popular (in case of a document).

To emphasize the most connected nodes, the GUI offers a check box that expands the size of every node in relation to its degree (see also 6.2.2). The user will most often only call a visualization of a workspace that he is himself a member of. It may be his interest to see not only the entire workspace, but to become aware of a subset of users that is connected to each other via an intermediate node. It may be his goal to become aware of the users that he is connected to. Therefore, the user can pick a node and check the box labeled "highlight users that are connected via one artefact", and all the nodes that are connected to the picked one(s) will be highlighted with a bold stroke.

View

The view of the graph is dynamic in several aspects. It allows the user to zoom in and out of the picture with the scroll wheel of the mouse or with the button provided at the GUI. The whole graph can be panned by clicking and dragging when *mouse mode* is set to "transform", or single nodes or parts of the graph can be moved around the drawing area after selecting them by picking with the mouse or drawing a rectangle around them with the mouse when *mouse mode* is set to "picking". Alternatively, the user can select an *examination lens* that can be dragged with the mouse across the graph, which magnifies either the layout or the view of the graph, depending on which option the user chooses from the respective radion button menu. The lens adheres to the focus & context principle of information visualization: to display the central data at the focal point at full size and detail and display the surrounding information to contextualize the relation[1].

6.3 Discovering Mediated Interpersonal Networks

The identification and visualization of interpersonal networks that underlie user-artefact networks is a central concern of this thesis. It was hypothesised in this work that different degrees and types of working relations exist that can be identified and visualized as a cognitive aid for the collaborative worker's self- and network awareness. In the following, I will develop the idea of the types and degree of working relations that are possible between users. These ideas will be underlying the design of an inference mechanism that identifies a mediated interpersonal network from any given user-artefact network.

6.3.1 Types of Working Relations

The types of working relations are easy to identify, at least when underlying the simplified three-partite model of "read", "create" and "modify" events. From the BSCW events, we can only conclude mediated relations between

[1] see for example: `http://www.usabilityfirst.com`

users, since BSCW is a bipartite network where only explicit relations between users and artefacts exist, but never explicitly between users. This is rather a technical affordance than based on empirical reality. For instance, it is possible to send an e-mail to one or more users from the workspace, but this is not logged in the permanent event storage (true!!?). Therefore, the relations between users always have an intermediate artefact element which means that a relationship between any two users always consists of two parts. Following the three-partite model, between any two users the following bipartite relationships are possible:

1. *create-read relationship*
2. *create-modify relationship*
3. *read-read relationship*
4. *read-modify relationship*
5. *modify-modify relationship*

Figure 6.3 shows a typical bipartite relationship. The number and combination of the types of relations is arbitrary and theoretically unlimited. With the restriction that the "create" event is always the first event on an artefact, therefore this has to be mirrored in the first bipartite working relation. The types of working relation result in the degree of the working relation, which we will look at in the next section.

Fig. 6.3. A bipartite relationship: user *bscw-admin* has created a document *hypergraph_378.xml* that user *user2* has read.

6.3.2 Degree of Working Relations

Until this point, the visualization has been developed straightforward, meaning that the visual characteristics represent the "real" BSCW events. The degree of working relations however, can only be inferred from the combination of types of working relations. The most common relation between a user and an artefact is a "read" event. An evaluation of 5.9 million server accesses over a period of 220 working days and the resulting log file have revealed that the vast majority of more than 72% of operations have been read operations [Appelt, 2001, pg. 339]. Of the remaining 28% of all the used functionality only 24,7% are create operations and only 12,4% are modify operations (ibid., pg. 340).

Due to the vast amount of read events, it doesn't seem right to lay too much importance or closeness on a *read-read* or *create-read* relationship between two

users. In collaborative work, a lot of activity is reading the peers work, rather than truly collaborating in creating a piece of work commonly. This intuitive assumption is mirrored by the empirical requirements analysis. The statement "If I *read* documents of another person in BSCW, then a direct relation to this person exists in my working network." was agreed upon by 46% (16% agreed strongly) of the eProfessionals and only 32% of the students agreed (7% agreed strongly). Nonetheless, the agreement is too apparent to neglect that these kinds of relationships link users together.

Comparing the answers to this statement with the answers to the statement where only one word is changed: "If I *modify* documents of another person in BSCW, then a direct relation to this person exists in my working network." the differences are apparent. A vast majority of 73% of the eProfessionals agrees (even 40% agree strongly) and a majority of 58% of the students agrees (14% agree strongly). Apparently, the bipartite working relationships including a modify event weigh more. The statement "I work closer with people whose documents I modify than with people whose documents I read." directly compares the subjective impact of read and modify relations on the individual's working network. Out of the group of eProfessionals, 59% agree with that statement (33% agree strongly) but only four persons disagree. A slight majority of the students of 55% agree too (11% agree strongly), whereas 25% disagree.

The comparison of the two event types suggests that the degree of working relations is stronger if there exists a modify-relation in the bipartite relationship. Experiences with BSCW show that a user would hardly modify another users artefacts without his (even implicit) agreement.

It is quite a difficult task to order the bipartite relationships identified above into an ordinal scale. From what the requirements analysis suggests, we can derive that *create-read* and *read-read* relationships are working relationships, but rather weak ones. A *read-modify* relationship cannot be valued strongly either, because this would result in a lot of strong relationships between users that simply read the document and one user who modified it. Bipartite relations that do suggest a strong working relationship are *modify-modify* and *create-modify* relationships. To gradate these two is a difficult task and would be too much speculation. For instance, a pair of users may have agreed upon that one user uploads an artefact that contains all of his contributions already and the other user makes his subsequently. In another case, two users may have agreed to change the artefact gradually, each user modifying the existing version. How can you be sure which of the two approaches to collaborative work implies a stronger relationship?

Concluding the findings and thoughts about type and degree of working relationships, I have decided on a simple binary mapping of the five types of relationships to the degrees *weak* and *strong*. In this case, degree seems an odd expression because it suggests at least ordinal scale, but in absence of any better terms that are not too vague, *degree* will be the term that refers to

Table 6.1. Assignment of types of working relationships to degree of working relationships

	degree of working relationships	
	weak	strong
types of working relationships	*create-read relationship* *read-read relationship* *read-modify relationship*	*create-modify relationship* *modify-modify relationship*

this nominal scale. Table 6.1 shows the binary mapping of types of working relationships to the degree of working relationships.

6.3.3 Topology of the Mediated Interpersonal Network

When underlying the mentioned bipartite working relations to the discovery of the mediated social ties it is clear that the original graph holds many of these kinds of derived graph - one graph for each user. Thus it is clear that a holistic view as in the above described user-artefact network is no longer feasible. Also, it is assumed that the user is especially interested in his or her own mediated interpersonal network and not particularly in the other ones. The requirements analysis supports that assumption. It revealed that out of the eProfessionals 80% favor a view that centers the node representing the current user (where 20% agree strongly). 55% of the students favor this egocentric view.

An egocentric view requires that the visualization has some knowledge about the current user. Since the visualization is XML-based and the XML file does not hold information about the current user, the visualization has to get the user to discover the mediated network for as a parameter at run time. For this, the user can pick a user name from a drop-down list entitled "Personal Network" for who the mediated interpersonal network will be visualized.

6.3.4 Extracting the Mediated Interpersonal Network

Starting with the holistic user-artefact network, how can we derive the set of users that are connected to each other via at most one intermediate artefact? JUNG provides mechanisms to assemble sets of vertices that are connected to a certain vertex. Figure 6.4 shows the process of extracting a subset of connected users (red nodes). Since our starting point a bipartite user-artefact network, the set of successors `v_succs` that we derive for the selected user will contain only artefacts (blue nodes). For this set of artefacts, we then have to assemble the successors for each of the artefacts by iterating through the set and retrieving the successors for each artefact - the arrow indicates that for each of the four artefacts a set `a_succs` has to be discovered. The set(s) `a_succs` contain all the vertices connected to the selected vertex via at most one intermediate artefact; plus itself. For the case of the sample graph in figure

6.4, all of the vertices are connected to the user *bscw-admin*. Now the vertices can be copied into the new graph, that is, the mediated social graph.

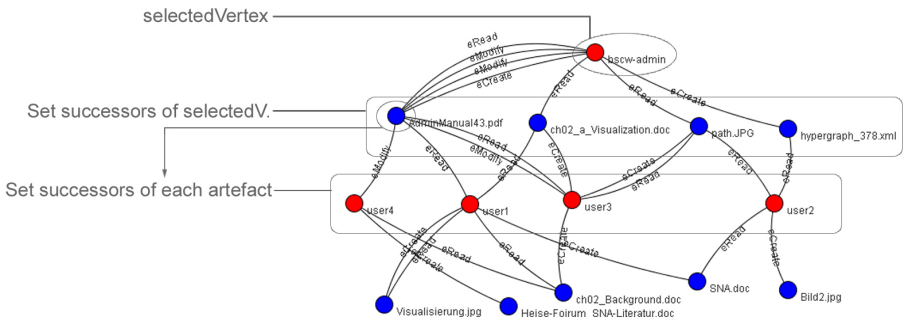

Fig. 6.4. The process of extracting a subset of connectected users.

Similarly as for the vertices, the edges are extracted. An algorithm (see 7.5.5) computes the edge sets from the original user-artefact network that exist between two user vertices, that is, a bipartite relation through an intermediate artefact vertex. The bipartite relationships between two users are mapped to the the degree of working relationships as indicated above. To do this, the two parts of the relationship are evaluated. If one of the edge part types is a create or modify relation, the part's user datum "social tie" is set to *strong*, otherwise it is set to *weak*. Then the other part of the bipartite relationship is evaluated in the same manner. Finally, the two parts are processed together, deciding the degree of the new user-user relationship in the mediated network. The new relationship only becomes *strong* if the two parts of the bipartite relationship are strong, meaning that the bipartite relationship has to be a *modify-modify* or *create-modify* relationship. In the resulting visualization of the mediated relationship (see figure 6.5), the selected vertex is centerd and the edges' strokes are grey if there is a weak social tie, otherwise, the edge stroke is bold and green. Additionally, the edges are labelled as either "weak: read same document" or "strong: jointly modified document", and a tool tip explains the origin of the mediated social relationships.

6.3.5 Inferring Common Interest

SNA-techniques can be used to analyze relationships in networks to infer common interest. Schwartz and Wood [1993] present an approach which deduces shared-interest relationhships between people based on the history of email communication. The approach to inferring common interest applied here is somewhat similar, as it also is a "white pages" problem, providing suport for

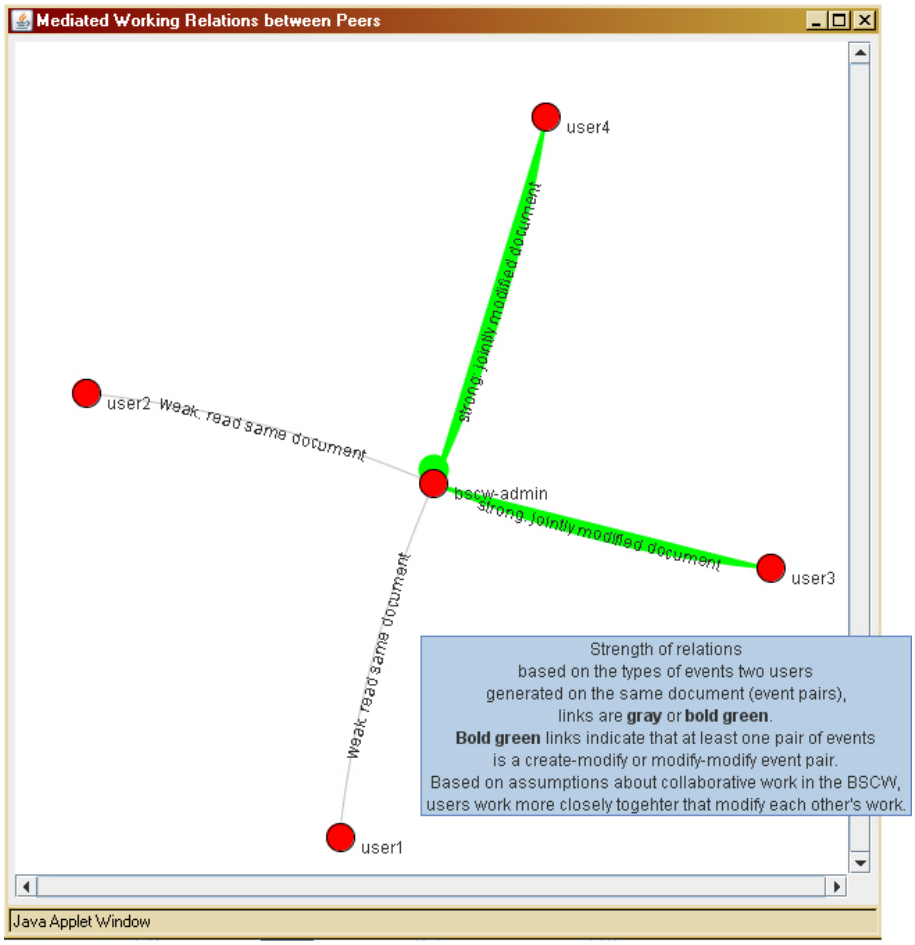

Fig. 6.5. The mediated interpersonal network with tool tip.

locating particular users, as opposed to the "yellow pages" problem, the problem of finding users with a particular interest or expertise. The approach to common interest employed here is inspired by the notion of *common ground* (see 2.1.4) that represents a joint awareness of artefacts of work or mutual knowledge. An indicator of common interest in BSCW is the number of common documents. This number itself however, is meaningless, as for a user that has relations to 500 documents 50 common documents are less telling as for a user that has 50 documents and shares the awareness of all of these with another user. However, the percentage of common documents of a pair of users to all documents of a certain user is an indication of common interest, so that a common interest index can be computed for each pair of users:

$$CII = sd/ad$$

where CII is the common interest index, sd are the number of shared docuements and ad are the total number of documents of the user the CII is computed for.
In plain, this relation is the percentage of shared documents with another user of all documents a user has. This functionality expresses its power where visual orientation fails becuase the network consists of hundreds of nodes and edges. Figure 6.6 shows the table of common interest for user Wolfgang Prinz for the entire ECOSPACE project folder.

Common Interest Indices for all 580 Documents of user: Wolfgang Prinz

User Name	# of Total Documents	# of Common Documents	Common Interest Index (in %)
Elke Hinrichs	835	410	70,69
Christian Seeling	448	238	41,034
Marc Pallot	473	228	39,31
Hans Schaffers	405	190	32,759
Santiago Cáceres	278	153	26,379
Hermann Loeh	417	144	24,828
Kjetil Kristensen	262	125	21,552
Michael Vonrüden	561	125	21,552
Tunde Kallai	194	121	20,862
Erik Spin	177	114	19,655
Matteo Villa	235	111	19,138
Antonio F. Gómez Skarmeta	193	100	17,241
Emilio García			
sergio GUSMEROLI			
Mª Antonia Martínez Carreras			
Viktor Kaufman	229	87	15
Stefan Decker	117	83	14,31
Konrad Kloeckner	165	82	14,138
Rudolf Ruland	267	82	14,138
Vassilios Peristeras	206	81	13,966
FINA DANIELA	137	79	13,621
Encarna Pastor	151	79	13,621
Steffen Budweg	212	79	13,621
Marco Di Girolamo	104	76	13,103

Common Interest is computed for the selected user based on the number of documents in common (read/create/modify-events) with another user in relation to all documents of the selected user.

Fig. 6.6. Common Interest Indices for a selected user.

7

Specification of the Prototype

Here, I will outline the architecture of the implementation and point out the main characteristics of the developed BSCW add-on visualization tool referred to as "BSCW Peer Network". Instead of an exhaustive description of all the components involved in the functionality, the description will focus on the main parts of the architecture.

7.1 Extending BSCW with packages

The concept of BSCW propagates separation of logic functionality and UI appearance. That makes the extension of the functionality fairly easy. The BSCW's functionality can be extended by packages. Packages all have the same basic structure. The structure of the packages mirrors the concept of separation of concerns. On the top level, there are usually the folders

- `messages`, which contain the language dependent part of the user interface in English and German. Depending on which language the user set his browser to, the UI is either displayed in English or German.
- `resources`, contain the icons that are displayed in the UI and style sheets that assign the layout of the site.
- `src`, contains the source code written in Python, responsible for the operational logic of the functionality. This folder also contains interface `templates` in XHTML that are language independent UI templates.

7.2 Starting Point: The package "Readers"

As it has been mentioned, the BSCW already provides the functionality to visualize workspaces as graphs in a HyperGraph Java applet (see 4.2.1). The needed BSCW functionality of extracting the events and writing them into an XML file is already there, so only little of the package has to be altered

to accommodate for the visualization in the newly developed "BSCW Peer Network".

"Readers" contains three main operations, `op_readers`, `op_get_graph` and `op_map`. Usually, the user will not type in the operation directly into the address of the browser, so he sets out by choosing the arrow right to the objects in the regular BSCW where all of the functionality then becomes visible to the user in a pop-up menu. Then he selects "Readers" which calls `op_readers` for the current workspace object. This object is dispatched as a request object to `op_readers`. In the next view, the user can display lists of documents that have been either read, modified of created by a certain user, that the current user can pick from a drop-down list that contains all the users of the current workspace. Secondly, the user can display an "event map" of either the documents he just selected or of all the documents in this folder. The operations behind this call is what calls the visualization. First, the button click on "show" calls `op_map`, and again hands over the request object – the current workspace – to the operation. `op_map` calls the `op_get_graph`, which is responsible for writing and returning the GraphXML file representation of the graph and writes it to the temporary file repository of the BSCW. Within `op_get_graph` the methods `write_graph`, `write_node` and `write_edge` are called. These methods identify the passed in arguments as nodes if they are users or documents or as edges if they are events and then write the representation according to the GraphXML file format specification.

7.3 GraphXML format as interface

XML, the *Extensible Markup Language*, has become a valuable data representation format, especially because it provides a homogeneous way for web-applications that exchange data among various services in a platform-independent manner [Hein and Zeller, 2003, pg. 24]. XML [W3C, 2006] is a specification developed by the World Wide Web Consortium and "its goal is to enable generic SGML to be served, received and processed on the Web in the way that is now possible with HTML." [Bray et al., 2006]. XML makes it possible to describe arbitrary data in human-readable form. This is advantageous when system infrastructures are coupled by employing XML to interchange data because it makes the interface between system parts transparent. [Hein and Zeller, 2003, pg. 24]. XML's extensible character, its capability to define arbitray data structures according to a syntax definition of the language through its *Document Type Definition* (DTD) has lead to the specification of many XML dialects. GraphXML is a graph description language in XML, and the authors say its goal "is to provide a general interchange format for graph drawing and visualization systems, and to connect those systems to other applications." [Herman and Marshall, 2000, pg. 1]. Listing 7.1 shows a GraphXML representation as it is generated by the Python operation `get_graph`. This file then is stored temporarily and gets parsed by the custom

Java XML handler (see 7.4.2) to visualize the graph. The prologue defines encoding type and points to the location of the *GraphXML.dtd* which defines the syntax of the content of the file. The root element `<GraphXML>` includes the XLink attribute that can be used similar to the anchor-tag in HTML to point to a different URL. The `<graph>` element encapsulates the actual information on the graph. It has an `id`-attribute which is not processed, but the `isDirected`-attribute is important, if not specified, the parser will return an error. GraphXML allows the definition of a `<style>` element, but other than for the implementation of the event map in HyperGraph, our Java code defines the style through the `class`-attribute of the following `<node>` element. Here, the nodes are defined. The `class`-attribute either has the value `"nUser"` or `"nDoc"`. The `name`-attribute of the nodes is the internal BSCW object reference that is always unique. The `label`-attribute is the human-readable form of the user name and the document name, as seen in the BSCW UI. The `<data>` element can be used to point to an URL, in this case, to the BSCW information site "more on...", that exists for each document and user in BSCW. However, this is not processed yet by the visualization tool and therefore can be omitted.

In the next paragraph, the `<edge>` element is introduced. It requires the specification of the `source` and the `target` of the edge as attributes – its values are the respective user- and document-nodes. Additionally, the attribute class distinguish the type of events in the BSCW – it takes either a `"eCreate"`, `"eModify"` or `"eRead"`-value. Finally, the element has an attribute `date` in the format `"YYYY-MM-DD HH:MM"` which is logged by BSCW for each event and also written to this file.

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE GraphXML SYSTEM
3  "http://localhost:8080/bscw_resources/jung/GraphXML.dtd">
4  <GraphXML xmlns:xlink="http://www.w3.org/1999/xlink">
5  <graph id="BSCW_Graph" isDirected="false">
6
7  <node class="nUser" name="33" label="bscw-admin">
8    <data>http://localhost:8080/bscw/bscw.cgi/33?op=inf</data>
9  </node>
10 <node class="nUser" name="162" label="user3">
11   <data>http://localhost:8080/bscw/bscw.cgi/162?op=inf</data>
12 </node>
13 <node class="nDoc" name="311" label="hypergraph_378.xml">
14   <data>http://localhost:8080/bscw/bscw.cgi/311?op=inf</data>
15 </node>
16 <node class="nDoc" name="304" label="path.JPG">
17   <data>http://localhost:8080/bscw/bscw.cgi/304?op=inf</data>
18 </node>
19 <node class="nDoc" name="243" label="AdminManual43.pdf">
20   <data>http://localhost:8080/bscw/bscw.cgi/243?op=inf</data>
21 </node>
22 <node class="nUser" name="121" label="user2">
23   <data>http://localhost:8080/bscw/bscw.cgi/121?op=inf</data>
24 </node>
25
26 <edge class="eRead" source="121" target="311"
27 date="2007-09-26 14:47"/>
28 <edge class="eCreate" source="33" target="311"
```

```
date="2007-09-19 19:51"/>
<edge class="eCreate" source="33" target="243"
date="2007-09-10 16:56"/>
<edge class="eModify" source="33" target="243"
date="2007-09-10 17:12"/>
<edge class="eRead" source="162" target="304"
date="2007-09-19 19:49"/>
<edge class="eRead" source="33" target="304"
date="2007-09-19 19:51"/>
<edge class="eRead" source="162" target="243"
date="2007-09-15 12:08"/>
<edge class="eCreate" source="162" target="304"
date="2007-09-19 19:49"/>
<edge class="eModify" source="162" target="243"
date="2007-09-28 21:26"/>
<edge class="eCreate" source="121" target="340"
date="2007-09-26 14:48"/>
<edge class="eRead" source="33" target="243"
date="2007-09-10 17:11"/>
<edge class="eRead" source="121" target="304"
date="2007-09-26 14:47"/>

</graph>
</GraphXML>
```

Listing 7.1. Sample GraphXML code describing relations between three users and three documents

GraphXML allows a lot more elements such as `<style>` that describe the graphical elements and also, e.g. the specification of geometrical data. Since layout is a concern of the Java code in this implementation, we do not require more than these few elements and attributes.

The graph representation format GraphXML chosen provides the interface between the native BSCW source code written in Python and the visualization written in Java. This avoids complicated data structure transformations between the two languages and additionally, it renders the visualization as XML-based.

7.4 Parsing the GraphXML data

This section outlines the approach to parsing the input data for the Java applet. Initially, the two common approaches of parsing an XML document and their Java implementations, event-oriented and tree-oriented parsing are discussed. Then the chosen parsing approach and the implementation of the custom parser is described.

7.4.1 Event-Oriented vs. Tree-Oriented XML parsing

The event-oriented parser reads the document into a stream and decomposes it into its elements that are then forwarded to the application one by one, whereby the parser validates the syntactic correctness of the document according to its DTD. Hein and Zeller [2003, pg. 78] state that this approach

is advantageous because the document doesn't have to be kept in the main store in its whole, the document is being read from one element to the next.

The tree-oriented parser reads in the whole document and transforms it into a data structure that reflects the tree-like structure of the XML document. This means that the whole document will be kept in the main store, which consumes more capacities. The main argument against using a tree-oriented parser lies elsewhere: the data structure represented by the parsed document may not be optimal for the application that processes the data – that implies that the structure has to be re-copied to reflect the desired structure. This increases programming effort, time and storage capacity. Since the main elements of the visualization – graph, vertices and edges – have to be created one after another, the tree-oriented approach does not seem feasible for our requirements. Instead, we will rely on an event-oriented parser that uses the readily available Java API *Simple API for XML* (SAX).

The Java package `org.xml.sax` includes an implementation of `DefaultHandler`, that provides the most important parsing operations through its implemented interfaces `ContentHandler`, `ErrorHandler`, `EntityResolver` and `DTDHandler`. All we need to do for our visualization applet is to write a custom parser that extends `DefaultHandler`.

7.4.2 Custom GraphXML Parser

Figure 7.1 shows the class diagram of the custom GraphXML Parser, the `BipartiteGraphXMLFileHandler`. Basically, it gets the GraphXML file as an input argument to the method `load()`, does all the parsing of the XML elements by calling the method `startElement()` then creates the graph, vertices and edges and annotates them with the necessary data, and returns the `KPartiteGraph` object to the calling method, the `getGraph()` method from the main applet class that the user interacts with.

The custom extension to `DefaultHandler` does not simply create a JUNG `Graph` object, it creates a `KPartiteGraph` object where the vertices are allocated to two distinct partitions depending on if they represent artefacts or users. This is necessary as the algorithms that execute the desired transformations to a graph according to its mediated relations require a k-partite, in this case, bipartite graph object. This accomodates to the design requirements outlined in the previous chapter.

Listing 7.2 shows the method `load(String filename)` which is called by the main Java class of the applet and returns the fully processed `KPartiteGraph` object. First, a new `SAXParserFactory` instance is needed, then a new `SAXParser` instance is returned by the method `newSaxParser()`. The method `parse(filename, DefaultHandler)` takes in the temporary filename that is passed to the `load(filename)` method on invocation and the instance of `BipartiteGraphXMLFileHandler` that extends the `DefaultHandler` as arguments. This method in turn invokes `startElement()` which does the main parsing and is detailed in the following. Finally, the `KPartiteGraph` object is

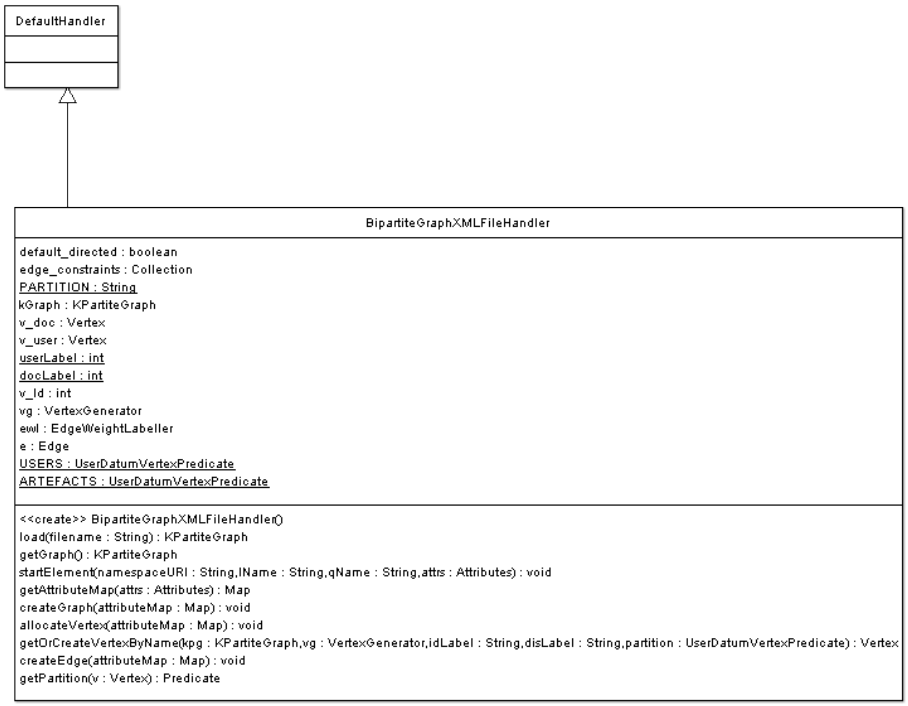

Fig. 7.1. Class diagram of the custom GraphXML parser.

returned to the Java applet's class.

```
public KPartiteGraph load(String filename) {

    // Use the default (non-validating) parser
    SAXParserFactory factory = SAXParserFactory.newInstance();
    try {
        // Parse the input
        SAXParser saxParser = factory.newSAXParser();
        saxParser.parse(filename, this);

    } catch (Exception e) {
        throw new FatalException
        ("Error loading graphml file: " + filename, e);
    }

    return this.getGraph();
}
```

Listing 7.2. The method `load(String filename)`.

Listing 7.3 shows the inherited and extended method `startElement()`. Its task is to parse the document by creating an `attributeMap` of the contents of the file, and as long as the the qualified name (the element tags `<graph>`, `<node>` and `<edge>` in the GraphXML file) equals one of the three elements

in the GraphXML file, passes the relevant part of the map to the methods that create the major JUNG elements that are visualized: `KPartiteGraph`, several objects of the type `Vertex` as prescribed by the number of `<node>` elements in the GraphXML file, and several `Edge` objects connecting the vertices. For the creation of vertices, first `allocateVertex()` is called which determines the partition for each vertex, then the vertex is created by calling `getOrCreateVertexByName()` (see figure 7.1).

```
public void startElement(
        String namespaceURI,
        String lName, // local name
        String qName, // qualified name
        Attributes attrs) throws SAXException {

    Map attributeMap = getAttributeMap(attrs);

    if (qName.toLowerCase().equals("graph")) {
        createGraph(attributeMap);

    } else if (qName.toLowerCase().equals("node")) {
        allocateVertex(attributeMap);

    }
    else if (qName.toLowerCase().equals("edge")) {
        createEdge(attributeMap);
    }
}
```

Listing 7.3. The method `startElement()`.

As indicated above, the main applet class only has to create an instance of this custom parser and call its `load()` method to get returned a graph that it can display and apply all kinds of changes to its visual appearance, such as filtering out certain elements. Visualization and interaction is the responsibility of the applet's main class, which will be looked into now.

7.5 The visualization applet

Figure 7.2 shows the class diagram of the main applet class. The implementation extends JApplet and overrides its `init()` function to get the GraphXML filename from the HTML tag `<param name="file" value="%%(graph)s">` from the template file *map.xhtml*. After `init()` is called by the browser on load of the applet code the browser calls `start()`. Here, the applet's `ContentPane` is called and the main `JPanel` is added to the applet by calling `startFunction()` which returns the parent `JPanel`. The `startFunction()` instantiates all the necessary JUNG instances to visualize the graph in the UI. An instance of `PluggableRenderer` and an instance of `FRLayout` that takes in the graph as an argument is needed to create an instance of the `VisualizationViewer` which is in fact the main drawing area. Before `startFunction()` returns, the UI-controls are added to the bottom of the

panel and all the action listeners (for buttons) and change listeners (for the sliders), that define the behavior of the controls are registered.

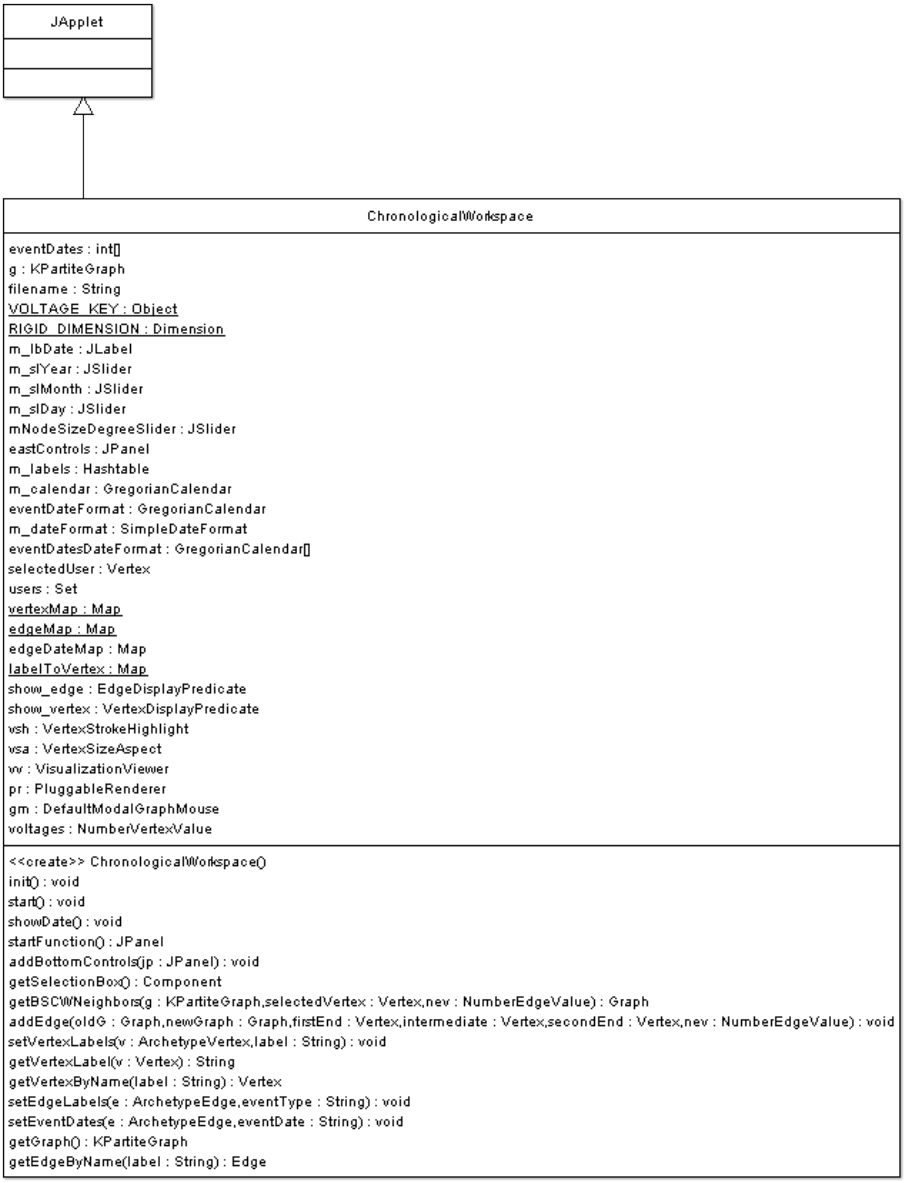

Fig. 7.2. Class diagram of the main applet class.

In the following, I will outline the most important instances of the applet class and some selected methods that reflect the design goals made in the previous chapter on filtering and selection of a subgraph that represents the mediated relations between users.

7.5.1 Force-directed layout: The Fruchterman-Reingold algorithm

The chosen `FRLayout` lays out the nodes according to the Fruchterman-Reingold algorithm [Fruchterman and Reingold, 1991]. JUNG provides an implementation of that algorithm that is easy to use and lays out our undirected bipartite graph neatly. The algorithm is based on the spring-embedder simulation concept, where nodes are replaced by steel rings and edges by springs to reflect the nature of a mechanical system. Within this system, "repulsive forces are calculated between every pair of vertices, but attractive forces are calculated only between neighbours" [Fruchterman and Reingold, 1991, pg. 1130]. The algorithm is concentrated on undirected graphs and aims at:

1. Distribute the vertices evenly in the frame.
2. Minimize edge crossings.
3. Make edge lengths uniform.
4. Reflect inherent symmetry.
5. Conform to the frame.

[ibid., pg. 1129]

Also, the authors state that their goal is to produce an aesthetically-pleasing two dimensional picture of a graph. Implementing the algorithm, they concentrate on speed and simplicity and come to the conclusion that their algorithm achieves "interactive speed" [ibid., pg. 1161].

7.5.2 The renderer

The renderer's implementation `PluggableRenderer` main task is to determine what to draw and how the elements are shaped. JUNG's implementation provides all sorts of buttons to press and dials to turn. As a developer, you can override the default properties for the paint, stroke, size and shape of the vertices and edges as well as attach labels to them for the user to read. For our implementation, we set the labels for vertices and edges, set the paint function that defines the colours for vertices and edges (also its change when they are picked), set the `VertexStrokeFunction` that is used to highlight vertices that are connected via at most one intermediate element and set the `VertexShapeFunction` that is used to enlarge the vertices according to their degree (the number of edges that connect them). Also, the renderer is responsible for filtering of chosen elements. Since the user shall be able to filter out edges and the then orphaned vertices, the renderer's `EdgeIncludePredicate` and its `VertexIncludePredicate` is set as well.

7.5.3 The drawing area: `VisualizationViewer`

The class `VisualizationViewer` extends a `JPanel` and represents the main drawing area. Here, the listener components concerned with user interactions have to register. Among them is `DefaultModalGraphMouse`, a JUNG component that implements the functionality of listening to mouse events (click, wheel and motion). Tool tips, which the user gets when hovering with the mouse over the drawing area or over a UI-component are also set here. `VisualizationViewer` finally gets put into a `GraphZoomScrollPane`, that enables zooming, scrolling and panning of the visualization.

7.5.4 Chronological Workspace

Although JUNG offers a large number of convenient classes, without which the richness of functionality of the visualization tool could not have been achieved, quite a few modifications, extensions and own contributions to JUNG's source code was necessary to adapt the visualization to the specific design outlined in the previous chapter. Therefore, the implementation of the dynamic chronological and edge filtering mechanism shall receive some attention here.

The filtering of edges relies on the custom implementation of `EdgeDisplayPredicate` that forms an inner class of the main class of the visualization applet. To the method `evaluate()` (see listing 7.4) of this class, every edge is handed to be evaluated if it passes the predicate and thus the renderer knows if it to display the edge or not. At first, we need to get the `"eventDate"` for each edge that was set as a `UserDatum` when the XML file was parsed and trim it so that the Strings represent year, month and day of the event date (lines 4-8). Next, the Strings need to be converted to an integer so that the date can be set as an actual `GregorianCalendar` date (lines 12-15). By calling `getTime();` on this type, a `Date` is returned which is required to neatly process dates (line 17). The `"eventType"` is used to check if the edge is meant by the filter (line 19).

The following three conditions are sufficient to express if an edge shall be filtered out by the renderer. Each condition is comprised of two parts linked by a logical `OR` – only one part of the condition has to be true that the edge is filtered out. Furthermore, the second part of the condition again comprises two parts, one part checks whether the edge does not pass the upper bound and the other part checks whether it does not pass the lower bound. The boolean types `hide_read`, `hide_modify` and `hide_create` are `true`, when the user selected the correspondent check boxes at the UI labelled "filter out read-/modify/create events". Thus, the first part of the condition evaluates to `true` and then filters out the edges, if the button is selected and the edge is of the selected type (lines 21, 28, 35). The second part of the three conditions evaluates to `true` if the date `upper_bound` happened before the `eventDatePerEdge` or if the date `lower_bound` happened after the `eventDatePerEdge` (lines 22-23, 29-30, 36-37). The date `upper_bound` is selected by the user by means

of the date slider at the UI, and the date `lower_bound` is selected at the UI at the date cycler input field labeled "show only events after". This enables the user to limit the visualization to a period of interest as specified in 6.2.3. Edges are returned and thus shown only if the date of the edge is in between the lower and the upper bound.

```
public boolean evaluate(Object arg0) {
  Edge e = (Edge)arg0;
  //get Date for each edge
  String evD = e.getUserDatum("eventDate").toString();
  //trim to convert
  String yearS = evD.substring(0, 4);
  String monthS = evD.substring(5, 7);
  String dayS = evD.substring(8, 10);

  int year, month, day;

  GregorianCalendar eventDateE = new GregorianCalendar();
  eventDateE.set((year = Integer.parseInt(yearS)),
                 (month = Integer.parseInt(monthS))-1,
                 (day = Integer.parseInt(dayS)));

  Date eventDatePerEdge = eventDateE.getTime();

  String eventType = e.getUserDatum("eventType").toString();

  if ( (hide_read && eventType.equals("eRead")) |
       ( (upper_bound.before(eventDatePerEdge)) |
         (lower_bound.after(eventDatePerEdge)) ) ) {

       return false;
  }

  if ( (hide_modify & eventType.equals("eModify")) |
       ( (upper_bound.before(eventDatePerEdge)) |
         (lower_bound.after(eventDatePerEdge)) ) ) {

       return false;
  }

  if ( (hide_create & eventType.equals("eCreate")) |
       ( (upper_bound.before(eventDatePerEdge)) |
         (lower_bound.after(eventDatePerEdge)) ) ) {

       return false;
  }

  else
       return true;
}
```

Listing 7.4. The method `evaluate()` determines if the edge will be filtered out by the renderer.

In analogy to the filtering of edges, the user should be able to specify that *orphaned* vertices should be filtered out as well, when using the means to set the upper and lower bound of the desired period of time. The implementation of the vertex filtering is a more difficult problem. While for the filtering of edges, every edge is evaluated which has only one type (*read*, *create* or *modify*) and date, for the filtering of vertices, each vertex is evaluated, which can

have an arbitrary number of edges of different types and dates. The challenge is to evaluate a vertex not according to its own attributes, but according to the edges' attributes that connect the vertex. Figure 7.3 demonstrates the differences and the arising problems. In general an edge that is to be evaluated if it passes the predicate has only one type and date, the case of *Vertex1* has three events. The chronologically first event that connects the vertex represents a *create* event that happened before the period that the user is interested in, as specified by the lower bound. The next event, a *read* event falls into the period of interest of the user. The third event again, happened outside the range of the desired period, this time after the upper bound. While the edges are filtered out as indicated by the red crosses, the vertex itself is not, because there is still an event that happened within the period of interest of the user. The case of *Vertex2* shows the conditions that have to be met to filter out a vertex. The earliest date of an event of a vertex has to be below the lower bound *and* the latest date of the event has to be above the upper bound *and* the vertex may not be connected by events within the period of interest of the user. The entire code of the filtering mechanism for vertices is considered too long to be discussed in detail here, the interested reader can check the source code of `VertexDisplayPredicate` in the `ChronologicalWorkspace` class on the enclosed CD.

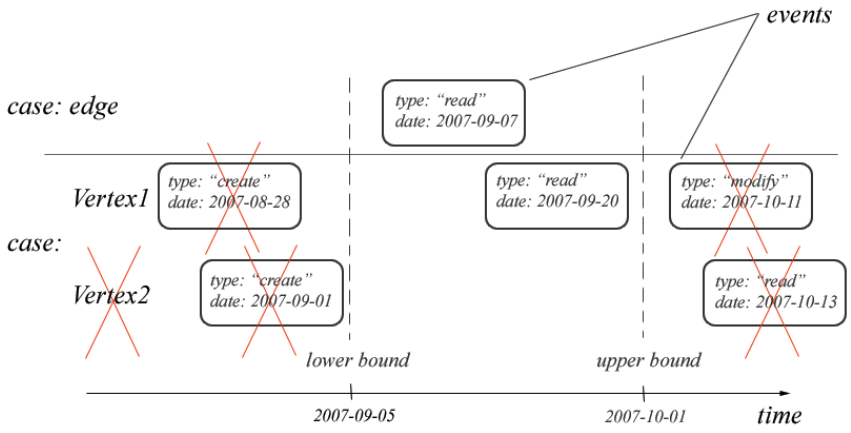

Fig. 7.3. Filtering mechanism for edges in comparison to vertices.

7.5.5 Mediated Interpersonal Network

The user can display a mediated interpersonal network for a user from the current workspace that is selected in the drop-down list entitled "Personal Network" at the UI. This function will pop up a new window with a derived subset of the original graph that shows the mediated social relationships between BSCW users. The algorithm that discovers the social relationships is split up in two main steps or methods. Listing 7.5 shows the first step in discovering the social network – the method `getBSCWNeighbors()` is responsible for finding the vertices that are connected to the selected vertex via at most one intermediate artefact, or more specific: by a vertex that represents an artefact. The algorithm is inspired by JUNG's `FoldingTransformer`, a transformation algorithm that the documentation describes as follows:

> "A class for creating a "folded" graph based on a k-partite graph or a hypergraph. A "folded" graph is derived from a k-partite graph by identifying a partition of vertices which will become the vertices of the new graph, copying these vertices into the new graph, and then connecting those vertices whose original analogues were connected indirectly through elements of other partitions" [Fisher and O'Madadhain, 2007].

Before we look at the second step of the algorithm that discovers the type of relations that connect the users, let us walk through what the first step does – the method `getBSCWNeighbors()`. This method is responsible for discovering the three vertices that are connected in a bipartite working relationship as described in 6.3.1. It identifies the two users and the intermediate artefact. On call, the method receives the entire original `KPartiteGraph` and the `Vertex` that the user has selected from the drop-down list. The `NumberEdgeValue` can be used optionally, but as yet is not used further in this implementation. An empty new `Graph` is created first (line 5). The `selectedVertex` is the first vertex that is copied into the new graph (lines 8-10). Then this vertex is assigned to be `v_new` (line 11). As described in 6.3.4, the set of successors (all the vertices that are connected to the `selectedVertex`) is assembled (line 13), which can only contain the nodes representing the artefacts of work, since we are dealing with a bipartite graph. This set is iterated upon to extract the single vertices (lines 15-16), which represent the intermediate artefact `s` that mediates the interpersonal relationship.

The next step is to derive a set of the artefact's successors (line 18), which in turn, following the bipartite logic, contains only vertices representing users – the users the initial selected user (represented by `selectedVertex`) is connected to via an intermediate artefact. The artefacts are extracted from the set in another loop (lines 20-21) – these vertices are then copied into the new graph as `t_new`, one by one with each iteration through the loop (lines 27-28). To add the eges representing the mediated working relationship to the new graph representing the mediated interpersonal network, the original graph,

the new graph, the two users the new edge must connect and the intermediate artefact are passed as arguments to the method `addEdge()`.

```
protected Graph getBSCWNeighbors (KPartiteGraph g,
                                  Vertex selectedVertex,
                                  NumberEdgeValue nev) {

  Graph newGraph = new UndirectedSparseGraph();

  // get vertices for the specified partition, copy into new graph
  if ( (Vertex) selectedVertex.getEqualVertex(newGraph) == null){
    selectedVertex.copy(newGraph);
  }
    Vertex v_new = (Vertex) selectedVertex.getEqualVertex(newGraph);

    Set succs = selectedVertex.getSuccessors();

    for (Iterator s_iter = succs.iterator(); s_iter.hasNext();) {
        Vertex s = (Vertex) s_iter.next();

        Set s_succs = s.getSuccessors();

        for (Iterator t_iter = s_succs.iterator(); t_iter.hasNext();) {
            Vertex t = (Vertex) t_iter.next();

            // if t is in the partition of interest
            // and has not been covered (undirected graphs only)
            if (!users.contains(t)) continue;

            if ((Vertex) t.getEqualVertex(newGraph) == null ) {
                t.copy(newGraph);
            }

            Vertex t_new = (Vertex) t.getEqualVertex(newGraph);

            addEdge (g, newGraph, v_new, s, t_new, nev );
        }
    }

        return newGraph;
}
```

Listing 7.5. The method `getBSCWNeighbors()` is the first step in finding the mediated social relations.

The method `addEdge()` in listing 7.6 completes the discovery of the individual user's mediated interpersonal network. It creates a new edge between two users that represents a bipartite working relationship and assigns to the new edge the degree of the mediated working relationship. The original user-artefact graph is needed to extact the equivalent three vertices (lines 8-10), which are connected in the bipartite relationship. The original vertices are needed because they carry the required meta data tag `"eventType"`. The set of edges (because parallel edges are possible) that link the first user vertex (`old1`) and the artefact (`old2`) is needed (line 17) to continue with the evaluation of the first part of the bipartite relationship (lines 18-36). The loop iterates through the set of edges and gets the edge type for each edge. If an edge represents a read event, the first part of the bipartite relation, the `UserDatum "socialTie"` is set to *weak*. If the edge represents a create or

modify event, the first part of the bipartite relation is set to *strong* and a `boolean isStrong1` is set to `true` and the evaluation is stopped. The algorithm repeats the process for the second half of the bipartite relation (lines 38-58).

The new edge, which represents the mediated working relationship that connects the two users in the mediated interpersonal network is created in (lines 60-65). Finally, the two halves of the original bipartite relation are evaluated (lines 72-91), which have either been set to *strong* if representing a *create* or *modify* event, or set to *weak* if representing a *read* event. Since we iterate through the edges and the final edge `v_t` that represents the bipartite relationship could originally have consisted of multiple parallel edges, it is possible that a degree is already assigned to the edge. This has to be checked (line 68) to make sure not to override a degree *strong* with a degree *weak*. If no degree has been assigned to the edge yet, it is done now (lines 72-77). Only if both parts evaluate to `true` the degree *strong* is assigned to the new edge (lines 76, 77). If a degree has been assigned in a previous iteration, the degree should only be changed if it has been set to *weak* (lines 82-90). Before it iterates through the next bipartite relationship, the boolean values are reset (lines 97, 98).

```
1  protected void addEdge(Graph oldG,
2                         Graph newGraph,
3                         Vertex firstEnd,
4                         Vertex intermediate,
5                         Vertex secondEnd,
6                         NumberEdgeValue nev) {
7
8    Vertex old1 = (Vertex)firstEnd.getEqualVertex(oldG);
9    Vertex old2 = (Vertex)intermediate.getEqualVertex(oldG);
10   Vertex old3 = (Vertex)secondEnd.getEqualVertex(oldG);
11
12   Edge oldE1 = null;
13   Edge oldE2 = null;
14
15   boolean isStrong1 = false;
16
17   Set edgesOld1 = old1.findEdgeSet(old2);
18   if (edgesOld1 != null)
19       for (Iterator oE_iter = edgesOld1.iterator();
20            oE_iter.hasNext(); ) {
21           oldE1 = (Edge) oE_iter.next();
22           String edgeType = (String) oldE1.getUserDatum("eventType");
23
24           if (edgeType.equals("eRead"))  {
25               oldE1.setUserDatum("socialTie", "weak", UserData.SHARED);
26           }
27           if ((edgeType.equals("eCreate")) |
28               (edgeType.equals("eModify")) ) {
29               oldE1.setUserDatum
30               ("socialTie", "strong", UserData.SHARED);
31               isStrong1 = true;
32               break;
33           }
34           if (oldE1.getUserDatum("socialTie").equals("strong"))
35               break;
36       }
37
```

```
38    boolean isStrong2 = false;
39
40    Set edgesOld2 = old2.findEdgeSet(old3);
41    if (edgesOld2 != null)
42        for (Iterator oE_iter = edgesOld2.iterator();
43             oE_iter.hasNext(); ) {
44            oldE2 = (Edge) oE_iter.next();
45            String edgeType = (String) oldE2.getUserDatum("eventType");
46            if (edgeType.equals("eRead"))  {
47                oldE2.setUserDatum("socialTie", "weak", UserData.SHARED);
48            }
49            if ((edgeType.equals("eCreate")) |
50                (edgeType.equals("eModify")) ) {
51                oldE2.setUserDatum
52                ("socialTie", "strong", UserData.SHARED);
53                isStrong2 = true;
54                break;
55            }
56            if (oldE2.getUserDatum("socialTie").equals("strong"))
57                break;
58        }
59
60    Edge v_t = firstEnd.findEdge(secondEnd);
61
62    if (v_t == null) {
63        v_t = new UndirectedSparseEdge(firstEnd, secondEnd);
64        newGraph.addEdge(v_t);
65    }
66
67    try {
68        String sT = (String) v_t.getUserDatum("socialTie");
69    }
70    catch (NullPointerException e) {
71
72        if (!isStrong1 | !isStrong2)
73            v_t.setUserDatum("socialTie", "weak", UserData.SHARED);
74        else if ((!isStrong1 & isStrong2) | (isStrong1 & !isStrong2))
75            v_t.setUserDatum("socialTie", "weak", UserData.SHARED);
76        else if  (isStrong1 & isStrong2)
77            v_t.setUserDatum("socialTie", "strong", UserData.SHARED);
78    }
79
80    String sT2 = (String) v_t.getUserDatum("socialTie");
81
82    if (sT2.equals("weak")) {
83        if (!isStrong1 | !isStrong2) {
84            v_t.setUserDatum("socialTie", "weak", UserData.SHARED);
85        }
86        else if ((!isStrong1 & isStrong2) | (isStrong1 & !isStrong2)) {
87            v_t.setUserDatum("socialTie", "weak", UserData.SHARED);
88        }
89        else if  (isStrong1 & isStrong2) {
90            v_t.setUserDatum("socialTie", "strong", UserData.SHARED);
91        }
92    }
93    else if (sT2.equals("strong")) {
94    // do nothing
95    }
96
97    isStrong1 = false;
98    isStrong2 = false;
99
100 }
```

Listing 7.6. The method `addEdge()` discovers mediated social relations between users.

7.6 Summary

This chapter outlined the implementation of the prototype as a package to the server directory of BSCW. The enclosed CD contains the extended package `readers`, now called `NetVis`, and provides the ready-to-install visualization tool to any local BSCW installation version 4.3 and above. The structure of the package adheres to the conventions of BSCW packages[1] and contains the following directories:

- `messages`, which contain the language dependent part of the user interface in English and German. Depending on which language the user set his browser to, the functionality is either displayed in English or German at the UI.
- `resources`, contains the main contribution in the folder `jung`. A README-file explains how to install the software. The extended JUNG jar-archive contains the Java source code and compiled code, documentation and examples and my extensions to JUNG. The commented source code of the developed Java applet (>1500 lines) ChronologicalWorkspace.java and the custom XML-parser can be found in packages/MyNetwork/src in the jung-1.7.6 directory.
- `src`, contains the native BSCW source code written in Python by Rudolf Ruland, responsible for creating the GraphXML-file as interface for the Java applet. This folder also contains interface `templates` in XHTML that are language independent UI templates.

[1] according to the FIT-internal BSCW 4 Developers' Tutorial by Thomas Kreifelts and Rudolf Ruland

8

Evaluation

The design and implementation of technological artefacts is an interventionistic practice that is conducted according to goals of the designer to support or enable certain tasks or processes of the users. The intended implications and the yielded effects of an implemented technology on the user however, often diverge. Technological interventions often lead to unintended effects that emerge from the usage of the tool, or more generally speaking, from the implementation of a technology. Such interdependent effects of technological artefacts and society have been reported to be inherently political [Winner, 1986]. Suchman [1994, pg. 178] adopts Winner's notion of inherently political artefacts to coordination technologies in CSCW and observers an agenda of discipline and control over organizations member's actions by superimposing a certain theory as a foundation for system design. If it was not through a process of reflection on technology, such consequences of technology-in-use would remain obscured.

From a pragmatic point of view, an evaluation gives important insights of the tool's added value for the user. Here, the designer is provided with feedback from the users that is important information for the process of iterative redesign of a tool. Does the user's perspective of the tool's affordances mirror the designer's goals of supporting the desired process or task? Which scenarios of use are conceivable for the user? Prior to an evaluation of a tool, the "Why, How and What to Evaluate" must be clarified before the evaluation process begins [Andriessen, 1996, pg. 107]. This chapter on evaluation begins with the introduction of the *What to Evaluate* by pinpointing the level of analysis according to Neale et al. [2004]. The approach is further elicited by stating success criteria and research questions of the evaluation. The section on methods clarifies the *How to Evaluate* and establishes the applied method of ethnographic expert interviews. In the section findings, the results of the evaluation are presented and discussed.

8.1 Evaluation Approaches and Level of Analysis

Neale et al. [2004, pg. 114] introduce three general different approaches to evaluation of CSCW systems. *Methodology-oriented frameworks* outline the general methods available to CSCW systems but they provide little guidance for choosing among the available methods to study the chosen research questions. *Conceptual CSCW frameworks* describe the group factors that should be subject of the evaluation. However, approaches of this category usually fail to map methods that can be used to evaluate the conceptual constructs sufficiently. "*Concept-oriented frameworks* focus on specific aspects of group behavior, such as communication or coordination" [Neale et al., 2004, pg. 114] and how methods can bed used to measure concepts like awareness. The authors continue to define levels of anlysis that help the evaluator to choose an adequate approach for evaluation. The levels of anlysis they suggest are the individual level where one user interacts with a system, the group/team level, the organizational and the industrial level. Despite being integrated in a groupware, the visualization tool developed and evaluated in this thesis is a single-user tool, it provides the individual with awareness information. Apart from questions of usability, the concern of the evaluation is to evaluate the potential of the tool as an awareness information provider. Therefore, the level of analysis is the individual level and the chosen approach is concept-oriented. In addition, the identification of possible scenarios of everyday use of the tool is strived for, accomodating the insight of emergence of unintended usage patterns of technology-in-use.

8.2 Research Questions and Success Criteria

The overall goal of the evaluation will be to test if the visualization tool enhances the user's awareness: Do the perspectives on the collaborative space and the interaction means provided by the tool construct an enhanced awareness of relations of actors and their activities on artefacts and the user's individual position in the working context? As this is hard to assess directly, the achievement of more specific success criteria pointed out in the following are regarded as an operationalization of enhanced awareness.

In general, the visualization tool features two different approaches of providing awareness information. First, it visualizes actors and their activities on shared artefacts. This is a direct representation of actors and their events as it is propagated by other awareness mechansims such as the *Daily Report* or the *History* of artefacts. The goal and success criteria is that the visualization tool provides visually explorable perspectives that can be used effectively and satisfyingly to give rapid oversight of actors and their activities on shared artefacts and answer more specific, user-formulated tasks such as:

- Which time period is characterized by a lot of activity in the workspace? What kind of activity (creation, reading or modification of artefacts) was prevalent in a certain time?
- How did the workspace evolve chronologically over time? Which artefacts have been there for a long time, which users have been active for a long time, which artefacts and users are recent ones?
- Which are the most central artefacts and the most active users in a workspace?
- Which users are aware of the same artefacts as a specific user?
- Are there islands of unconnected user-artefact subgroups in a workspace?
- Where is a specific user positioned in the user-artefact network and how is the relation to other user's activities? Are they more or less or equally active? How does the amount of artefacts a specific user is aware of compare to the other users?

Note that the *specific user* in the above formulation of tasks can be the current user of the tool if he is a member of the workspace he visualizes. Or it can be any other user. Thus, the tool can be used to support the individual user's tasks by answering questions for himself, and/or it can be used as an analytic tool of general oversight of workspace activity. It will be subject of the evaluation to find out what kind of the above tasks the users discover themselves and which perspective the users prefer to take on.

In order to fulfill tasks with the tool, visual orientation is an important trait. Does the visualized picture give rapid oversight of the workspace events? The layout algorithm places nodes in a meaningful way onto the drawing area. But is the user aware of that? Does the resulting image of the network mirror the member's subjective perception of his working network? Or is the impact of the communication channels that are unaccounted for, such as face-to-face or email communication, as significant that the network view is obsolete because it does not reflect the co-workers subjective perception of his network? As Nomura et al. [1998, pg. 20] emphasize, a shared workspace

> "(...) indicates each members actions done inside the workspace and ignores the actions done outside."

Second, the visualization tool provides indirect information that is inferred from relationships between actors mediated by the intermediate artefacts. Into this category falls the functionality that is called *Personal Network* and *Common Interest* at the UI. This functionality is based upon assumptions and does not represent (the technological) reality directly. With respect to those types of functionality, it is essential to evaluate how well the visualization's image of the personal network and the table of common interest map the individual's notion of his personal network and the co-workers he shares interests with. This kind of indirect information is provided by the integration of SNA-techniques. Therefore, a side-effect of the evaluation will also be the assessment of the meaningfulness and chances of the integration of SNA-techniques into an awareness information providing tool.

To identify emergent usage patterns, the user will not be told prior to the interview which tasks and goals the tool supports, because this is the author's and designer's subjective perspective. Purpose of a tool is individually constructed on the basis of bodily bound knowledge and prior experiences. Therefore, the user will be asked which purpose the tool takes on in his perception, what kind of information the tool conveys in his opinion, and will be asked to state a scenario in which he uses the tool, this will give important insights on the user constructed purpose of the tool and the goals and tasks it supports.

8.2.1 Usability Questions

Usability is defined as the extent to which a product can be used by a user in a specified context to achieve specified goals efficiently, effectively and satisfyingly in the ISO norm 9241 on *Ergonomic Requirements for Office Work with Visual Dispaly Terminals*. ISO 9241-10 on software ergonomic requirements of system design. Part 10 of the standard decomposes usability further into granular ergonomic dialogue principles that are intended to be used in specifications, design and evaluation. The visualization tool is evaluated with respect to the following principles:

- Conformity with user expectations. Before the user starts to interact with the visualization exploratively in the evaluation, he will be asked what he expects the layout and the entities of the visualization to represent and what he expects from the use of each interaction component to happen to the visualization.
- Self descriptiveness / suitability for learning. Tool tips are provided for the less intuitive features of the tool. Furthermore, a help document is provided. The user will be asked during the test of the tool for each feature if it is intuitive, or else if the provided help suffices. The learnability of the tool can be observed during the interaction.
- Controllability. This concerns the direct manipulation of the visualization. Is the user able to control the interaction and able to do what he wants to do? This will be observed and the user will be asked about possible problems with the interaction.

Usually, *suitability for the task* is a well-regarded principle. However, as it has been said earlier, the goal of this evaluation is to discover emergent use cases, including the user's goals and tasks. Several specific user tasks that can be achieved by means of the tool were listed above. However, an important aspect of this evaluation is to find out which tasks the users actually use the tool for. It can very well be that users perceive this tool as an analytic tool that provides oversight of the workspace or that they percieive it as a tool that they use to support their individual tasks. Therefore, suitability for the task is not explicitly evaluated, since no task is prescribed by the tool. Instead, the

scenarios the user conceives in which he uses the tool will reveal the emergent goals and tasks the tool supports.

8.3 Method of Evaluation

The chosen method of evaluation is a face-to-face ethnographic expert interview. The evaluation takes place in the offices of the participating eProfessionals, at their own computer. The sample consists of XX!! eProfessionals that are members of either one or both of the two EU-projects ECOSPACE or CoSpaces in whose shared workspaces in BSCW the tool is tested. This is to cater for an everyday work context of the eProfessionals, so the tool can be tested in an authentic working environment, as proposed in ethnographic methodology [Dix et al., 2003, pg. 470]. The inteviews are voice-recorded for documentation and can be found on the enclosed CD.

8.3.1 Ethnographic Expert Interviews

Experts of their work are interviewed in this qualitiative approach to evaluation. An interview guideline (in German, see Appendix B) structures the course of the interview and the open questions that serve to answer the research questions introduced in the above section. The interview is structured into the following phases:

1. Choosing a workspace. The user is asked to navigate into a workspace that he frequently works in.
2. Locating and initializing the tool. Is the user aware where to find the tool and of the different initializing perspectives the tool affords? For instance, the tool can be called from any point in the hierarchy of the workspace, then the user has the option to visualize all documents in the current workspace (and by default all subordinate workspaces), or only documents created, modified or read by a specific user.
3. User expecations of functionality before exploring. The user will be asked to state what happens according to his expectations when he presses a button or uses a slider.
4. Functionality exploration and think aloud. The user is asked to explore the functions one by one and talk through what he is doing, if the effect of his action adheres to his expectations, what this is telling him, if the function is intuitive or else if the provided help is sufficient. Where appropriate more specific questions on the functionality will be asked, e.g. if the visualized personal network mirrors his subjective perception of the personal network he has in mind for the work in the current workspace.
5. Post exploration discussion. Here, relevant use cases and scenarios are discovered after the user has learned the functionality by exploration. This includes questions on the perceived purpose of the tool, strengths and weaknesses of the tool.

8.4 Findings

The findings for the three eProfessionals that were interviewed each for about 30-40 minutes are subsumed here, whereby the findings are structured according to the chronologic phases of the interiew as described in the last section. Even though one might argue that the number of interviewees even for a qualitative evaluation is small, the surprising level of consistency in the answers recorded suggest that the interpretations (and also: misunderstandings) are quite straightforward and consistent across different individuals. Where appropriate, the user's opinions will be supported by quotations. As the interviews were conducted in German, the quotations are literal translations of the respondents' answers. The interviews were recorded and can be found as MP3 files on the enclosed CD.

8.4.1 Locating and initializing the tool

After having chosen a subordinate workspace of either the ECOSPACE or CoSpaces project BSCW where the participants work on a regular basis, they were asked whether or not they are aware of the functionality "Readers" that provides the starting point for initilizing the visualization applet. Neither of the three interviewees knew where to find the functionality nor what it does. The interviewees hat to be informed that "Readers" provides an overview of who has read the documents created or modified by a special user that can be picked in a drop-down list. If the visualization tool is decided to be integrated in BSCW distributions, one has to think about a more obvious name at the UI, so that users will be able to locate the tool at all. For the purpose of evaluating a prototype, this is not the decisive issue. When being asked if they are interested in either the events on the documents that they have created or on all documents in the folder, 2 responded to be only interested in the events on the docuements they have created, whereas one eProfessional was interested in all events on the documents in the workspace.

8.4.2 Initial Interpretation of the Visualization

When the visualization appeared on the interviewees' screen, he was asked: "What do you see there?" Answers vary interpersonally, but suggest that they all grasped the idea of looking at a network representation of user's activities on shared documents in the workspace they chose, for instance one person said that he sees "co-workers and their access of documents". One falsely interpreted blue and red nodes as different types of users, he suggested they could be "creators" and "readers" of documents, however, when looking more closely, he corrected his mistake without the interviewer's interference. Generally, the user's quickly identified the colors of the nodes and their representations as well as the colors of the edges and that they represent read, create and modify events. Figures **??** show the initial networks for two of the

interviewees as it is layed out by the visualization tool, and figure 8.3 shows the network for interviewee No. 2 with filtered out read events. Notice how islands of unconnected nodes emerge (top right and top left) that indicate lack of collaborative working relationships on these artefacts.

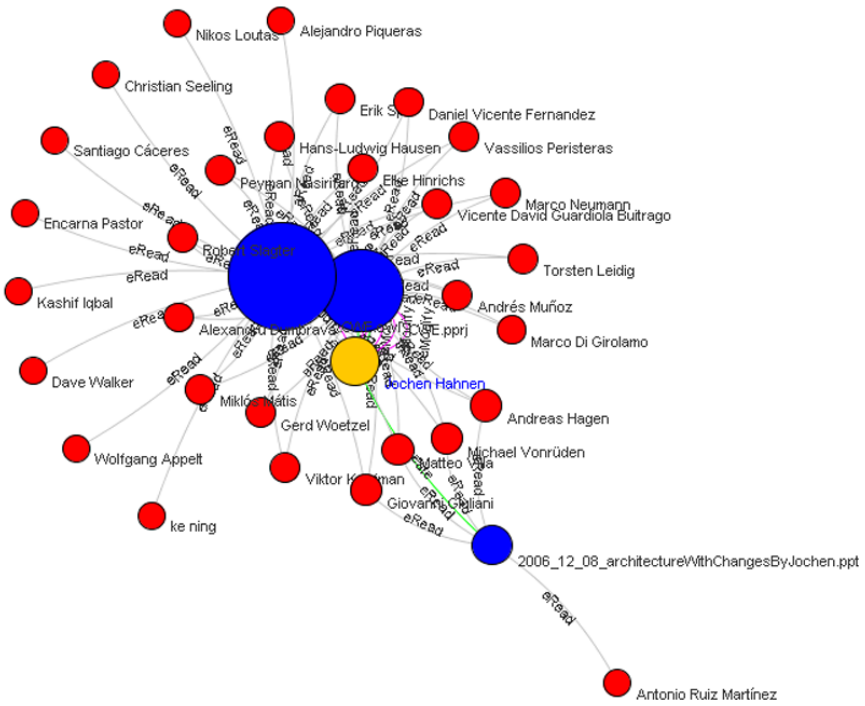

Fig. 8.1. The user-artefact network of interviewee No. 3 as layed out by the visualization. Most connected nodes are emphasized, the interviewee is highlighted in yellow in the network.

The participants were asked what in their opinion the relative positioning of the elements of the graph means. All participants correctly identified the meaning of centrality and periphery of nodes, that more central nodes are characterized by more connections and less distance between two nodes is characterized by the amount of parallel edges between two nodes. The intuitiveness of the layout's meaning suggests that the layout algorithm works well:

> "Obviously, distance is...the amount of events that a user generated" [Interviewee 1]. " Aha, the more events...or the better the relation is, the closer to the center they [the nodes] seem to be (...) This [the relative positioning] could meand that somebody that is closer to the documets has a more intesive relationship to the docuements as some-

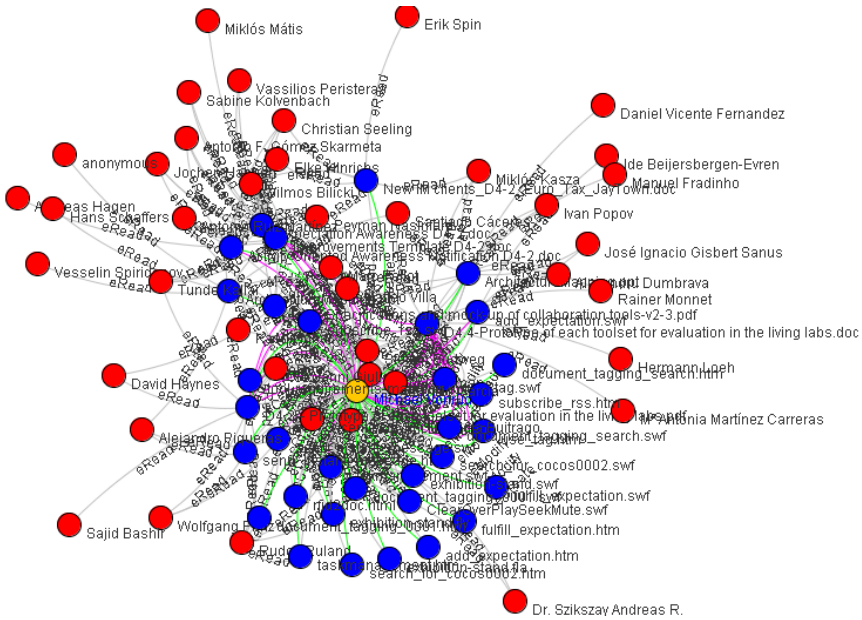

Fig. 8.2. The user artefact network of interviewee No. 1. The interviewee himself is highlighted in yellow in the network.

> body that is farther apart (...)." and he further anticipated "Ah, so then the active persons can be quickly identified, they are in the center then, and the important...or let's say the frequently used documents" [Interviewee 2].

Next, the interviewee was asked to identify the node that represents himself. As the interviewees picked a workspace in which they are very active, their node was placed in the center of the visualization by the alogrithm. One user correctly anticipated that fact when he was asked and correctly assumed his node to be in the center. One participant identified his node before I even asked him to find that node. The third participant found the node after a short time visually searching the picture. Two users missed a search field when I asked them to locate the node that represents themselves. The visualization of those two were cluttered with many nodes, this makes the purely visual identification of nodes cumbersome. However, when using the Zoom function and panned the enlarged, now better readable network, they quickly found what was asked.

The interviewee was then asked if the position of node in the network and the relative position to the other elements represents the picture that they imagine of the workspace's network "in their mind".

> "That [the visualization] definitely represents the picture", one participant said. Another one said: "that's correct that I'm the closest

person to that documents, I mean, I created it, modified it, and the other's read it."

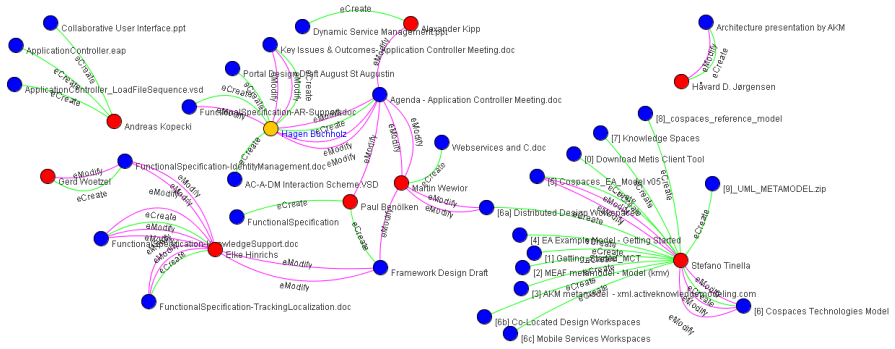

Fig. 8.3. The user-artefact network of interviewee No. 2. The interviewee himself is highlighted in yellow in the network. Read events have been filtered out.

With respect to the orientation within the visualization, the chosen layout algorithm and the color coding of the elements works well for the sizes of workspaces that have been picked, as is suggested by the observed intuitive visual orientation of the participants and the mapping of the layout with the coceived internal representation of a workspace network. Initially cluttered pictures can be zoomed into to lighten up the cluttered areas. A search field to locate users more quickly seems to be a useful additional feature to highlight desired users more quickly.

8.4.3 User Expectations

In order to assess the usability of a tool, it is common to test the conformity of with user expectations. In this phase, the users were asked to state what their expected to happen when they use the interaction components of the UI prior to exploring the interaction. For this phase, it is important that users do not have any prior experiences with the usage of the tool. None of the three interviewees have interacted with the visualization tool before.

Figure 8.4 shows the Java Swing interaction components of the visualization tool. The participants were asked to walk through the interface's components and state their expecations of the effects of using the UI components. If they did not know instantly what the component does, they were asked to read the provided tool tip to see if the provided help suffices.

With regards to the *Filter* interaction components the first three buttons *filter out read/create/modiy events* were intuititve for all three interviewees. They correctly estimated that the effect is that the corresponding edges get

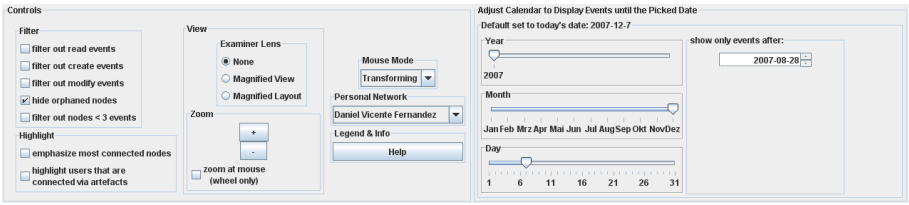

Fig. 8.4. The Java Swing interaction components of the visualization tool's UI.

filtered out. The button *hide orphaned nodes* was less intuitive, especially because the english word *orphaned* is less known among the interviewed native german speakers. However, the provided tool tip, which reads *events have to be filtered out first, then this button hides unconnected nodes* clarified the function of this button in all three cases. The expectations of the effects of button *filter out nodes < 3 events* corresponded to the actual effect in all cases.

The button group *Highlight* provided more difficulities for the interviewees. The first button was less difficult, the interviewess correctly assumed that the appearance of the nodes would change when checking the box entitled *emphasize most connected nodes*, only they did not know whether this would be done by color or size of the nodes. The interviewees were all puzzled about the effects that the checkbox *highlight users that are connected via artefacts* would produce, and when testing the functionality, the interaction proved not to be intuitive and no clear purpose could be assigned to the functionality (this is detailed in the next section).

The expectations of the effects in the button group *View* adhered to the actual effects of the components, except that the interviewees could only guess what the difference between *magnified view* and *magnified layout* is, but they correctly assumed a magnification lens.

The drop-down list *Mouse Mode* lead to correct assumptions that the mouse pointer behavior would change, when picking eiter *Transforming* or *Picking*.

The functionality *Personal Network* invoked false expectations in all interviewees. One responded that they would expect that the picked user from the drop down list would be selected in the current view, another one expected that one's own relationship to the picked person would be displayed. The third expected both of these effects.

The effect of the *Help* button was correctly anticipated. However, one expected a graphical, and not a textual legend.

The slider entitled *adjust calendar to display events until the picked date* and the additional field *show only events after* was correctly identified as another means for filtering, chronological filtering.

8.4.4 Functionality Exploration and Think Aloud

The functionality of the checkboxes arranged in the *Filter* button group was intuitive and controllable for the interviewees, they also considered the functionality as important.

The effect of the checkbox *emphasize most connected nodes* did not only conform to the user's expectations, the interviewees found the functionality helpful. One interviewee stated:

> "I would have expected that...that is great, that the things that are used more often, are displayed larger. That is neat like this."

As indicated in the previous section, the effect of the checkbox *higlight users that are connected via artefacts* was hard for the interviewees to estimate. In addition, the interaction seemed problematic, as this button only works in conjunction with a previously picked user, for which *Mouse Mode* has be set to *Picking*. Despite this precondition being displayed in a tool tip, the users had some trouble to do all necessary interactions to see the effect. As the users managed to see the effect, they correctly interpreted what they see. One user observed:

> "That probably means that this are the users, which...via an artefact...are connected with that selected user."

But despite this correct understanding, the interviewee failed to recognize the added value of this functionality:

> "One only sees who has been connected to him via some artefact, that is not really informative. I would prefer, that when I click on a document, I see who is working on it."

All in all, this functionality was difficult to control and its purpose was not clear to the interviewees. As it represents the same subgroup of users that are connected to a picked user as the Personal Network, this functionality could be omitted in future versions of this tool.

The *Examiner Lens* and the inherent metaphor of a magnification glass itself was understood. However, the interviewees were puzzled about the difference of the two options. As they studied the effect of both options, they tried to interpret more into it than there actually is. In summary, the function has its purpose, but does not need two different options of magnification. In the next, version, this will be omitted. The *Zoom* was frequently used during the participants interaction and is an important functionality especially for the exploration of large, cluttered networks.

JUNG software relies on *Mouse Mode*, as this changes the mouse pointer behavior. *Tranforming* allows panning of the whole graph, *Picking* affords selecting of single nodes and dragging only those on the drawing area. This principle was understood rather quickly by the participants, whereas picking

of nodes led to the assumption that additional information could be displayed for the nodes. One user thought about the value of being able to pick and drag nodes, and arrived at a value for printing networks on paper, so that the network is more readable.

Personal Network

As it has been said, the *Personal Network* visualizes mediated working relationships between peers, as inferred by the common read events on a document and by the common create-modify resp. modify-modify relations on a document. The evaluation aimed to find out if this inferred relationship is understood by the users and if the assumptions underlying the inference are appropriate. As indicated above, the interviewees did not expect that a new window would open, but as they studied the newly popped up window on call, they quickly grasped that they are looking at interpersonal relationships:

> "Ah...there you see the connections between him and all the others, in fact, if they have worked together on a document or if they've just read it."

Furthermore, the edges labels that display either *weak: read same document* or *strong: jointly modified document* lead to the correct interpretation that common read events lead to a weak relationship in this view, and common modify events on a document lead to a strong relationship. When asked whether they think these assumption that lead to either strong or weak relationships are appropriate, one interviewee said:

> "In this context...partly...because I know that I have worked togehter with this person [points at node with strong relationship], but with others, which are indicated as weak here, I have worked together with them a lot, too. Only they have sent me their input in a different way and I integrated their content into the document."

And another interviewee, whose personal network is depicted in figure 8.5, said:

> "Yes, I would say that when somebody modifies a document, that indicates interest in a document...and reading, that happens often, for example [says colleague's name] makes updates frequently and then it looks, as if she had read all the documents, but in reality she has only downloaded it to her hard drive. This is almost meaningless in BSCW. Insofar, these [modify] relations are much stronger, I think that's correct."

Whereas the assumptions that common read events represent weak and common modify events strong relationships are accepted by the interviewees,

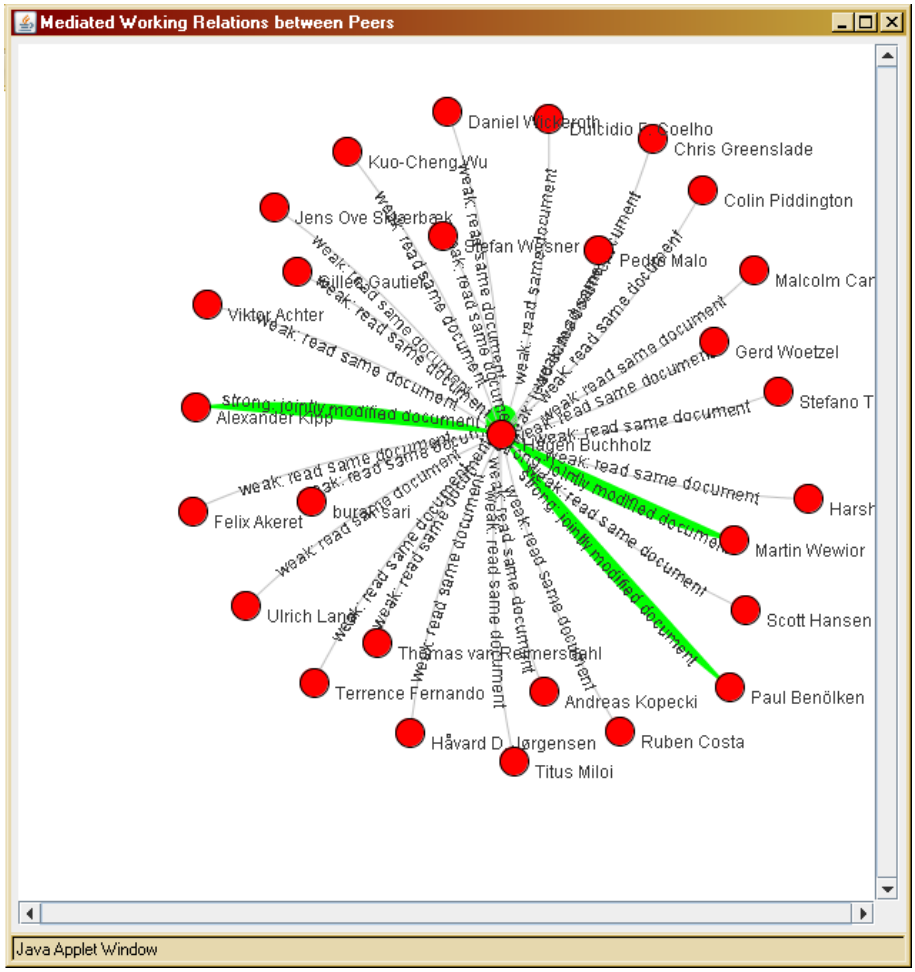

Fig. 8.5. The Personal Network of the chosen workspace of interviewee No. 2.

the two quotes indicate a problem that is inherent in the inferrence of interpersonal relationships of BSCW events: the visualization tool knows only the relationships that exist within the boundaries of BSCW, it has no notion of other employed communication channels that are used within interpersonal relationships. This is furhter emphasized by another statement of interviewee No. 2, when he was asked whether the visualized picture represents the picture of his interpersonal relationships in mind:

> This surprises me. [Lists user names he has strong connection to in visualization]: I had expected some other ones. Those three are absolutely among the ones that I expected, but I expected more...[pauses]. Ahh! [insightful exclamation] The problem in this workspace is, that

> the people work very much via eMail...und we have worked on a document together, but sometime an eMail version was send around, then they were uploaded again...of course, that is the reason why it is so falsified. Some people just can't do it - those three get displayed, that shows you who knows how to use BSCW."

Those findigs suggest that a personal network can only be partly inferred from workspace events. The prevalent work practice includes other channels of communication apart from the groupware. Those are not only used to coordinate or to point to work in the workspace, but also simply, to send work around attached to eMails in the old fashioned way. This has some implications on awareness of cooperative work in general, which will be taken on in the conclusions of this thesis.

Chronological Slider

From the designer's perspective, the chronological slider sets out to capture the dynamic nature of a workspace, and make its evolution visually explorable. All of the three inteviewee's thought this functionality provided an added value. Interviewee No. 1 stated:

> "This gives a good overview, when this document has been created. [moves slider] That must have been between the 6th and 7th November. Interesting to see, who then joined...who has read it from that point of time. Interesting is also...when I move the slider to a deadline...to monitor who actually read what until this deadline."

Interacting with the slider, interviewee No. 2 said:

> "I am moving the slider, at the same time nodes disappear...it's good, that you see it directly."

And when asked about the added value of the slider, he adds:

> "It has an added value because it's interactive. If I want a point of time, on which a certain docuemnt has not been created...then you search...move the slider as far back, until the document disappeared, then the point of creation can easily be filtered out. One could check that through other functionality in BSCW, but like this you can see it interactively, which document was created when and also, which people are in connection with it. I think, this is definitely an added value."

Between the lines of these quotes one can already anticipate the use cases of the visualization tool. It seems to be a good tool to monitor activities as especially the quote of interviewee No. 1 indicates. The next section will deal with possible use cases cases that have been revealed in the post exploration discussion.

8.4.5 Revealing Use Cases

After the participants learned the functionality of the tool in the exploration a post exploration discussion set out to reveal use cases or tasks of daily work where the visualization tool could be used.

As I said, the goal of the evaluation was also to find out possible scenarios, in which the tool could be used and to identify tasks that can be supported by the tool's provided functionality. The interviewees were asked to state typical tasks for which they would use the tool and/or if they can conceive a scenario in their daily work where they would use the tool. Interviewee No. 1 responded:

> "The question is for me to find out, who actually has a strong relation to a specific document, that is important to me."

To the same question, interviewee No. 2 answered:

> "One could for instance easily control, if project partners work on a document or show the effort in the domain that they should show. Now, as a project manager, this is very convenient. One cannot arrive at fail-safe conclusions, but with the visualization, one can certainly keep the overview and that is certainly practical, I mean, you usually don't have that overview...The controlling function was very interesting, to see who is involved, that was partly surprising, who doesn't keep an eye on what happens, who actually should keep an eye on it. Insofar, I would take a look at it [the visualization] on and off, especially before meetings, to be able to judge a statement of a certain person on a better basis. The question is if you can really see it, but it gives you a hint, that you can cross-check, but it gives you an idea of who doesn't have a clue. Some persons have been surprisingly inactive, whereby this could also be due to the eMails...[see previous section]. But if you really use BSCW the way you should, than you can control it in between, also among oneselves, [*jokingly:*] that is *public ashaming*."

And interviewee No. 3 said:

> "As soon as I had some management function, that I really had a lot to do with texts, then this would be interesting...for the heavy user this would be interesting. If I was a manager [of a workpackage in a project] it would be interesting to see, for example an important deliverable, when had work on it been started, how long did it take...at what point of time did which partners get involved with it...then there's a control functionality, to look at sometime, did everyone really read it, this is sort of a surveillance function...Or one can check if all parties, that should contribute, have actually contributed yet. I think, if you're responsible for a certain workpackage, and want to see how far is the

> progress, which people have not yet worked on it, or which persons to send it to, or trigger it, I think it's interesting for that purpose."

Note that the interviewees responded to the questions independent of each other, the interviews were conducted one on one in the interviewee's respective office. The similarity of the conceived tasks in these quotes, suggests a clear interpretation that the tool serves a purpose of monitoring a workspace, its evolution of the contributions of the people in the workspace. This is a task of the workspace management, that is often inherent in the tasks of managing a workpackage in research projects. Interviewee No. 1 actually is a workpackage leader and he has instantly recognized this functionality when interacting with the tool, by using the chronological slider to check contributions to a deliverable before and after a deadline (see previous section). Interviewee No. 2 mentioned, that he would use the tool "especially before meetings", a similar task as it had been described in the use case in **??**. The possible dual purpose of the tool that has been assumed earlier in this thesis, as conceiving it either as a tool to support individual tasks and/or to see it as a tool of control and monitoring progress and contribution is supported by the use cases the interviewees state. However, the perceived functionality of analysis and control seem to outweigh the individual functionality, that is only mirrored by the meeting scenario. After the interviewees stated use cases of the tool, I asked them explicitly, if they rather see the tool as providing an analytic overview for controlling purposes or as a tool that supports individual tasks. All interviewees stated that it is rather a function that provides analysis and control. Interviewee No. 1 states:

> "Rather controlling, I would say. Since I initiated this process of the deliverable and manage it, that is interesting to see, who interacts how, of the persons that I expected to do so, that is interesting for controlling, simply."

Although it is difficult to draw a boundary where individual tasks end and monitoring and analysis tasks begin: the workpackage manager's individual task is to coordinate - and therefore monitor and control - the workspace's evolution. The "regular" project participant needs to know who works with whom on what to classify the partner's statement at the next meeting. It is safe to say that the visualization tool supports task that afford a reflective overview of who contributed to which artefacts of work and the resulting relationships that arise out of the workspace evolution.

Another aspect the evaluation aimed at, was to find out in which way the communicated visual information was interpreted by the user and whether the information provided by the tool is redundant in the user's perception, e.g. if he thinks that existing BSCW functionality already feed the information needs. The participants were asked what kind of information the tool communicates in their opinion and if they are aware of related or similar functionality in BSCW. Interviewee No. 1 responded:

> "That gives an overview, who has a relationship to me on the basis of this document. One could abstract this to all documents I have ever uploaded and generate something like a personal network from that."

When asked about related functionality, all interviewees mentioned the HyperGraph visualization, that they had seen before, but never used. It is interesting to see that even though I inquired about related function, they initially all stated a functionality that is similar in form, but less similar in function. Two of the three interviewees could not think of any other similar functionality in BSCW, even though basically, all awareness functionality is similar to the visualization tool. At least, it can be concluded that the information provided by this tool is not redundant. Only one interviewee came up with more ideas of related functionality, Interviewee No. 2 said:

> "...the activities on the right side [refers to BSCW *Event Icons*]. The events are being tracked, but you the frequencies are not displayed, nor the relationships between the people. There are single components in BSCW that show this information somehow, but not at a glance...you don't see the interconnection at one glance, like in this tool...One can get notified by eMail [refers to BSCW's *Daily Report*], so you can reach a certain awareness - I think that's what this is about - one could also count the eMails on this document, then you'd have the same. But it would not be so clearly arranged, one is relieved of this work. And of course, one sees the awareness of people, there's this toolbar [refers to *SmartMaps*] where you see who is currently active, who is more or less active, indicated by colors...so this is something similar, but you don't see the interconnections between documents and persons, that [SmartMaps] is only related to the workspace."

This interviewee correctly identifies this tool as providing awareness information to the user. Also, he notices similar functions, names three awareness functionalities, but does not fail to appreciate the difference between the evaluated visualization tool and the other awareness functions. Especially, he names the function of achieving an overview at a glance, and seeing the interconnectedness of people and documents in the workspace.

8.5 Summary

The evaluation tested the developed visualization tool in an ethnographic approach to qualitative interviews and aimed at the assessment of the general usability of the tool, user-conceived use cases and scenarios and emergent purpose of the tool.

Regarding usability, the tool's functionality mostly adheres to the user's expectations, with some minor changes to be made especially to naming and grouping of functionality and change of tool tips. In most cases, the provided

help in tool tips and the help file is sufficient, with the exception of the second button in the *Highlight* button group, which requires additional conjunctive interaction (see above). The purpose of the functionality however, was conceived to be useful and the functionality was considered required. The observation of the participant's interaction with the tool suggest that the tool's functionality is controllable.

We have seen that the tool provides not only workspace awareness information but also serves to infer social awarenees, as the interrelatedness of doucments and people is visualized. In the interviews, the participants often referred to the visualization's spatial properties, emphasizing closeness or distance and thus interpreting levels of cooperation. From the congruency of the visualization with the subjective network the interviewee's had "in mind", it can be concluded that the spatial properties of the layout are a good indicator for the quality of cooperation, and therefore an indicator for relevance of the co-workers activity. The tool suggests to the user a quality of her cooperative network, that she may or may not agree with, but it will make her think about it. After having studied the network perspectives the tool affords, the result may be an altered perception of the relevance of forthcoming awareness information. By communicating a qualitative overview of her cooperative network, the task of grading information the groupware provides as more or less relevant is facilitated. For instance, a user that reads the ´´Daily Report" after having overviewed her network visualization may rate the activities of certain users as more important as that of others. This may lead to a sharpened conception of priorities in her daily work.

As tasks that the tool could be used for emerged naturally from the interviews, this suggess that the tool's concept and its realization can be used for sense-making and insight. The evolution of a workspace can be controlled in a chronological fashion to support tasks of management such as monitoring work progress and trigger project partner input, or such individual practical purposes as to assess the workspaces state of contribution to prepare for a meeting. The evaluation proved insightful with regards to the research questions formulated and with respect to the success criteria, it can be concluded that the tool's meaning-making capabilities and afforded reflecting perspectives on cooperative work are a success in that it amends BSCW with an original awareness information providing tool.

We furthermore learned that a personal network cannot be concluded solely from activities in the workspace, the daily workpractice suggests cooperation over different communication channels that are not represented in the employed groupware. The implications of this are of relevance to CSCW and groupware design in general, therefore, they will be discussed in the conclusions of this thesis.

9

Conclusions

In this thesis, the development process of an awareness information visualization tool was described on the foundations of research insights from Socio-Technical Systems, CSCW and Social Network Analysis. Insights on the role of time in communication and cooperation led to the conception of a visualization tool that not only makes contextual activity identifyable post hoc, it allows for the visualization of the dynamic, chronologic evolution of a shared workspace.

The evaluation showed that by providing a network perspective on the interrelated nature of actors and their activities in a shared workspace, awareness information was augmented to include a *quality* of cooperation that indicates the *relevance* of a co-worker's activity for the individual that uses the tool. Whereas traditional awareness information includes *who?, what?, when?* information, the tool's network perspective augments that information to include *how?* information such as: how close is somebody to me in the network? How do the other's activities relate to mine? How did the workspace evolve over time?

The evaluation of the tool proved to be insightful with respect to the research questions and the tool can be regarded as a successful augmentation to the functionality provided by BSCW. The visualization of awareness information proved to be especially sucessful at leading to a conception of the state of the shared workspace at a glance. The chosen layout showed to mirror the interviewee's subjective perception of the cooperation in the workspace. The dynamic interaction that the tool affords allows meaning-making from different perspectives such as from a project workpackage manager's point of view or from a co-worker's perspective, that wants to prepare himself for a meeting. The functionality filtering according to event type and chronologic progress, zooming and emphasis of the connectedness of a node were seen as especially helpful. Despite the existence of several awareness information providing functions in BSCW, the information the tool provides did not lead to a redundant conception of awareness in the interviewee's opinion. The visualization tool shows its strength in depicting the actor's activity on joint artefacts

of work and the resulting interpersonal relationships. Whereas most awareness functions focus on the events on the artefacts in the workspace hierarchy, this tool emphasizes a networked perspectives of *people* and their interrelated events on shared artefacts; the visualization across several workspaces is also possible.

As the networked visualization of workspace activity to enhance workspace awareness proved sucessful, the evaluation of the inferrence mechanism that visualizes interpersonal networks of mediated working relationships gave insight that peer networking in cooperative work involves diverse communication channels with different levels of media richness [Daft and Lengel, 1986] such as eMail, face-to-face communication or video conferencing. An accurate mapping of a user's subjective interpersonal network cannot be inferred from the interpretation of activities within the boudaries of only one medium, in this case a groupware. This finding indicates that by providing workspace awareness, a holistic awareness encompassing the individual's interpersonal network cannot be achieved. The challenge of evaluating awareness of activity across different levels of interaction has also been described in Neale et al. [2004].

For the future, if the goal is to provide more holistic, robustly funded awareness information, the integration of information from differrent media channels has to be considered. Whereas non-integrated applications have their right and purpose to create a rich landscape of tools for all conceivable tasks, awareness support for cooperative work would benefit from more integrated means to collect interaction information.

This work contributes to the discussion of awareness support for groupware by introducing SNA-techniques to visualize interrelated workspace activity and thereby augments workspace awareness with insights on the quality of cooperation. The quality of cooperation assists the assessment of relvance of other actors' activities and thereby contributes to a facilitated orientation in the idividual's cooperative network.

10

Outlook

Groupware functionality that aims at fostering workspace awareness hitherto bases its information on the activity a user does in a shared workspace. However, awareness of a network of actors and artifacts in which a user does his work comprises much more than events on shared artefacts. The semantic relationships of the documents is an important signpost through the document repository. However, the system has no knowledge about the semantic connection of the artefacts of work. The browsing and reading of documents to find related work is often cumbersome, at least it has to rely on the other users naming and describing their documents meaningful. Because often related work passes by unnoticed in the everyday context of distributed projects, communication through the artefact is hindered. As I have mentioned, the concept of communication through the artefact is described as essential in the process of awareness [Dix et al., 2003]. As a result, potentially useful connections between peers remain concealed. It is important to have a notion of the domain of interest and expertise of the co-workers, this meta information can be deducted with the SNA-techniques outlined in this thesis, by using *create*, *read* and *modify events* of documents as indicator of expertise and interest.

Another approach would require the users to explicitly provide this information, for instance by techniques such as *tagging*. However, this is costly as it means an overhead of which the benefit for the individual might not be identifyable. Grudin [1988] names the demand for unbeneficial activities as one factor, why groupware fails.

This meta information could be enriched by deducting semantic meta information from the content of the documents the user provides for the groupware. Implicit techniques such as *text mining* seem promising to map persons to their areas of expertise and interest. Furthermore, content may be structured in ontologies that represent domains of expertise and consider semantic similarities between the documents. Such content-aware techniques in conjunction with the presented SNA-techniques of deducting interpersonal networks from activities in shared workspaces present a promising approach for enhancing groupware with recommender services. By comparing the activities

a user does in a workspace, his interests, expertise and his interpersonal networks can be deducted. This information can be compared to a semantically structured document repository.

This cumulated information can then be used to recommend documents of the same domain or with similar content as identified by the ontologic relationships or text mining techniques based on the workspace activity of the user, which indicates areas of interest and expertise by the read and created documents. In the same way, co-workers can be recommended to be addressed as carriers of expert knowledge in a domain that interest is deducted in by the activities conducted in the shared workspace. Figure 10.1 shows an application of this concept. Two groups work in the same workspace on two similar documents, but they are not aware of that as their small networks are not connected to each other, as deducted by the SNA-techniques. For in this concept the system knows about the semantic connection between the documents, it can recommend to the groups to talk to each other about the domain and benefit from each other's experiences.

To avoid false or redundant recommendations because the systems bases its recommendations only on the activity within the system's boundaries, communication activity in diverse technical media channels can be respected by providing components that log interaction across different technical communication applications such as eMail and instant messagers.

As cooperative work is situated in modern society, its members are not excepted from the danger of being flooded with information. The identification of relevant information and knowledge carriers is crucial especially in knowledge-intense domains. SNA-techniques that interpret workspace activity in conjunction with a refined semantic content repository that describe the artefacts of work's content may provide the individual worker with valuable information by recommending documents and people of interest and desired expertise. This approach is worth to be investigated in the future, as it seems promising to provide the individual with what is relevant for his work.

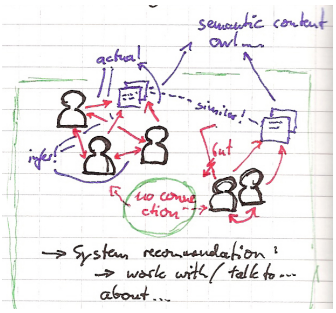

Fig. 10.1. Future work: SNA-techniques enhanced with a semantic document repository.

A

Appendix A

:Joel.Fischer/Masterarbeit Social Networks in CSCW/RequirementsAnalysis_eProfessionals

Show Votes

Umfrage zu BSCW und sozialen Netzwerken

Guten Tag! Diese Umfrage wird im Rahmen einer Masterarbeit an der Universitaet Duisburg-Essen durchgefuehrt. Grundlage fuer diese Umfrage ist die Annahme, dass mittels des BSCW Personen kooperieren, die in ein soziales Netzwerk eingebettet sind. Jede Person hat in der Regel mehrere soziale Netzwerke. Die Fragen dieser Umfrage beziehen sich auf Ihr soziales Netzwerk, in dessen Kontext Sie Ihre Arbeit machen und den BSCW zur Unterstuetzung verwenden. Der Einfachheit halber wird deswegen der Begriff "Arbeitsnetzwerk" verwendet. Vielen Dank, dass Sie sich die Zeit nehmen, ein paar kurze Fragen zu ihrer Person, zu BSCW und sozialen Netzwerken zu beantworten. Ihre Antworten bleiben anonym.

Dieser Button wurde aus technischen Gründen erstellt.	6	100%
Total	6	

1. Inwieweit stimmen Sie den folgenden Aussagen zu, die sich mit der derzeitigen Sicht auf den BSCW befassen?

	stimme voll zu 1	2	neutral 3	4	lehne stark ab 5	Total
1. Die Sicht auf den BSCW stellt eher die Artefakte (z.B. Ordner und Dokumente) in den Vordergrund und nicht die Personen, die mit den Artefakten arbeiten.	56% 17	33% 10	6% 2	3% 1	0	30
2. Die gegenwaertige Sicht auf den BSCW ermoeglicht mir, die Art und den Grad der Arbeitsbeziehung zu Personen in meinem Arbeitsnetzwerk zu erkennen, z.B. mit wem ich eng zusammen arbeite.	6% 2	20% 6	13% 4	50% 15	10% 3	30
3. Die Metapher der "hierarchischen Ordnerstruktur" macht es mühsam, die Uebersicht zu behalten, wessen Dokumente ich lese und bearbeite und wer meine Dokumente liest und bearbeitet.	16% 5	30% 9	33% 10	13% 4	6% 2	30
4. Die gegenwaertige Sicht auf den BSCW gibt mir das Gefuehl, Teil eines Arbeitsnetzwerks zu sein.	13% 4	30% 9	33% 10	20% 6	3% 1	30

2. Inwieweit stimmen Sie den folgenden Aussagen zu?

	stimme voll zu 1	2	neutral 3	4	lehne stark ab 5	Total
1. Wenn ich die Dokumente von einer anderen Person im BSCW lese, dann besteht eine direkte Beziehung zu dieser Person in meinem Arbeitsnetzwerk.	16% 5	30% 9	33% 10	20% 6	0	30
2. Je mehr Dokumente ich von einer Person lese, je näher steht sie mir in meinem Arbeitsnetzwerk.	20% 6	43% 13	16% 5	16% 5	3% 1	30
3. Wenn ich die Dokumente von einer anderen Person im BSCW bearbeite, dann besteht eine direkte Beziehung zu dieser Person in meinem Arbeitsnetzwerk.	40% 12	33% 10	13% 4	13% 4	0	30
4. Je mehr Dokumente ich von einer Person bearbeite, je näher steht sie mir in meinem Arbeitsnetzwerk.	40% 12	26% 8	23% 7	6% 2	3% 1	30
5. Ich arbeite enger mit Personen zusammen, dessen Dokumente ich bearbeite als mit Personen, dessen Dokumente ich lese.	33% 10	26% 8	26% 8	3% 1	10% 3	30

3. Stellen Sie sich ein grafisches Netzwerk mit Knoten und Kanten als alternative Sicht auf den BSCW vor. Die Knoten wuerden fuer Personen und Artefakte stehen und die Kanten fuer die Beziehungen zwischen den PERSONEN und den ARTEFAKTEN (z.B. erzeugen, lesen und bearbeiten). Inwieweit stimmen Sie den folgenden Aussagen zu, die sich mit einer solchen Sicht auf den BSCW befassen?

	stimme voll zu 1	2	neutral 3	lehne stark ab 4	5	Total
1. Eine solche Sicht auf den BSCW wuerde meinen Ueberblick ueber mein Arbeitsnetzwerk verbessern, da ich auf einen Blick sehe, wer wessen Dokumente liest und bearbeitet.	6%	63%	16%	13%		
	2	19	5	4	0	30
2. Eine solche Sicht auf den BSCW hat das Potential, die Wahrnehmung über die Art und den Grad der Arbeitsbeziehung zu Personen in meinem Arbeitsnetzwerk zu verbessern.	10%	50%	23%	16%		
	3	15	7	5	0	30

4. Stellen Sie sich ein grafisches Netzwerk mit Knoten und Kanten als alternative Sicht auf den BSCW vor. Dieses mal wuerden die Knoten fuer Personen stehen und die Kanten fuer die Beziehungen zwischen den PERSONEN. Inwieweit stimmen Sie den folgenden Aussagen zu, die sich mit einer solchen Sicht auf den BSCW befassen?

	stimme voll zu 1	2	mittel 3	4	lehne stark ab 5	Total
1. Eine solche Sicht wuerde den Knoten, der meine Person verdeutlicht in den Mittelpunkt stellen und die Personen, mit denen ich im BSCW zusammen arbeiten, um mich herum gruppieren, je nachdem wie eng wir zusammen arbeiten.	20%	60%	13%	3%	3%	
	6	18	4	1	1	30
2. Eine solche (in 4.1 dargestellte) individualzentrische Sichtweise hat das Potential, die Wahrnehmung der Art und des Grades meiner Beziehungen zu Personen in meinem Arbeitsnetzwerk zu verbessern.	23%	40%	30%	3%	3%	
	7	12	9	1	1	30
3. Eine solche Sicht wuerde den gesamten (ausgewählten) Arbeitsbereich abbilden und nicht mich, den gegenwartigen Nutzer, sondern die Person zentrieren, die z.B. die meisten Beziehungen hat.	6%	26%	36%	20%	10%	
	2	8	11	6	3	30
4. Eine solche (in 4.3 dargestellte) gruppenzentrische Sichtweise hat das Potential, die Wahrnehmung der Art und des Grades meiner Beziehungen zu Personen in meinem Arbeitsnetzwerk zu verbessern.	16%	46%	30%	3%	3%	
	5	14	9	1	1	30
5. Eine solche (in 4.3 dargestellte) gruppenzentrische Sichtweise birgt die Möglichkeit, dass Dritte meine Position, Art und Grad von Beziehungen im Netzwerk kontrollieren. Daher lehne ich eine solche Sichtweise ab.	10%	10%	23%	40%	16%	
	3	3	7	12	5	30

5. Bitte schaetzen Sie Ihr Niveau als NutzerIn des BSCW ein!

Anfänger	6	20%
fortgeschritten	13	43%
Experte	11	36%
Total	30	

6. Ihr Geschlecht:

weiblich	7	23%
maennlich	23	76%
Total	30	

7. Wie alt sind Sie?

20-25 Jahre alt	1	3%
26-30 Jahre alt	13	46%
31-35 Jahre alt	4	14%
36-45 Jahre alt	5	17%
46-55 Jahre alt	2	7%
56-65 Jahre alt	3	10%
66 + Jahre alt	0	
Total	28	

8. Wie verständlich waren die Fragen in dieser Umfrage?

sehr leicht verständlich	2	6%
leicht verständlich	13	43%
mittelmäßig verständlich	8	26%
schwer verständlich	7	23%
unverständlich	0	
Total	30	

Vielen Dank für die Beantwortung! Haben Sie abschliessend noch einen Kommentar?

Please vote...

:Joel.Fischer/Masterarbeit Social Networks in CSCW/RequirementsAnalysis_Students

Show Votes

Umfrage zu BSCW und sozialen Netzwerken

Guten Tag! Diese Umfrage wird im Rahmen einer Masterarbeit an der Universitaet Duisburg-Essen durchgefuehrt. Grundlage fuer diese Umfrage ist die Annahme, dass mittels des BSCW Personen kooperieren, die in ein soziales Netzwerk eingebettet sind. Jede Person hat in der Regel mehrere soziale Netzwerke. Die Fragen dieser Umfrage beziehen sich auf Ihr soziales Netzwerk, in dessen Kontext Sie Ihre Arbeit machen und den BSCW zur Unterstuetzung verwenden. Der Einfachheit halber wird deswegen der Begriff "Arbeitsnetzwerk" verwendet. Vielen Dank, dass Sie sich die Zeit nehmen, ein paar kurze Fragen zu ihrer Person, zu BSCW und sozialen Netzwerken zu beantworten. Ihre Antworten bleiben anonym.

Dieser Button wurde aus technischen Gründen erstellt.	9	100%
Total	9	

1. Inwieweit stimmen Sie den folgenden Aussagen zu, die sich mit der derzeitigen Sicht auf den BSCW befassen?

	stimme voll zu 1	2	neutral 3	4	lehne stark ab 5	Total
1. Die Sicht auf den BSCW stellt eher die Artefakte (z.B. Ordner und Dokumente) in den Vordergrund und nicht die Personen, die mit den Artefakten arbeiten.	51% 14	40% 11	3% 1	3% 1	0	27
2. Die gegenwaertige Sicht auf den BSCW ermoeglicht mir, die Art und den Grad der Arbeitsbeziehung zu Personen in meinem Arbeitsnetzwerk zu erkennen, z.B. mit wem ich eng zusammen arbeite.	0	18% 5	18% 5	51% 14	11% 3	27
3. Die Metapher der "hierarchischen Ordnerstruktur" macht es mühsam, die Uebersicht zu behalten, wessen Dokumente ich lese und bearbeite und wer meine Dokumente liest und bearbeitet.	14% 4	44% 12	22% 6	14% 4	3% 1	27
4. Die gegenwaertige Sicht auf den BSCW gibt mir das Gefuehl, Teil eines Arbeitsnetzwerks zu sein.	0	48% 13	25% 7	22% 6	3% 1	27

2. Inwieweit stimmen Sie den folgenden Aussagen zu?

	stimme voll zu 1	2	neutral 3	4	lehne stark ab 5	Total
1. Wenn ich die Dokumente von einer anderen Person im BSCW lese, dann besteht eine direkte Beziehung zu dieser Person in meinem Arbeitsnetzwerk.	7% 2	25% 7	33% 9	29% 8	3% 1	27
2. Je mehr Dokumente ich von einer Person lese, je näher steht sie mir in meinem Arbeitsnetzwerk.	3% 1	37% 10	25% 7	25% 7	7% 2	27
3. Wenn ich die Dokumente von einer anderen Person im BSCW bearbeite, dann besteht eine direkte Beziehung zu dieser Person in meinem Arbeitsnetzwerk.	14% 4	44% 12	18% 5	18% 5	3% 1	27
4. Je mehr Dokumente ich von einer Person bearbeite, je näher steht sie mir in meinem Arbeitsnetzwerk.	14% 4	33% 9	29% 8	14% 4	7% 2	27
5. Ich arbeite enger mit Personen zusammen, dessen Dokumente ich bearbeite als mit Personen, dessen Dokumente ich lese.	11% 3	44% 12	18% 5	14% 4	11% 3	27

3. Stellen Sie sich ein grafisches Netzwerk mit Knoten und Kanten als alternative Sicht auf den BSCW vor. Die Knoten wuerden fuer Personen und Artefakte stehen und die Kanten fuer die Beziehungen zwischen den PERSONEN und den ARTEFAKTEN (z.B. erzeugen, lesen und bearbeiten). Inwieweit stimmen Sie den folgenden Aussagen zu, die sich mit einer solchen Sicht auf den BSCW befassen?

	stimme voll zu 1	2	neutral 3	4	lehne stark ab 5	Total
1. Eine solche Sicht auf den BSCW wuerde meinen Ueberblick ueber mein Arbeitsnetzwerk verbessern, da ich auf einen Blick sehe, wer wessen Dokumente liest und bearbeitet.	29% 8	22% 6	22% 6	18% 5	7% 2	27
2. Eine solche Sicht auf den BSCW hat das Potential, die Wahrnehmung über die Art und den Grad der Arbeitsbeziehung zu Personen in meinem Arbeitsnetzwerk zu verbessern.	25% 7	51% 14	18% 5	0	3% 1	27

4. Stellen Sie sich ein grafisches Netzwerk mit Knoten und Kanten als alternative Sicht auf den BSCW vor. Dieses mal wuerden die Knoten fuer Personen stehen und die Kanten fuer die Beziehungen zwischen den PERSONEN. Inwieweit stimmen Sie den folgenden Aussagen zu, die sich mit einer solchen Sicht auf den BSCW befassen?

	stimme voll zu 1	2	mittel 3	4	lehne stark ab 5	Total
1. Eine solche Sicht wuerde den Knoten, der meine Person verdeutlicht in den Mittelpunkt stellen und die Personen, mit denen ich im BSCW zusammen arbeiten, um mich herum gruppieren, je nachdem wie eng wir zusammen arbeiten.	18% 5	37% 10	29% 8	14% 4	0	27
2. Eine solche (in 4.1 dargestellte) individualzentrische Sichtweise hat das Potential, die Wahrnehmung der Art und des Grades meiner Beziehungen zu Personen in meinem Arbeitsnetzwerk zu verbessern.	7% 2	59% 16	25% 7	3% 1	3% 1	27
3. Eine solche Sicht wuerde den gesamten (ausgewählten) Arbeitsbereich abbilden und nicht mich, den gegenwartigen Nutzer, sondern die Person zentrieren, die z.B. die meisten Beziehungen hat.	14% 4	22% 6	40% 11	14% 4	7% 2	27
4. Eine solche (in 4.3 dargestellte) gruppenzentrische Sichtweise hat das Potential, die Wahrnehmung der Art und des Grades meiner Beziehungen zu Personen in meinem Arbeitsnetzwerk zu verbessern.	7% 2	25% 7	51% 14	7% 2	7% 2	27
5. Eine solche (in 4.3 dargestellte) gruppenzentrische Sichtweise birgt die Möglichkeit, dass Dritte meine Position, Art und Grad von Beziehungen im Netzwerk kontrollieren. Daher lehne ich eine solche Sichtweise ab.	7% 2	25% 7	25% 7	25% 7	14% 4	27

5. Bitte schaetzen Sie Ihr Niveau als NutzerIn des BSCW ein!

Anfänger	6	22%
fortgeschritten	19	70%
Experte	2	7%
Total	27	

6. Ihr Geschlecht:

weiblich	5	18%
maennlich	22	81%
Total	27	

7. Wie alt sind Sie?

20-25 Jahre alt	13	48%
26-30 Jahre alt	13	48%
31-35 Jahre alt	0	
36-45 Jahre alt	1	3%
46-55 Jahre alt	0	
56-65 Jahre alt	0	
66 + Jahre alt	0	
Total	27	

8. Wie verständlich waren die Fragen in dieser Umfrage?

sehr leicht verständlich	2	7%
leicht verständlich	4	14%
mittelmäßig verständlich	17	62%
schwer verständlich	4	14%
unverständlich	0	
Total	27	

Vielen Dank für die Beantwortung! Haben Sie abschliessend noch einen Kommentar?

Please vote...

B

Appendix B

Interviewleitfaden für eProfessional: _____
Anmerkung: *kursiver Text: Evaluationsfragestellung oder Hinweis for Evaluator!*

Voraussetzung: eProfessional ist aktives Mitglied im Projekt ECOSPACE oder CoSpaces

1. Auffinden des VisTools:
Das Tool ist über den Testzugang des BSCW http://cwe-projects.eu/bscw zugreifbar. Bitte navigiere in einen WS / Unterordner des Projects ECOSPACE oder CoSpaces, in dem du regelmäßig arbeitest. Rufe bitte die Visualisierung auf. Frage: bekannt, wie das Tool aufzurufen ist?
Bekannt? Ja _ Nein _
Wenn nein: Ist dir die Funktion "Readers" / "Leser" bekannt?
Bekannt? Ja _ Nein _
Wenn nicht, Zugang über "Readers" / "Leser" zeigen.
Interessieren Dich alle Events des WS (a) oder nur Events auf Dokumenten, die Du eingestellt / bearbeitet / gelesen hast (b)?
(a) _ (b) _

Bitte rufe die Visualisierung für (a) oder (b) auf.
Wenn nicht bekannt wie, hier Hilfestellung leisten.
Hilfestellung erfolgt: ja _ nein _

Ausgangspunkt: Visualisierung BSCW Peer Network

2. Themenkomplex Usability
Selbsterklärbarkeit:
Was siehst du?
Evtl. Frage verfeinern: Was repräsentieren die visuellen Elemente der Visual-

isierung?
Wofür stehen die Farben der Elemente Knoten und Kanten?

Layout
Layout generell / Visuelle Orientierung
Was sagt dir die relative Anordnung der Elemente zueinander - das Layout der Visualisierung?
Was denkst du nach dem Bild entscheidet die relative Positionierung der Knoten zueinander?

Layout user-spezifisch
Finde den Knoten, der dich repräsentiert. Welche Position im Netzwerk nimmst du ein? Entspricht das Bild der Vis. deinem persönlichen Eindruck deiner Position?

Interaktion
Fragestellung Evaluation: Erwartungskonformität Funktionalität
Schauen wir uns nun die Interaktionsmöglichkeiten mit der Visualisierung an. Bevor du sie ausprobierst, was erwartest du, passiert, wenn du die Interaktionskomponenten bedienst?
Durchgehen der Funktionen nach Gruppierungen der Elemente in:
a) Filter
b) Highlight
c) Examiner Lens
d) Zoom
e) MouseMode
f) Personal Network
g) Legend Info
h) Date Slider
i) Lower boundary setter

Probiere nun die Funktionen der Reihe nach aus. Bitte erzähle mir, was du tust, während du die Funktionalitäten ausprobierst.

Während des Ausprobierens:
Entspricht die Reaktion deinen Erwartungen? Wenn nein, was hättest du erwartet? Sind die angebotenen Hilfen (Tool Tips und Help) hilfreich? Funktion sinnvoll, evtl.: Was sagt dir das?

3. Themenkomplex: Funktionsspezifische Fragen
Personal Network
Was siehst du jetzt?
Welche Elemente siehst du?

Was ist der Unterschied zum vorher gesehenen Peer NW?
Wofür stehen die Kanten zwischen den Usern?

Was denkst du, welche Voraussetzungen müssen erfüllt sein, damit die Kanten zwischen den Usern entsprechende Gestalt annehmen?
Was hältst Du von der Vermutung, gemeinsame Read Events auf einem Dokument sind eine "weak" Arbeitsbeziehung und gemeinsame modify Events, bzw. ein modify und ein read-Event eine "strong" Arbeitsbeziehung?
Was sagt dir das Bild?
Wenn du es mit dem Bild "in deinem Kopf" von deinem Kooperationsnetwerk vergleichst, kommt es diesem Bild nahe? Wenn nein, warum nicht?
Was ist gelungen? Was ist verbesserungswürdig?

Common Interest
Was zeigt dir die Tabelle deiner Meinung nach?
Filtere die Tabelle doch mal nach der Spalte Common Interest. Es entsteht eine Rangordnung. Entspricht diese Rangordnung deinen Erwartungen / spiegelt sie deine persönliche Rangordnung von Usern gleichen Interesses wider?
Ist die Relation aller Dokumente eines Users zu den Dokumenten, die der User gemeinsam mit einem anderen hat, ein angemessener Indikator für gemeinsames Interesse? Wenn nein, wofür dann?

Legend & Info
Ist man auf diese Hilfe angewiesen oder ist das Erlernen des Tools durch intuitive Exploration möglich?

Chronological slider
Was kannst du mit dem Slider machen?
Was ist die Funktion des Feldes: show only events after?
Ist die Betrachtung von Events in beliebigen Zeiträumen interessant?
Für welche Aufgaben deiner täglichen Arbeit ist das von Interesse?

4. Themenkomplex: Fragen nach der Exploration
Zweck: Use Cases entdecken:

User-percived functionality
Welche Funktion hat das Tool für dich?

What is conveyed? What effects on action does the tool have?
Welche Informationen vermittelt dir das Tool?
Führen die Informationen dazu, dass du eine Handlung Aktion ausführst? Wenn ja, welche z.B.?

User-perceived purpose
Welchen Zweck erfüllt das Tool deiner Meinung nach?

Related functionality identified?
Kennst du vergleichbare / alternative Funktionalitäten im BSCW? Welche sind das?

Use cases
Wann würdest du das Tool benutzen?
Könntest du dir vorstellen, das Tool in deinem täglichen Arbeitskontext zu verwenden? Wenn nein, warum nicht, wenn ja, beschreibe mal ein Szenario, in dem Du es benutzen würdest!

Analytic tool vs. Individual support of tasks
Erfüllt das Tool eher analytische Zwecke einer Draufsicht auf Arbeitsbeziehungen oder eignet es sich für die Unterstützung deiner individuellen Aufgaben?
Deiner Meinung nach, was sind Stärken und Schwächen des Tools?
Hast du abschließen einen Kommentar oder Verbesserungsvorschläge an das Tool?

References

J.H.E. Andriessen. The why, how and what to evaluate of interaction technology: A review and proposed integration. In P.J. Thomas, editor, *CSCW Requirements and Evaluation*, chapter 1, pages 107–124. Springer Verlag, London, 1996.

ApacheJakartaProject. Commons collections, 2007. URL `http://commons.apache.org/collections/`.

Wolfgang Appelt. What groupware functionality do users really use? analysis of the usage of the bscw system. *pdp*, 00:337, 2001.

Wolfgang Appelt. Www based collaboration with the bscw system. In *Conference on Current Trends in Theory and Practice of Informatics*, pages 66–78, 1999.

Wolfgang Appelt and Peter Mambrey. Experiences with the bscw shared workspace system as the backbone of a virtual learning environment for students. In *Proc. of the World Conference on Educational Mutlimedia, Hypermedia and Telecommunications ED-Media 99*, Seattle, 1999.

C. Argyris and D. Schön. *Die lernende Organisation. Grundlagen, Methoden, Praxis*. Klett Cotta, Stuttgart, 1999.

Albert-László Barabási. *Linked - How Eerything is Connected to Everythin Else and What It Means for Business, Science and Everyday Life*. Penguin Group, New York, 2003.

V. Batagelj and A. Mrvar. Pajek: Program for large network analysis, 2007. URL `http://vlado.fmf.uni-lj.si/pub/networks/pajek/`.

S. Benford, J. Bowers, L. Fahlen, C. Greenhalgh, and D. Snowdon. User embodiment in collaborative virtual environments. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI '95)*, pages 242–249, 1995.

Steve Benford and Lennart Fahlén. A spatial model of interaction in large virtual environments. In *ECSCW'93: Proceedings of the third conference on European Conference on Computer-Supported Cooperative Work*, pages 109–124, Norwell, MA, USA, 1993. Kluwer Academic Publishers.

R. Bentley, T. Horstmann, K. Sikkel, and J. Trevor. Supporting collaborative information sharing with the WWW: The BSCW shared workspace system. In *Proc. of the 4th Int. World Wide Web Conf.*, pages 63–74. O'Reilly & Associates, 1995.

D. Benyon. Beyond navigation as metaphor. In *ECDL '98: Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries*, pages 705–716, London, UK, 1998. Springer-Verlag. ISBN 3-540-65101-2.

Uwe M. Borghoff and Johann H. Schlichter. *Computer-Supported Cooperative Work. Introduction to Distributed Applications.* Springer, Berlin, 2000.

U. Brandes, M. Eiglsperger, M. Kaufmann, and J. Lerner. The graphml file format. Technical report, 2007. URL `http:/graphml.graphdrawing.org/`.

Tim Bray, Jean Paoli, C.M. Sperberg-McQueen, Eve Maler, and François Yergeau. Extensible markup language (xml) 1.0 (fourth edition). Recommendation, World Wide Web Consortium, 2006. URL `http://www.w3.org/TR/2006/REC-xml-20060816/`.

Steffen Budweg, Joel E. Fischer, Peter Mambrey, and Uta Pankoke-Babatz. Kommunikative vernetzung in der universitären lehre. In A.M. Heinecke and H. Paul, editors, *Mensch & Computer 2006. Mensch und Computer im StrukturWandel*, pages 303–312. Oldenbourg Wissenschaftsverlag GmbH, München, 2006.

Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman. Information visualization. In Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman, editors, *Readings in Information Visualization. Using Vision to Think*, pages 1–34. Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1999.

Kathleen M. Carley. Dynamic network analysis. In Ron Breiger and Kathleen M. Carley, editors, *Summary of the NRC workshop on Social Network Modeling and Analysis.* National Research Council, 2003.

John M. Carroll. *Making Use: Scenario-Based Design of Human-Computer Interactions.* The MIT Press, 2000. ISBN 0262032791.

CERN. The colt project, 2007. URL `http://dsd.lbl.gov/~hoschek/colt/`.

H.H. Clark, R. Schreuder, and S. Buttrick. Common ground and the understanding of demonstrative reference. *Journal of Verbal Learning and Verbal Behavior*, 22:245–258, 1983.

James M. Clark and Allan Paivio. Dual coding theory in education. *Educational Psychology Review*, 3(3):149–210, September 1991.

Cohen and Bacdayan. Organizational routines are stored as procedural memory: Evidence from a laboratory study. *Organizational Science*, 5(4):554–568, 1994.

Rob Cross and Andrew Parker. *The Hidden Power of Social Networks - Understanding How Work Really Gets Done in Organizations.* Harvard Business School Press, Massachusetts, 2004.

R L Daft and R H Lengel. Organizational information requirements, media richness and structural design. *Manage. Sci.*, 32(5):554–571, 1986.

Wouter de Nooy, Andrej Mrvar, and Vladimir Batagelj. *Exploratory Social Network Analysis with Pajek (Structural Analysis in the Social Sciences)*. Cambridge University Press, 2005.

Deussen, Juling, and Thum, editors. *Nukath - Die Notebook-Universität Karlsruhe*. Universitätsverlag Karlsruhe, 2004.

Alan Dix, Janet E. Finlay, Gregory D. Abowd, and Russell Beale. *Human-Computer Interaction (3rd Edition)*. Prentice Hall, 2003.

Paul Dourish and Victoria Bellotti. Awareness and coordination in shared workspaces. In *CSCW '92: Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, pages 107–114, New York, NY, USA, 1992. ACM.

K. D. Eason. The process of introducing information technology. In *Behaviour and Information Technology*, pages 197–213. 1982.

Clarence A. Ellis, Simon J. Gibbs, and Gail Rein. Groupware: some issues and experiences. *Commun. ACM*, 34(1):39–58, 1991.

C. Erten, P. Harding, S. Kobourov, K. Wampler, and G. Yee. Exploring the computing literature using temporal graph visualization. Technical report, Department of Computer Sciences, University of Arizona, 2003.

Danyel Fisher. Using egocentric networks to understand communication. *IEEE Internet Computing*, 9(5):20–28, 2005.

Danyel Fisher and Paul Dourish. Social and temporal structures in everyday collaboration. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 551–558, New York, NY, USA, 2004. ACM Press.

Danyel Fisher and Joshua O'Madadhain. Jung 1.7.6 api documentation. Java library documentation, UCI, 2007.

C. Floyd and R. Keil. Softwaretechnik und betroffenenbeteiligung. In R. Oppermann, editor, *Beteiligung von Betroffenen bei der Entwicklung von Informationssystemen*, pages 137–167. Campus Verlag, Frankfurt, New York, 1983.

Linton C. Freeman. Visualizing social networks. *Journal of Social Structure*, 1, 2000.

Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Softw. Pract. Exper.*, 21(11):1129–1164, 1991. ISSN 0038-0644.

Ludwin Fuchs, Uta Pankoke-Babatz, and Wolfgang Prinz. Supporting cooperative awareness with local event mechanisms: the groupdesk system. In *ECSCW'95: Proceedings of the fourth conference on European Conference on Computer-Supported Cooperative Work*, pages 247–262, Norwell, MA, USA, 1995. Kluwer Academic Publishers.

H.P. Grice. Studies in the way of words. Harvard University Press, 1989.

Tom Gross and Christoph Oemig. Priminality: Towards human-centred instant messaging infrastructures. In C. Stary, editor, *Mensch & Computer 2005. Kunst und Wissenschaft - Grenzüberschreitung der interaktiven Art*, pages 71–80, 2005.

Tom Gross and Wolfgang Prinz. Modelling shared contexts in cooperative environments: Concept, implementation, and evaluation. *Computer Supported Cooperative Work*, (13):283–303, 2004.

Wolfgang Gräther and Wolfgang Prinz. Visualizing activity in shared information spaces. In J. Stephanidis and J. Jacko, editors, *Proceedings of 10th International Conference on Human - Computer Interaction*, pages 1096–1100, Mahwah, NJ, 2003.

Jonathan Grudin. Groupware and social dynamics. *Communications of the ACM*, 37(1):92–105, 1994.

Jonathan Grudin. Why cscw applications fail: problems in the design and evaluation of organization of organizational interfaces. In *CSCW '88: Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, pages 85–93, New York, NY, USA, 1988. ACM.

Carl Gutwin and Saul Greenberg. A descriptive framework of workspace awareness for real-time groupware. *Computer Supported Cooperative Work*, (11):411–446, 2002.

R.A. Guzzo and G.P. Shea. Group performance and intergroup relations in organizational experience. In H.P. Sims and D.A. Gioia, editors, *The Thinking Organization: Dynamics of Organizational Social Cognition*, pages 49–74. Jossey-Bass, San Francisco, CA, 1992.

Jeffrey Heer and Danah Boyd. Vizster: Visualizing online social networks. In *INFOVIS*, page 5. IEEE Computer Society, 2005.

Manfred Hein and Henner Zeller. *Java WebServices . Entwicklung plattformübergreifender Dienste mit XML und SOAP (Programmer's Choice)*. Addison-Wesley, 2003. ISBN 3827320712.

I. Herman and M.S. Marshall. Graphxml - an xml based graph interchange format. Technical report, Centrum voor Wiskunde en Informatica, 2000. URL `http://ftp.cwi.nl/CWIreports/INS/INS-R0009.pdf`.

H. Ulrich Hoppe and Katrin Gassner. Integrating collaborative concept mapping tools. In G. Stahl, editor, *Computer support for collaborative learning: foundations for a CSCL community, Proceedings of CSCL 2002*, Boulder, Colorado, 2002.

R. Johansen. *Groupware: Computer Support for Business Teams*. The Free Press, NY, 1988.

Michael Kerres, Steffi Engert, and Hans-Dieter Weckmann. *Das Duisburg eCompetence-Modell für Faculty Engagement: Gewinnung einer zweiten Welle von Lehrenden für den innovativen Medieneinsatz in der Lehre*, volume E-Learning-Strategien und E-Learning Kompetenzen an Hochschulen, pages 341–354. Bertelsmann, Bielefeld, 2004.

Guido Krüger. *Go To Java 2*. Addison Wesley Verlag, München, Germany, 2 edition, 2001.

J.R. Larson and C. Christensen. Groups as problem-solving units: Toward a new meaning of social cognition. *British Journal of Social Psychology*, 32: 5–30, 1993.

Jean Lave and Etienne Wenger. *Situated Learning. Legitimate Peripheral Participation (Learning in Doing)*. Cambridge University Press, 1991.

Jonathan Lazar and Jenny Preece. Online communities: Usability, sociability and users' requirements. In H. van Oostendorp, editor, *Cognition in the Digital World*, pages 127–151. Lawrence Erlbaum Associates Inc. Publishers, Mahwah, NJ, 2002.

Peter Mambrey, Reinhardt Oppermann, and Uta Pankoke-Babatz. *Computer und Partizipation: Ergebnisse zu Gestaltungs- und Handlungspotentialen*. Westdeutscher Verlag, Opladen, 1986.

Peter Mambrey, Volkmar Pipek, and Markus Rohde. *Wissen und Lernen in Virtuellen Organisationen. Konzepte, Praxisbeispiele, Perspektiven*. Physica-Verlag, 2003.

G. Meier, C. Prange, and L. v. Rosenstiel. Psychological perspectives of organizational learning. In M. Dierkes, A. Berhoin, J. Child, and I. Nonaka, editors, *Handbook of Organizational Learning and Knowledge*, pages 14–34. Oxford University Press, Oxford, UK, 2001.

M. Merleau-Ponty. *Phänomenologie der Wahrnehmung*. Walter de Gruyter Verlag, Berlin, 1966.

Stanley Milgram. The small world problem. *Physiology Today*, 2:60–67, 1967.

Dennis C. Neale, John M Carroll, and Mary Beth Rosson. Evaluating computer-supported cooperative work: models and frameworks. In *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 112–121, New York, NY, USA, 2004. ACM.

M. Newman. Coauthorship networks and patterns of scientific collaboration. In *Proceedings of the National Academy of Sciences*, pages 5200–5205, 2004.

Takahiko Nomura, Koichi Hayashi, Tan Hazama, and Stephan Gudmundson. Interlocus: workspace configuration mechanisms for activity awareness. In *CSCW '98: Proceedings of the 1998 ACM conference on Computer supported cooperative work*, pages 19–28, New York, NY, USA, 1998. ACM.

I. Nonaka and H. Takeuchi. *The Knowledge Creating Company*. Oxford University Press, Oxford, 1995.

D. Norman. *The Design of Everyday Things*. MIT Press, Cambridge, MA, 1998.

Donald A. Norman. *Things that Make Us Smart*. Addison-Wesley, Reading, MA, 1993.

Joshua O'Madadhain, Danyel Fisher, Scott White, Yan-Biao Boey, and Padhraic Smyth. Analysis and visualization of network data using jung. *Journal of Statistical Software*, 2005. To be published.

Marc Pallot, Rudolf Ruland, Stanislav Traykov, and Kjetil Kristensen. Integrating shared workspace, wiki and blog. In Klaus-Dieter Thoben, Kulwant S Pawar, Marco Taisch, and Sergio Terzi, editors, *Proceedings of The 12th International Conference on Concurrent Enterprising: Innovative Products and Services through Collaborative Networks*, June 2006.

Uta Pankoke-Babatz. *Designkonzepte für Systeme zur computergestützten Zusammenbarbeit unter Nutzung der Behavior-Setting Theorie*. PhD thesis, Universität Dortmund, 2003.

Uta Pankoke-Babatz, Wolfgang Prinz, and Leonie Schaefer. Stories about asynchronous awareness. In Francoise Darses, Rose Dieng, Carla Simone, and Manuel Zacklad, editors, *Cooperative Systems Design: Scenario-based Design of Collaborative Systems (Frontiers in Artificial Intelligence and Applications)*, pages 23–38. IOS Press,US, 2004.

Adam Perer. Making sense of social networks. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1779–1782, New York, NY, USA, 2006. ACM Press.

Christopher Plaue, Todd Miller, and John Stasko. Is a picture worth a thousand words?: an evaluation of information awareness displays. In *GI '04: Proceedings of Graphics Interface 2004*, pages 117–126, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2004. Canadian Human-Computer Communications Society.

L. Press. Lotus notes (groupware) in context. *Journal of Organizational Computing*, 2(3):315–319, 1992.

Wolfgang Prinz. Awareness. In Gerhard Schwabe, Norbert A. Streitz, and Reiner Unland, editors, *CSCW-Kompendium*, pages 335–350. Springer, Berlin, 2001.

Wolfgang Prinz. Nessie: an awareness environment for cooperative settings. In *ECSCW'99: Proceedings of the sixth conference on European Conference on Computer Supported Cooperative Work*, pages 391–410, Norwell, MA, USA, 1999. Kluwer Academic Publishers.

Wolfgang Prinz, Uta Pankoke-Babatz, Wolfgang Gräther, Tom Gross, Sabine Kolvenbach, and Leonie Schäfer. Presenting activity information in an inhabited information space. In David N. Snowdon, Elizabeth F. Churchill, and Emmanuel Frécon, editors, *Inhabited Information Spaces: Living with your data*, pages 181–208. Springer, London, 2003.

G. Probst, S. Raub, and K. Romhardt. *Wissen managen. Wie Unternehmen Ihre wertvollste Resource optimal nutzen*. Gabler, Wiesbaden, 2nd edition, 1999.

PythonSoftwareFoundation. Python programming language, 2007. URL `http://www.python.org`.

Günther Ropohl. Philosophy of socio-technical systems. *Society for Philosophy and Technology*, 4(3), 1999.

Robert Rosenthal and Lenore Jacobson. Pygmalion in the classroom. *The Urban Review*, 3(1):16–20, September 1968.

L. Schäfer, U. Pankoke-Babatz, W. Prinz, A. Fatah gen. Schieck, and A. Oldroyd. Docudrama. *Virtual Reality*, 7(1):43–53, 2003.

Michael F. Schwartz and David C. M. Wood. Discovering shared interests using graph analysis. *Commun. ACM*, 36(8):78–89, 1993.

Christian Seeling, Wolfgang Prinz, Rudolf Ruland, and Andreas Becks. Kooperatives arbeiten unter der lupe. In *Proc. of Mensch & Computer*, 2007.

B. Shneiderman. *Designing the User Interface*. Addison-Wesley, New York, 3rd edition, 1998.

Lucy Suchman. Do categories have politics? *Computer Supported Cooperative Work*, 2(3):177–190, September 1994.

P.J. Thomas. Introduction. In P.J. Thomas, editor, *CSCW Requirements and Evaluation*, chapter 1, pages 1–9. Springer Verlag, London, 1996.

Jonathan Trevor, Thomas Koch, and Gerd Woetzel. Metaweb: bringing synchronous groupware to the world wide web. In *ECSCW'97: Proceedings of the fifth conference on European Conference on Computer-Supported Cooperative Work*, pages 65–80, Norwell, MA, USA, 1997. Kluwer Academic Publishers.

J. W. Turkey. *Exploratory Data Analysis*. Addison-Wesley, Reading, MA, 1977.

W3C. Xml, 2006. URL `http://www.w3c.org/XML`.

Paul Watzlawick, Janet H. Beavin, and Don D. Jackson. *Menschliche Kommunikation*. Huber, Bern, 10th edition, 1969.

D.M. Wegner. Transactive memory: A contemporary analysis of the group mind. In B. Mullen and G.R. Goethals, editors, *Theories of Group Behavior*, pages 185–208. Springer Verlag, New York, 1986.

Eric W. Weisstein. Bipartite graph. *Math World - A Wolfram Web Resource*, 2002. URL `http://mathworld.wolfram.com/BipartiteGraph.html`.

Etienne Wenger. *Communities of Practice. Learning, Meaning and Identity*. Cambridge University Press, Cambridge, UK, 1998.

P. Wilson. *Computer-Supported Cooperative Work*. Intellect Books, Oxford, UK, 1991.

Langdon Winner. Do artifacts have politics? In Judy Wajcman and Doald MacKenzie, editors, *The Social Shaping of Technology*, pages 26–38. Open University Press, 1986.