Helge Kahler, Volkmar Pipek (2004):
Tailoring together: A systematization and two cases.
In International Reports on Socio-Informatics (IRSI)

Tailoring together: A systematization and two cases

Volkmar Pipek
Laboratory of Human-computer Interaction and Group Technology
Department of Information Processing Science
University of Oulu, Linnanmaa
FIN-90570 Oulu, FINLAND
volkmar.pipek@tol.oulu.fi
and
IISI - International Institute for Socio-Informatics
Wilhelm-Levison-Straße 22
D-53115 Bonn, GERMANY
volkmar.pipek@iisi.de

Dr. Helge Kahler
IISI - International Institute for Socio-Informatics
Heerstr. 148
D-53111 Bonn, GERMANY
helge.kahler@iisi.de

Abstract. In this paper we depict collaborative aspects of tailoring software. We provide a categorization distinguishing between three levels of intensity of user ties regarding tools usage (“shared use”, “shared context”, and “shared tool”) and discuss approaches to support collaborative tailoring in these scenarios. For the two levels with the most intense ties (“Shared Context” and “Shared Tool”) we provide the relevant theoretical background as well as empirical evidence from our own fieldwork. Additionally, we point out existing problems with collaborative tailoring support in “shared infrastructure” scenarios, which reflect the technological richness at today’s workplace. Finally, it turns out that there is a need as well as an opportunity for a broader technological support of processes of technology appropriation including better means of communicating and negotiating about aspects of the joint use of a software artefact.

1 Introduction

More often than ever software is involved in the collaboration of computer users in offices. Thus, the way a software product is implemented, configured, and used influences the collaborative work of these users. In this paper we describe approaches to support end-users in collaboratively finding and implementing tool configurations that are adequate for their form and intensity of collaboration. “Tailoring” has been defined by Henderson and Kyng (1992) as ”changing stable aspects of an artefact” and distinguished from use as persistent manipulations that are not “being made to the subject matter of a tool” (e.g. a text), but “to the tool itself”[1]. Traditionally, approaches to improve the “Tailorability” (Trigg et al. 1987, Henderson and Kyng 1992) of software artefacts address the improvement of the necessary artefact flexibility (e.g. Malone et al. 1992, Stiemerling and Cremers 2000, Wulf 1999). But offering the necessary flexibility to make tools fit diverse and changing work contexts is only the first step of offering support. A deeper understanding of the role tailoring plays in the appropriation processes of individuals, groups and organizations has lead to new ideas to also support tailoring *as an activity* within the tailored artefacts. Being aware of the organizational aspects of tailoring activities we explore opportunities to support these activities with additional functions.

Especially longitudinal studies of groupware systems (Karsten and Jones 1998, Pipek and Wulf 1999, Törpel et al. 2003) show that tailoring an application is only the technological condensation of an individual or social activity of designing a work setting for performing a certain task. In line with earlier

[1] We use the terms “configuration” and “tailoring” synonymously, but with the former having a more technology-related notion, in contrast to the latter having a more socio-technical notion.

experiences (Stallmann 1981, Malone et al. 1988, Mackay 1990) they show that tailoring activities often involve user-user interaction and collaboration. So, the "individual" decision how to tailor an artefact is to a certain extent always also social, since large parts of the knowledge used in the decision processes usually have been acquired through social interaction (e.g. knowledge on the capabilities of computers and tools, knowledge on the task the tool should serve in, etc.). So there is always a notion of cooperation in a tailoring activity, as it is in every activity of technology appropriation processes.

This general consideration of the sociality of tailoring processes has to be concretized to discuss possible opportunities for offering technical support for collaborative tailoring. To discuss existing approaches as well as new challenges, it is necessary to give some structure to support scenarios. In the next section, we develop four scenarios of collaborative tailoring that guide us through our discussions in this paper. Focusing on technical support for particularly collaborative tailoring we will concentrate our discussion on those scenarios that show the biggest need for collaboration support. We close this paper by describing new challenges for collaborative tailoring at modern workplaces and discuss extensions of tailoring interfaces to support the appropriation of groupware technologies in a broader sense.

2 The collaborative dimension of tailoring activities

What are the important aspects that motivate or even enforce cooperation in tailoring activities? Abstract perspectives on tailoring functionality have usually been dominated by distinguishing different tasks and complexity levels of tailoring interfaces (Henderson and Kyng 1991, Mørch 1997).

In the literature on tailorability collaborative aspects have been mentioned at several occasions since the early 1980ies. Already Stallman (1981) reports that users not only think of small changes and try them, but also pass them over to other users. Mackay (1990) researched how people actively shared their tailoring files with each other. Oberquelle (1994) proposes a classification of groupware tailoring distinguishing tailoring actors, who can be individuals or a group, from persons affected by a tailoring activity, who can again be individuals or a group (see Figure 1). This can also be used to classify collaborative tailoring. Different aspects and intensities of collaborative tailoring of a single-user software product and of groupware fit in the resulting four categories:

- Individualization: individuals tailor for themselves and are the only ones affected by the tailoring activity – e. g. individual keyboard shortcuts or the window layout of an individual email client;

- Tailoring effective for group: individuals can tailor for a whole group who then agree or are obliged to use the tailoring files – e. g. a system administrator or expert user provides a letterhead to be used by the group;
- Individualization supported by group: a group can tailor synchronously or asynchronously for its members to use and change the tailoring file - e. g. several persons work on collection of macros that individuals can use;
- Group tailoring: a group can tailor synchronously or asynchronously and its members agree or are obliged to use the tailoring files – e. g. several persons work on the introduction of semi-structured email templates valid for the whole group.

		Actors	
		Individuals	**Group**
Persons Affected	**Individuals**	Individualization	Individualization supported by group
	Group	Tailoring effective for group	Group tailoring

Figure 1. Classification of Collaborative Tailoring (Kahler 2001b, following Oberquelle 1994))

The strict borders between the four different categories mentioned blur when we try to apply them to practical examples (cf. Kahler 2001b, p. 28). It is not easy to locate the different accountabilities in the process that lead to a tailoring activity and in the process of the tailoring activity itself. While it is usually clear who actually worked with the tailoring interface of an application, it is not clear whose assumptions, ideas and decisions influenced a new tool configuration.

The Oberquelle model focuses on the distinction of one or more actively designing user on the one hand (*the* tailoring activity), and the passive "target users" on the other. This distinction focuses too much on the tailoring activity itself, and it does not represent additional activities of counselling, discussing, evaluating, validating, idea creation, etc. that are also very relevant for collaboration success in tailoring activities.

We believe that, to understand and support collaboration in tailoring activities, it is necessary to distinguish ideas and approaches according to what actually motivates users to collaborate regarding the configuration of software tools. The classification we propose here aims at differentiating the field along the interests and technological ties that bind users together in the process of collaborative tailoring. These are, in our eyes, motivated by the organizational context of the users as well as by the architectural properties of the application to tailor. We believe that it is useful to distinguish four levels of ties that bind users together and motivate cooperation regarding the (re-)configuration of tools.

2.1 Tailoring in "Shared Use" scenarios

On the first level it is the tool usage itself that serves as a common denominator for cooperation. Users can be perceived as a "Community of Interest" regarding the usage of a tool. Tasks and activities may be related to acquiring knowledge about aspects of tailoring covering possible tool configurations as well as alternatives for tool replacement (cf. Stallman 1981, Robertson 1998, Kahler 2001b). The common task of individualization of the tool is performed individually, but the existence and liveliness of user forums on single-user-applications (e.g. text processors) on the web indicate the social dimension of the task (see Figure 2).

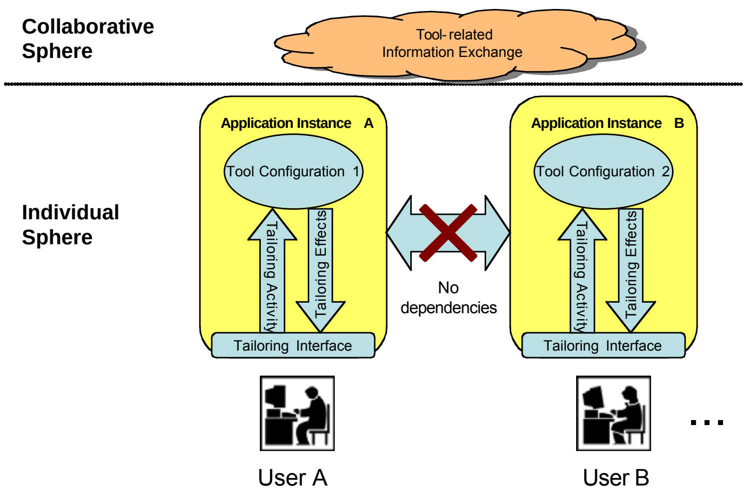

Figure 2. Collaboration in "Shared Use" scenarios

Almost any office application can be considered as an example in this category. If we look a common modern text processor, we usually find support for several tasks in the document types and functions it offers (Text, Thesis, web page, circular letter, etc.) and for several use situations (e.g. language contexts). Configurations affect how these functions modify the document (e.g. the automated spell checking function) or how it is being presented to the user. Though software producers usually offer manuals and training support, existence and use of several newsgroups and web discussion fora for some text processors show that users are aware of other users and their expertise. The problems encountered when trying a new task are solved by discussing possible changes with other users (Mackay 1990).

2.2 Tailoring in “Shared Context” scenarios

The interdependencies between different users and their usage of tools increase dramatically if the users collaborate. Both, a shared task as well as a shared organizational context, add new dimensions even to configuring single-user tools. A standardization of tool configurations may lower costs of computer administration and ease tool-related communication among users in an organization. Collaborating on a common task may require an agreement on tool versions and technological standards (e.g. file formats to use) among the users involved (even beyond organizational borders). In general, tool configurations of one user may have effects on the work of other users as well. Social interaction regarding tool configurations in these settings is more focussed and the shared context may support these interactions. Robertson (1998) described the occurrence of “tailoring cultures” in small design companies. So, these settings do both, they pose new requirements as well as offering new options for the technical support of tailoring. We call these settings “Shared Context” scenarios when we address technical support for collaborative tailoring (see also Figure 3).

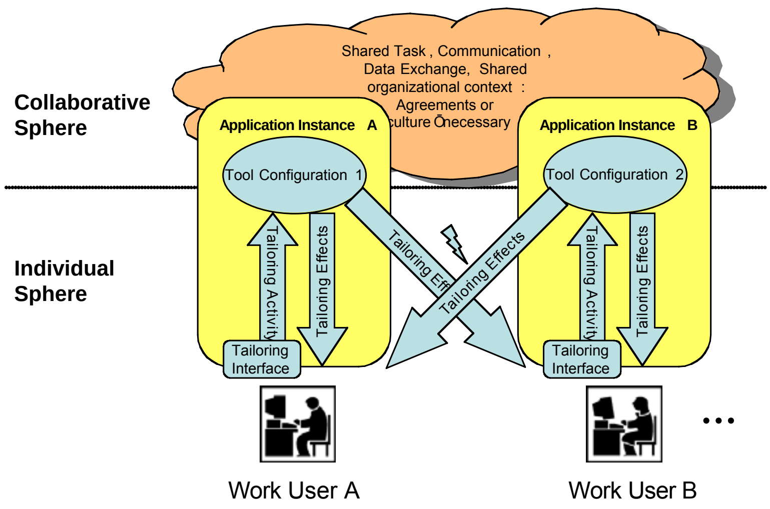

Figure 3. Collaboration in "Shared Context" scenarios

If we again consider the use of a text processor in an organization, we find use patterns where configuration settings are being passed around because users are sufficiently familiar with each other’s tasks and their similarities (Kahler 2000). Tailoring can be delegated to other users that have similar tasks, and the results are used without necessarily having to understand how tailoring has been done. A shared task forces users to agree on some aspects, e.g. document formats and naming conventions for the documents produced. However, users are basically still free in deciding on the configuration of their own text processor instance. They have to perform the tailoring individually at their own workstation, even if it is only following tailoring instructions of other users.

2.3 Tailoring in "Shared Tool" scenarios

In settings where a groupware tool is used for collaboration, it results in even stronger interdependencies (e.g. Pipek and Wulf 1999) of configuration decisions, and there are cases where for technical reasons there can be only one valid configuration (see below) for a group of users. In these cases, after an agreement on the desired tool configuration is achieved, it is impossible or disadvantageous to deviate from the common ground found, and the decision of individuals to disagree may deny them tool usage at all. The ties between the users are even stronger, and call for a collaborative solution for the tailoring process itself. We will address this context of technical support for collaborative tailoring as a "Shared Tool" scenario (see Figure 4).

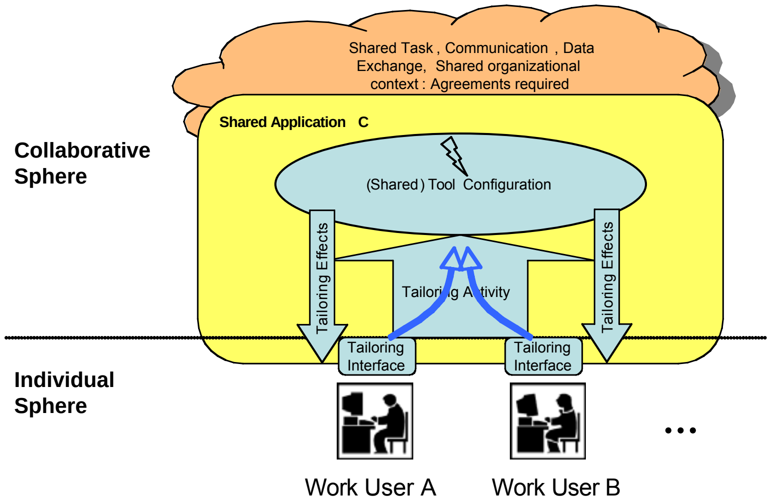

Figure 4. Collaboration in "Shared Tool" scenarios

An example of a "Collaborative Tool" scenario is the configuration of access rights for the shared workspaces of a groupware. There can only be one set of access rights of a workspace, so the structure of the workspace and the visibility of documents it contains will be configured for all groupware users. In this case, users have to agree on the configurations appropriate for the necessities of collaboration as well as for the protection of privacy. We will later refer to another example with similar dynamics.

2.4 Tailoring in "Shared Infrastructure" scenarios

To catch the dynamics of modern workplaces it is necessary to extend the notion of the interdependencies described above also to tool "infrastructures" (Robertson 1998, Dourish 1999, Dourish and Edwards 2000, Dittrich et al. 2002, Hansson et al. 2003), where similar situations occur when separate but tightly interwoven tools and technologies are being used (see Figure 5).

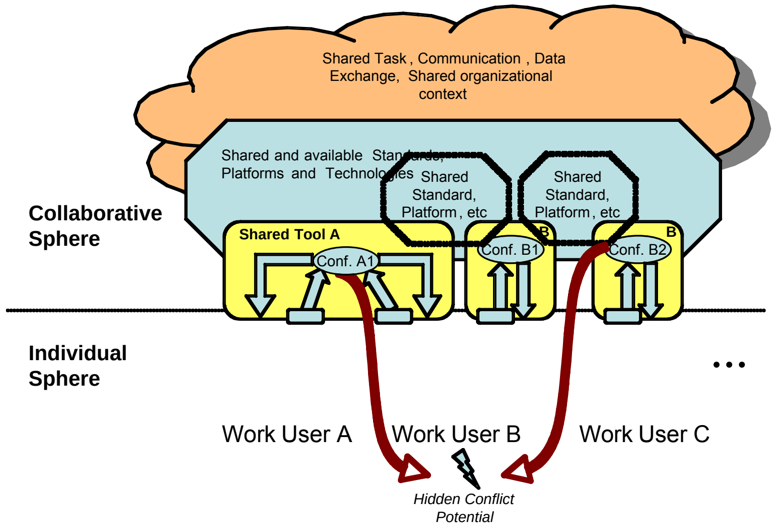

Figure 5. Collaboration in "Shared Infrastructure" scenarios

We can illustrate that with an example from our project practice: In an IT consultancy the decision to share contact information using the "LDAP" standard required an update of a Personal Information Management software. In this update, the designers of the tool have decided to change the default standard for address information exchange from the Electronic Business Card Standard "VCF 2.1" to version 3.0. Parts of the consultancy were involved in a research network where members shared contact information by storing ".vcf"-files in a shared directory, and suddenly the newest addresses from a central repository for Electronic Business Cards were not readable for some of the users. With the practices of sharing contact information via the LDAP protocol and of depositing VCF-files in a shared directory of a local network the users involved in both contexts introduced a hidden dependency into their work environment. In these "Shared Infrastructure" scenarios, dynamics similar to "Collaborative tool" settings can be encountered, but the dependencies are less obvious to users. These problems often occur because purchase of applications is a relatively uninformed decision where the applications' capabilities are more important than its dependencies, important future requirements are not anticipated and there is a lack of knowledge on the applications accountabilities (cf. Dourish 1997).

We now want to discuss different options to support the settings of tailoring that we sketched here.

3 Support for Collaborative Tailoring in "Shared Use" scenarios

"Shared Use" scenarios are those in which the users have the weakest ties; they may only share a common interest into a software artefact and its tailoring options. In contrast to "Shared Context" scenarios there may be no shared culture of usage or task congruency among the potential collaborators. There may be large variations of user interests regarding certain features of an application over space and time. A specific shared technological infrastructure can usually not be assumed beyond the basic Internet services (E-Mail, WWW, etc.). Consequently, there are only limited opportunities for a systematic support of collaboration in tailoring.

However, the emergence of tool-related discussion fora (partly initiated by the manufacturers themselves, e.g. Apple[2] and Microsoft[3]) on the web shows us that there is a solid baseline for support concepts in the idea of providing communication facilities for all kinds of tool-related problems. Similar positive experiences with newsgroups have also been reported in several studies (Mackay 1990, Okamura et al. 1994). Additionally, a number of cases have been published that showed a combination of usual user support measures (providing opportunities for finding help and giving feedback) for software products with measures to connect clients to each other. But these approaches usually reflected more organizational or philosophical concepts. The technology support did not go beyond the alternatives described above (Hansson et al. 2003, Dittrich et al. 2002).

Taking the need of communication about tools seriously, two aspects of the tailoring discussion should be highlighted. The first aspect is the need for tailoring objectifications (Henderson and Kyng 1991) as the entities that provide a meaningful closed subset of the possible manipulations of the software. It is also important to provide some kind of "tool ontology", names for these objectifications that may represent their meaning. Aside from using language, appropriate graphical representations of the applications and the tailoring

---

[2] http://discussions.info.apple.com/

[3] http://www.microsoft.com/communities/default.mspx

interfaces are also helpful. As a second aspect, the provision of easy access to these virtual meeting places is also important. Today, this requirement is often covered by providing links to these web pages in the applications themselves.

4 Support for Collaborative Tailoring in “Shared Context” scenarios

In this section we describe several aspects and approaches and one in-depth example for tailoring support in “Shared Context” scenarios. This is meant to clarify the qualitative difference of “Shared Context” scenarios as compared to “Shared Use” scenarios and how this new quality can be met by technical and organizational means.

4.1 Aspects and Approaches for “Shared Context” Scenarios

In “Shared Context” scenarios, of course all aspects described for “Shared Use” scenarios also apply. But with the shared user context, be it an organization or a shared project, additional dynamics evolve. A culture of collaboration is established between separate users, a fact that brings new dependencies as well as new options (e.g. the delegation of tailoring activities to colleagues).

4.1.1 Requesting and sharing ideas for Tailoring

Mackay (1990) did one of the first larger studies on the role of collaboration in tailoring activities. The study was conducted at two research sites. At the first site, 18 people using Information Lens (Malone et al. 1988) to tailor the management of their emails were observed over a period of three or more months. This included several interviews per participant and the collection of automatically gathered data. Mackay reports that several people shared Information Lens rules (i.e. text files containing information about the filtering of mails), including two manager-secretary teams who used the rules to support a standard form of communication. At the second site, a group of 51 users on a common project sharing Unix tailoring files were observed over a period of four months. The data gathered stem from one or more interviews per user and also included copies of their tailoring files. More than three-quarters of the participants received tailoring files from others since they had joined the project.

Depending on the job category (e. g. Manager, Secretary or Application Programmer) the different groups borrow and lend files with different intensity and have a different percentage (0% to 38%) of translators. To Mackay these are persons who actively share their files and talk to the recipients of the files. She concludes both cases by criticizing that staff members are often not rewarded for sharing tailoring files and requests that tailorable software should provide the ability to browse through others' useful ideas and that it should include better mechanisms for sharing customizations which then may serve to establish technical or procedural standard patterns.

Mackay (1991) studied the tailoring behaviour of 51 users of a Unix software environment over a period of four months. Four main reasons that lead to tailoring have been identified: external events like job changes or office moves, social pressure like contact to colleagues who suggest changes, software changes like breakdowns or upgrades (the latter often retrofitting new software to behave like the old version), and internal factors like spare time or running across a previously unknown feature. The topmost barriers for the persons she asked were the individual factor *lack of time* (cited by 63% of the users) and the technological factor that the software was *too hard to modify* (33%).

4.1.2 The importance of Expert Users

The role of a local expert was also highlighted by Gantt and Nardi (1992) who describe what they call patterns of cooperation among CAD users. They studied the use of a Computer Aided Design (CAD) system by conducting in-depth interviews with 24 informants and collecting and analysing the informants' CAD artefacts. They distinguish between local developers who write macros, programs and scripts and help end users in tailoring on one hand, and on the other hand gardeners as a sub-group of local developers. With gardeners, the informal position of a local developer has evolved into a formal or semi-formal position. They are responsible for writing and disseminating standard macros and programs at the corporate and department level. Usually, a gardener has both domain and computer knowledge and often starts from the domain side and then acquires the necessary computer expertise. Gantt and Nardi support the contention that the activities of local experts should be recognized and promoted since a local expert, and particularly a gardener, can save the organization's time and money by offering valuable resources, like macros and programs to the entire group. They admit, however, that it may be difficult to find a person with the right combination of technical and social skills.

Nardi and Miller (1991) report that spreadsheets offer strong support for cooperative development of applications. They present results from an in-depth-study where they conclude that spreadsheet co-development is the rule rather than the exception and that spreadsheets support the sharing of both programming and domain expertise. For spreadsheet applications it can be argued that using and tailoring them are closer together than for many other applications, since their usage - in the sense that you just put in numbers and calculate something - implies the prior work of defining code behind the spreadsheet's cells which is responsible for the calculation. This is usually done by persons who have domain knowledge and, as Nardi and Miller note, usually done cooperatively. Considering the fact that more and more off-the-shelf software needs tailoring and offers mechanisms for it, the presented results encourage the tighter integration of using and tailoring.

4.1.3 Tailoring between differentiation and standardization of tools

Contrary to a criticism voiced often, tailoring must not necessarily lead to an abundance of confusing individual configurations but may also help good solutions to become standards. Trigg and Bødker (1994) found an emerging systematization of collaborative tailoring efforts in a government agency. In their study, they examined the tailoring of word processors performed by four persons in a Danish administration over a year. Tailoring at their organization mainly means customizing the word processors button panels, macros, and standard forms. Trigg and Bødker explicitly distinguish tailoring from programming, since the latter moves from analysis to design and then to realization, while the former basically consists of trial and error where the starting point often is a personal solution that may become more stable and then used by several people after a constructive process of small improvements. Here, tailoring showed aspects of what Fischer and Giaccardi (2005) called unself-conscious design, where situated knowledge is more important than an explicit description and where people solve their own problems rather than those of others as in a self-conscious culture of design. Trigg and Bødker also observed that tailoring is often a collaborative process where the idea and the basic work is performed by the local developers who then pass on their partial solution to the programmer for improvement. Also, the learning process of tailors has distinctly collaborative character, as they ask each other and consider themselves to be on a learning staircase trying to move upwards. Over the time, the sharing of tailoring files had evolved from an opportunistic spreading where someone heard about tailoring done by a colleague and copied their tailoring files to a more systematic activity: ideas are conveyed to the local developers or the programmer who then implement them. While it is often argued that tailoring leads to an unmanageable abundance of individualized solutions, several aspects imply that tailoring in this organization does rather have a standardizing effect. Standards of particular text blocks and of macros and button panels that reflect the work practice can be developed and widely used because the organization explicitly supports individual and collaborative tailoring and the distribution of tailored files. Wulf (1999) provides an example how the complexity of a tailorable search tool can be managed by organizational and technical means.

4.1.4 Technical Support of Collaborative Tailoring

Wasserschaff and Bentley (1997) describe how they supported collaboration through tailoring by enhancing the BSCW Shared Workspace system, which is an extension to a standard web server providing basic facilities for collaborative work. It includes information sharing, document management, and event logging and notification. They designed multi-user interfaces for the BSCW system, which allow users to take a certain view on the data in the shared workspace. These Tviews can be added to the shared workspace as objects in their own right, so others may take them individually, use them, and modify them in the same way as documents and folders.

In their Buttons system MacLean et al. (1990) explicitly supported the sending of tailored files via email. They observed that via this opportunity, small improvements easily diffused throughout the user community and that a high amount of tailoring could be done by "begging, stealing or borrowing" (p. 178) appropriate buttons. Moreover, users often took buttons they had obtained from others as a basis to do some tailoring of their own by adding or changing something. MacLean et al. propose that the two possibilities to make systems more tailorable for workers are to make the tailoring mechanisms accessible and to make tailoring a community effort. After a while the users did not only share buttons with others but also over time appropriated buttons and started to perceive them as personal and positive. The notion of the importance of a community of people who tailor is supported by Carter and Henderson (1990). Based on their experiences with the Buttons system they claim that a Tailoring Culture is essential to the effective use of a tailorable technology. Carter and Henderson conclude that (p. 113) "tailorability is a relationship to rather than a property of technology. Tailorability addresses how technology fits into an organization and how groups and individuals make use of it."

The aforementioned examples stress that collaborative tailoring does not only occur among groupware users, but also in groups of users using the same software and thus being able to profit from the fact that this software is tailorable and that tailoring files may be exchangeable. Particularly the fact that more and more computers are connected to a local or wide area network creates the infrastructure to exchange tailoring files even of single user applications easily through the network. Therefore, the boundaries between collaborative tailoring of a single-user software product and a groupware become fuzzy (see also "Shared Infrastructure" scenario). We now look closer at one of our prototypes, which captures these dynamics.

4.2 Collaborative Tailoring of a Word Processor

Generic single user applications usually do not provide support to share tailoring files among its users. However, they are often tailored collaboratively. As described above there are several positive experiences with the sharing of tailoring files of single user applications, like word processors or spreadsheets. To support such collaborative aspects of tailoring single user applications, an extension to a common off-the-shelf software product that should allow the exchange of tailoring files was to be developed. In order to get a deeper understanding of how people collaborate in tailoring a word processor we conducted a field study. The result of the study was a number of different "collaborative tailoring-use" situations focusing on the exchange of document templates and toolbars. Based on an analysis of these use situations we implemented a tool, which provided collaborative tailoring functionality as "Microsoft Word" extension (Kahler 2001a).

4.2.1 Setting

To learn about users' habits and to inspire the design, we carried out a qualitative field study with users of Microsoft Word. 11 semi-structured interviews with users from 4 different fields were conducted (public administration, private company, research institute and home users).

The interviews started with general questions about the interviewee's qualification, their general task and the way they apply the word processor. In the following they were asked which tailoring functions were in use, which barriers hindered the usage of existing functions to tailor, whether and how collaborative tailoring did take place, and whether organization wide standards concerning the tailoring activities are existing. In the end of the interviews ideas concerning the design of support for tailoring activities and of improved user interfaces to ease tailoring were discussed. To be able to refer to the software, the interviews were performed next to the interviewee's computer.

The interviews took between 20 and 120 minutes with an average of about 45 minutes. They were audiotaped, transcribed and later analyzed. According to their self-estimation two interviewees had little to medium, two interviewees had medium, three interviewees had medium to good, three interviewees had good and one interviewee had very good knowledge about the word processor. Five of the interviewees were providing system support to other users in their organizations.

Together with the literature review these interviews were the basis for the requirements. The interviews are enriched by empirical studies concerning the usage and sharing of a tailorable search tool for groupware. A prototype of this search tool was presented, used and discussed in a workshop with users where some of the interviews about Microsoft Word had taken place. The workshop about the search tool and interviews about it revealed interesting aspects of sharing adaptations in this organization.

Depending on their field of application the interviewees reported about differences in the extent and the way tailoring is seen as a collaborative activity. To give an impression of this variety and to motivate the design, we will present the main statements of the interviewees concerning collaborative tailoring.

4.2.1.1 Use Situation I: Central Repository for Standardized Forms.

One group of persons interviewed were two system administrators and two researchers from a German national research institution.

The interviewees reported about little collaborative tailoring activities since their tasks are rather individualized. Nevertheless the organization uses an intranet to provide certain document templates in a standardized manner. These templates are created and updated by a central organizational unit. All the other users can just copy these templates. Ideas for new forms have to be proposed to that unit.

4.2.1.2 Use Situation II: Collaborative Tailoring and Organization-Wide Distribution.

Four of the interviewees were working for the representative body of a German state in the federal capital. The organization had been equipped with generally available desktop computers about three years before the interviews were made.

All of the interviewees reported about a rather intense exchange of adaptations. One of the employees from the administration site reported how she created a document template together with a colleague. Both of them carried out parts of the whole job. Then she put her part of the template on a disk and carried it to her colleague who pasted the parts together.

To make document templates publicly available, the representative body used the groupware system whose functionality offered shared workspaces to exchange documents. To publish newly created document templates within the whole organization, a specific workspace was used. Because several of the employees suffered from lacking computer skills, the right to change these templates or to add in new templates was reserved to the system administrator. The templates were seen rather as a collective resource than as a means to enforce organizational standards.

4.2.1.3 Use Situation III: Shared Document Templates and Notification of Users.

An experienced user working in the marketing division of a car-manufacturer described how he had implemented department-wide standards for presenting certain data by means of tables. Before, everybody in ”his” department had used his own mode to create these tables. He started to standardize the layout of these tables by creating a first template containing some macros. He then discussed it with his colleagues. Having found an agreement with them, he asked his boss for a final approval. When everything was set up, the interviewee informed his colleagues verbally about the location of the shared template on the LAN.

4.2.1.4 Use Situation IV: Experience Transfer Among Insulated Home Users.

The interviewees working at home were two law students. They used their word processor to work out law cases, which they had to deliver for getting certain credit points. Each student has to write these papers almost every semester by himself.

The students reported about few collaborative tailoring activities. One of them describes these rare occasions as follows. Occasionally when he meets other students applying the same word processor he sees an unknown tailoring feature – for instance a new toolbar. In such a case he asks how the feature has been constructed. After receiving a demonstration he goes home and tries to repeat on his own system what he has seen before.

4.2.2 From use scenarios to design requirements

While use situation IV (“Experience transfer among insulated home users) just deals with experience transfer, use situations I to III are based on an exchange of adaptations. In these use situations, this common use of adaptations is either technically non-supported (exchange of floppy disks) or supported by tools, which are realized apart from the word processor (intranet, LAN directory, groupware application). Both of these solutions seem to be problematic because they require the users to leave the application to acquire the adaptations. Therefore, it seems worthwhile considering to integrate the support for collaborative tailoring into the word processor’s functionality.

To design such an integrated support, the following considerations seem to be of special importance. Depending on the state of a tailoring activity there are different groups of users involved in carrying them out (e. g., use situation II - Collaborative Tailoring and Organization-Wide Distribution). The extent to which adaptations are reasonably shared obviously corresponds to the tasks that are supported by them. Such a task can be specific to an individual (e. g., use situation IV - Experience Transfer Among Insulated Home Users), a group or a department (e. g., use situations II - Collaborative Tailoring and Organization-Wide Distribution -and III - Shared Document Templates and Notification of Users) or even a whole organization (use situation I - Central Repository for Standardized Forms).

Thus, support for collaborative tailoring should allow differentiating among various groups of users when sharing adaptations. Sharing of adaptations can require different mechanisms. There are obviously situations where mail support seems to be appropriate to exchange adaptations. E.g., in cases an experienced user builds an adaptation especially required by a user for the task at hand, a mail tool seems to be the appropriate technical support for distribution. On the other hand, in case adaptations by a specific user are not required instantly, a publicly accessible store allows selecting among these adaptations at the moment required by the task at hand (e. g., use situations I to III).

Finally there is a need to make users aware of the fact that somebody else has produced an adaptation with relevance to them. An integrated tool to support sharing of adaptations could provide additional awareness within the system.

Evaluating the use situations and summing up the results of the final discussion with the interviewees, the following main requirements for the tool emerged. It turned out that this empirical evidence is in line with theoretical and empirical work described in the literature about tailorability:

- Tight integration in the word processor application;
- Mechanisms for sharing, sending and receiving tailoring files
  - A public store to provide a location to exchange tailoring files;
  - Mailing mechanisms for users to be able to send tailoring files directly to other single users and groups of users;
  - A private workspace for tailoring files, that may be copies of files from the public store or files received from others via the mailing mechanism;
- An awareness service that notifies users about modifications of tailoring files.

Consequently, the respective features were provided in a prototype as an extension ("add-in") to Microsoft Word implemented in Microsoft Visual Basic for Applications. The features included *sharing document templates and toolbars*, *identifying tailoring files in shared workspaces* by means of annotations and a preview mode, and a *notification service* to inform users of the arrival of a new tailoring file in their inbox.

Finally, a usability test of this add-in has been conducted by using the method *constructive interaction for testing collaborative systems - CITeCS* (Kahler 2000).

4.2.3 Evaluation

The usability test resulted in findings on different levels. Most obvious, there were some shortcomings of the interface. Some buttons created misunderstandings and needed to be renamed. One button's name needed to be changed from "delete" to "deactivate" since the action that it triggered was hiding a toolbar. Another button needed to be renamed from "copy" to "adopt" where participants could decide if they wanted to move an adaptation that was sent to them to their private folder.

Moreover, it became clear that there was a need to be able to delete an adaptation from within Microsoft Word 97 rather than having to use the file manager. This also resulted in the suggestion to introduce the role of an administrator as a person who is allowed to delete adaptations in the public folder.

All of the participants considered the possibility to save, connect and distribute adaptations to be very helpful for their work. Although not all participants were expert users they were all able to use the tailoring functionality and the sharing functionality. The overall usability of the tool was perceived to be good. One participant explicitly said that he would tailor his word processor more in the future since he now knew how to do it and was no longer afraid that the tailoring activities would make the software unusable. This was mainly due to the preview mode. The two participants who were network administrator and experienced user said that such a distribution of adaptations would be very helpful for their organizations. The discussion following the tasks revealed that the participants' conceptual model of how the distribution of files worked was very close to how we, the designers, had intended and implemented the distribution.

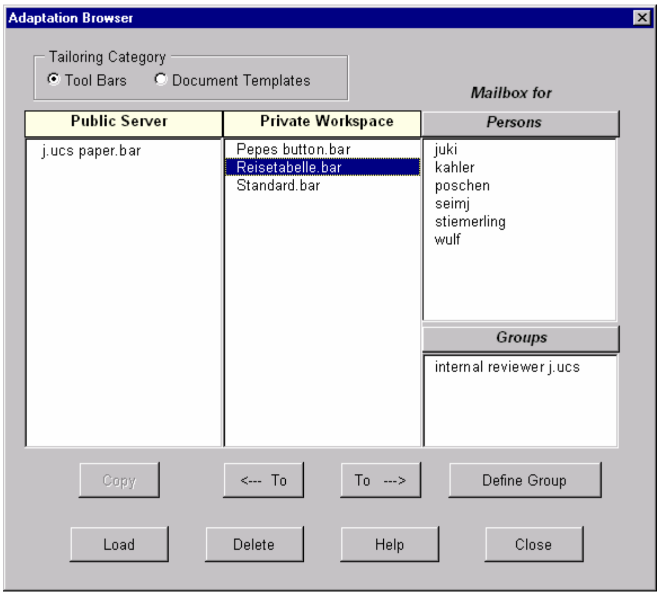

Figure 6. The browser to share adaptations

Besides the qualitative usability test a quantitative evaluation in which 32 persons participated was conducted (Kahler 2001a). The aim of this quantitative evaluation was to find out how the option to send and receive tailoring files with the extension performs in comparison to the sending mechanism already implemented in Microsoft Word in the file menu. Letting the participants rank the different solutions' abilities in sending and receiving revealed that the extension performed as well as the built-in sending mechanism. Despite the fact that the extension was only a prototype with the first version of the user interface the participants could obviously detect the value in the strong integration and the enhanced functionality of the extension.

The word processor case showed that even for single user applications collaborative aspects of tailoring are an issue. It also showed that, with relatively little effort, several important features for collaborative tailoring can be supported: the word processor was already tailorable on several levels (choosing from alternative behaviours, creating macros) and extensible by an add-in implemented in Basic; most organizations have their computers connected and already support some forms of shared workspaces. So the technology and infrastructure is mature enough to connect people who tailor single user applications individually in order to be able to introduce collaborative aspects.

Moreover, the case showed how fuzzy the borders between use and tailoring are: The tailoring files most interesting to the word processor users were document templates, which are basically write-protected documents. However, as templates they were considered to be tailoring files, since they modified the functionality of the word processor in use, because the users could start the word processor by double-clicking a template and immediately were able to do their task of, say, writing a formal request.

4.2.4 Remaining problems and future extensions

Our results hint to the fact that a tool for sharing tailoring objects as described may increase the frequency of tailoring activities. We assume that such a tool may also serve as a medium that encourages groups to discuss group standards, e.g. for letter templates that then can be shared. The systematization of customizations (Trigg and Bødker 1994) resulting from a collaborative tailoring process would then contribute to common norms and conventions needed for collaborative work (Wulf 1997).

Suggestions for the use of such a tool cannot be restricted to technical design requirements but must include organizational suggestions as well. We are convinced that the establishment of a ”gardener” (Nardi 1993) or ”translator” (Mackay 1990), e.g. a local expert responsible for the coordination of tailoring activities, is a vital part of tailoring measures in organizations.

Right now it seems that adaptations are most usefully applied in the organizational context of their emergence supporting the tasks they are made for. The sharing tool in its current form is most helpful for small work groups with a rather similar work context.

Our hypothesis about the technical and organizational scalability of such a tool is that the model of public and private spaces and the distinction between creator and user of the artefacts need to be enhanced to more than two levels when the group size exceeds a certain limit. Like in shared workspaces for general purpose, a more sophisticated access control model is needed (Pankoke and Syri 1997). Meta-information like annotations made by an adaptation’s creator may help to compensate for part of a lacking context. If such a tool would allow distributing adaptations worldwide, e.g. via the World Wide Web (WWW), one could even think of supporting global teams or even establish widely accessible libraries for adaptations. Whether this is, however, reasonable in the light of the poverty of organizational and task context is unclear. How context could possibly be provided and how large groups of participating contributors can be handled may be learned from experiences in distributed software development. This is particularly interesting when taking place without existence of a formal organization as in the case of the distributed development of Linux and its components. Anecdotal evidence shows that questions of ownership and membership play an equally important role here as they do in "ordinary" organizations and settings (Divitini et al. 2003).

However, in our experience it is clear, that collaborative tailoring does not scale easily. As always the question remains open how much administrative work the participating individuals are willing to contribute for the benefit of a group or organization and how much administrative effort is still reasonable to stay on the profitable side of collaborative tailoring. More refined tools to measure this and more refined categories to weigh the individual and group pains and gains against each other are needed.

An alternative to a tool presented above is the embedding of such a mechanism to exchange Microsoft Word related adaptations into a generic organizer of adaptations belonging to different applications. This organizer could combine mail mechanisms (or even be part of an email client) with the operating systems' functionality for access rights or shared workspaces and an enhanced explanation and commenting functionality.

5 Support for Collaborative Tailoring in "Shared Tool" scenarios

We now describe different approaches for technological support of „Shared Tool" scenarios. Again, approaches for tailoring support in "Shared Use" or "Shared Context" scenarios also apply here. But the necessity to agree on (parts of) a tool configuration increases the interdependencies among users again and requires a different kind of support. In the first part of this section we describe the relevant aspects of several approaches towards collaborative tailoring in the context of "Shared Tool" scenarios. In the second part we give an in-depth example for supporting the collaborative design of a tool configuration.

5.1 Aspects and Approaches for "Shared Tool" Scenarios

We describe four approaches that cover also the collaborative dimension of tailoring a shared tool, in most of these cases a Groupware application. All of these approaches highlight different aspects of the role of collaboration in tailoring. We extend the discussion by describing one approach in more detail later.

5.1.1 Tailoring groupware as a collaborative design process

Oberquelle (1994) investigated tailoring as a collaborative design process and proposed groupware support for several tasks within the tailoring process. He identified five tasks:

- *Discuss inadequacies:* A groupware system could provide an opportunity for “Meta-communication” to allow users to discuss inadequacies of the groupware system.
- *Conceptualize alternatives:* With the acknowledged requirement of “objectification” of tailoring alternatives given, this especially means to allow users to choose between or model a tailoring alternative.
- *Evaluate alternatives and decide:* All users should participate in the evaluation and the decision upon the tailoring task to process. The groupware system, with its messaging capabilities, could support this.
- *Implement the alternative chosen:* Finally, the tailoring alternative chosen should be implemented.
- *Notify affected users*: All users who are somehow affected by the tailoring alternative chosen should be notified that the change has been implemented now. Additionally, explanations of the “how” and “why” of tailoring can be given.

It is important to note that these task descriptions are not meant to be a model of consecutive, clearly distinguishable phases. Since tailoring processes can become quite complex regarding the functions modified in a tailoring activity, this would be an inappropriate restriction of the processes’ flexibility. The ideas presented provide a first weak process model of collaborative tailoring activities.

5.1.2 The “Meta-Design” perspective

We now only want to relate to some aspects of the “Meta-Design”-philosophy (described in Fischer and Scharff 2000) that we consider important for collaborative tailoring. The baseline of the argumentation is that most new technologies developed within the IT sector treat the user more as a consumer than as an active participant in a technological setting. Fischer and Scharff call for a notion of the user as an active (co-)designer of the technological settings he works with. Several perspectives on developing adequate design environments form the “Meta-design” philosophy. In a broader description of the idea and its context, Fischer (2002) called for designing technical infrastructure, organisational issues (learning environment and work organization) and a socio-technical environment for design-in-use at design time.

In this context, tailoring has the notion of a design-in-use activity where several designers (professional developers and users) are involved into designing the configuration of a tool. We point out some aspects and experiences that are relevant for particularly collaborative tailoring as an activity.

Fischer and Scharff stress the importance of *domain orientation* for design environments. They should reflect the entities that form the design problem as well as those relevant for describing possible solutions. In the design problem described, this is the domain of urban planning and mass transportation. The domain orientation does not only reflect in the concepts and abstractions used but also in their form: Tangible representations of the design problem result in more appropriate opportunities for articulations by participants.

In the example domain these tangible representations also play an important role for *collaborative representations*. Collaboratively developed representations are important for developing both, an individual understanding of the concepts and interests of others as well as a manifestation of the shared notions regarding the design problem.

In a related paper, Fischer and Ostwald (2002) described the combination of an *action space* with a *reflection space* as a basic useful architecture for collaborative design (e.g. in Urban Planning). In a way this can be associated with collaborative tools having a “use space” (the ordinary use interface) and a “tailoring space” (where there could be room for reflection).

The special facet of the “Meta-Design” idea to support *design communities* (instead of only individual designers) also is very relevant for collaborative tailoring. This aspect stresses that the design activity (or tailoring activity) also comprises secondary activities of learning, communicating and collaborating regarding issues important in the design process. These are also important to support. Finally, communities may provide a context of motivation and reward for participating in design activities (“good community citizenship”).

While pointing out important directions to think about, many of the ideas have to be concretized to be able to guide the development of support for collaborative tailoring. Another important problem of the approach is that it mainly addresses “big”, complex and separate design activities while tailoring activities are often much smaller and tightly interwoven with the work practice of the users.

5.1.3 The “Cooperative Hypermedia Approach”

Wang and Haake (2000) present CHIPS, a hypermedia-based CSCW toolkit with elaborated abstraction concepts (role models, process models, cooperation modes, etc.) in a three-level modelling scheme (meta-model, model and instance) that allows users to describe and tailor their cooperation scenarios. Generally based on an open hyperlink structure the system is extendable on any modelling level and thus should be able to support every cooperation scenario that may occur. In this respect it is similar to CSCW toolkits earlier proposed by Dourish (1995) or Malone et al. (1992).

Wang and Haake focus on the notion of tailoring as a collaborative activity. The main example they use to present the toolkit is the collaborative development of a work environment for a newly formed team. Their toolkit provides access rights systems and discourse representation facilities oriented at the method of issue-based information systems (IBIS, Rittel 1973, Conklin and Begemann 1988). The tailoring activity is incrementally described and performed within the hypermedia system as any other collaborative activity. In their example they use a weak process model with the steps idea creation, discussion of alternatives, decision-making and implementation that is similar to the ideas of Oberquelle (1994). The approach does not explicitly implement this process but the tools (especially the model and instance editors) are designed to facilitate that process.

In our eyes, the benefits of the approach are its openness, flexibility and extensibility. Although they do not provide an example from a real application field, it is credible that the architecture would cover a large set of collaboration scenarios and associated tailoring activities. The major drawback is that the cognitive costs for end users to apply the approach are very high (it requires understanding the modelling layers, the abstraction concepts and the tools). This might be acceptable in complex or model-oriented work environments (e.g. software engineering), but even then depending on the granularity of the modelled descriptions the work costs of keeping the models up to date might well outweigh the perceived benefits.

5.1.4 Using "Multiple Representations" and "Application Units"

Mørch and Mehandijev (2000) aim explicitly at supporting tailoring as a collaboration process between end users and professional software developers. They see this as a long-term cooperation to continuously improve software artefacts and develop concepts to support communication among the stakeholders involved in these settings (designers and end users).

They propose to provide *multiple representations* of the software artefact as well as the continuing design process and the design decisions taken so far. These representations can, for example, represent code, or more abstract perspectives like control flow diagram, design rationales or other documents produced in the context of earlier tailoring/design processes. To provide a better overview they should be grouped into *application units* that consist of representations of different abstraction levels that belong to one aspect of the software's functionality. These ideas have been successfully tested in two cases, and proved to produce a higher transparency within the process of continuous tailoring of an application. Mørch and Mehandijev formulate the finding of appropriate representations for users with different experience backgrounds and skill levels as an important open issue, a fact that also shows in their example cases, where representations still orient much at programming concepts.

5.1.5 "Lazy evaluation" of tailoring alternatives

The approaches presented before cover tailoring activities that take place before the actual use situation occurs. As an important alternative, Wulf (1997, Wulf et al. 2001) described an approach of tailoring the access rights in a Groupware application *during* the actual use situation. Consciously delaying the specification of the access rights for every user or role in advance, the framework provides different ways of negotiating access to a document by means of communication channels between the requesting and the granting user. Contrary to the usual notion of tailoring as producing *persistent* changes, these changes are temporary and can be handled in very flexible ways. Implementation of these kinds of strategy complements the conservative notion of tailoring and may be especially important for settings where users with very heterogeneous interests share one tool, e.g. communities or virtual organizations (Stevens and Wulf 2002).

5.2 Collaborative Tailoring as "Integrated Design by Discourse"

In our own approach we developed and implemented ideas similar to the perspectives described above for the design of a tailoring environment for an awareness service of a Groupware. Our approach focused on supporting articulations regarding the tailoring problem and embedding the necessary facilities into the artefact itself. Aside from the sheer provision of discourse support we also offer support for the tailoring activity as a design process. As this is typical for "Shared Tool" scenarios, those users that are affected by the reconfiguration take the "designer" role. Our approach is elaborating on and extending the tailoring facilities of a full-strength industrial groupware product.

5.2.1 Tailoring as a collaborative design process

The basic idea is to support "meta-use" of the groupware features, to use these features to (re-)design the groupware itself. Since the crucial aspect of the collaboration during the tailoring activity is the communication among the users affected, we decided to draw on relevant experiences from the field of (collaborative) software design.

Besides a notion of the tasks and their order within this design process (we rely here on the descriptions by Oberquelle 1994, see also above), it is important to consider the context the users are in when they want to tailor their application.

- In general, tailoring does not belong to the primary work task of users. Users can only spend a very limited amount of time on the tailoring activities themselves as well as building up the knowledge necessary for a qualified participation.
- Collaborative tailoring can be a task where group-inherent conflicts may occur. It is necessary to not artificially restrict communications related to this task with a formal process or discourse model (see also Shipman and Marshall 1999).
- Tailoring as a design task occurs at different degrees of maturity of an organizations' groupware infrastructure. There can be no general assumption on the duration and perceived complexity of a tailoring activity.

In this example we further explore the solution space by drawing on earlier experiences with technological support for software design processes. We then show a prototypical implementation of a design environment for collaborative tailoring of an event distribution service of a groupware.

5.2.2 Design Process Support in Software Engineering

With our example for collaborative tailoring in “Shared Tool” scenarios we tried to relate to earlier support on Software Development Processes. Computer Science’s process view on the task of software engineering has manifested in process models as well as support concepts and tools (e.g. in Sommerville 2000). Some approaches focus on “process enactment” (Dellen and Maurer 1996, Garg and Jazayeri 1996), by modelling the process itself and keeping track of its state and the different design decisions that have been made. These approaches are similar to computer support for workflow management, resulting in similar models with (sub-) processes, roles and (partial) products. The strong focus on detailed modelling results in a significant overhead that makes these approaches attractive only for very complex design processes with high quality requirements.

Other approaches focus on the documentation of the design decisions (“design rationales”, Moran and Carroll 1996) and their dependencies. These approaches use argumentations to document the different alternatives and the decisions made. Beyond only documenting the process, some of the argumentation-based approaches also aimed at supporting the designer-designer interaction during a design process (Fischer et al. 1996, Reeves and Shipman 1992).

Some of these approaches relate to the idea of Issue-Based Information Systems (IBIS) proposed by Rittel (1972). In this concept to support the solution of complex (“wicked”) problems he proposed the use of persistent structured conversations using the speech acts “issue”, “position” (opinion on an issue) and “argument” (for or against a position), which have defined relations. This concept has been implemented and improved (Conklin and Begemann 1988, Gordon and Karacapilidis 1997), but it has also been criticized for still being too formal to be attractive for all problems and all user groups (Isenmann and Reuter 1995, Shipman and Marshall 1999).

Some weaker-structured approaches do not support an explicit design process, but offer support for conversations related to an artefact or its representation with the goal of changing or commenting it. In the D3E-Environment of Sumner and Shum (1998), discussion on a document is supported in a way that comments and commented parts of the document always are visualized together, and the “Sticky Chat” concept Churchill et al. (2000) developed for a widely-used text processor supports persistent chats related to paragraphs of the text. Reeves and Shipman (1992) supported artefact-centred communication for computer network design. Mørch and Mehandijev (2000, see also above) also strengthen the necessity of supporting user-designer communication regarding (aspects of) software artefacts. All these approaches focus on the integration of discourse and issue representation, and informed our approach in this respect.

5.2.3 Our case: Tailoring of a Groupware's Awareness Service

Our case is a typical example of a "Shared Tool" scenario. The tailoring task we want to support is the collaborative tailoring of an event notification service in a groupware providing a usual set of functionality like shared workspaces, messaging services and basic workflow support. The "Awareness Service" (Dourish and Bellotti 1992) provides group awareness for members of distributed groups. Fuchs (1998, Fuchs et al. 1996) implemented the concept of an "Awareness Pipeline" as a means to control event flow and event visibility to match the needs for privacy as well as for transparency in collaborative settings (Bellotti and Sellen 1993). In this concept, the group conventions regulating these issues within and between organizational units have to be tailored in a "Global filter" (that complements the "Privacy" and "Interest" filters). The filters decide on event distribution as well as event visualization. While the tailoring of the privacy and the interest filter are individual tasks, the tailoring of the global filter is a collaborative task.

We worked with the groupware application "DEC LinkWorks" used in the collaboration between a federal authority of Germany, and the related authority of one German state. A more detailed description of this setting can be found in (Pipek and Wulf 1999).

This tailoring context is a strong example where it is not appropriate if the users were forced to delegate the design of tailoring alternatives to expert users. Since the configuration of the awareness service is highly important for both, individual privacy and joint collaboration needs, solutions of the tailoring problem should represent agreements negotiated among all group members. As a consequence, the functionality and the problem we described are target of a collaborative design process by their nature, not just because it is implemented in a way that makes collaboration necessary.

5.2.4 Basic design goals for an environment for collaborative tailoring

Our basic guideline for supporting collaborative tailoring was to support tasks and interactions related to the process, not to strongly enforce a pre-modelled design process. From the research background on design processes and our perception of the nature of tailoring activities we drew the following basic design decisions:

Support free articulation: Providing unrestricted means of communication within the design process is necessary to address conflicts and other social issues. We understood design as a targeted discourse process that concludes with a decision for one design alternative.

Discourse-Artefact integration: Tailoring, as “design in context” should not be conducted with separated tools. Our goal was to integrate the support for the tailoring process into the application itself. Additionally, we tried to better integrate the designed artefact into the discourse by supporting quotation of the relevant parts of the groupware.

User-oriented tailoring: Since it is inherently inadequate to expect high levels of dedication or expertise from the users regarding the tailoring activities, we decided to slightly re-design the tailoring mechanisms implemented (e.g. providing menu-guided tailoring, providing additional object-user-relations that users used when speaking about awareness configuration), and to develop appropriate representations of tailoring alternatives (e.g. natural language representations of rules).

Limit support complexity: To limit the perceived complexity of the tailoring support we wanted to offer, we consciously decided against some support issues for design process described in the literature that require formalization. We did not structure the discourse in any way, and we did not implement extensive group decision support functionality.

We developed a discourse-based approach to collaborative tailoring, in which the collaborative tailoring process is embedded in a technically supported discourse among the users of the groupware. New tailoring alternatives as well as their evaluations and further comments are introduced and visualized as discourse statements. The discourse ends with a decision for a tailoring alternative. We describe some details of these principles.

5.2.5 Supporting Discourse

Given our notion of a collaborative tailoring process as a goal-directed discourse process, we believe that it is substantial to improve the usability of the technical support for the discourse. For the discourse process, it is important to provide at least two views on the tailoring activity: a state-oriented view, representing the current state of tailoring alternatives during the tailoring process, and a process-, respectively discourse-oriented view, representing the discourse path which lead to these alternatives. Both of these views should be available and should relate to each other.

In our case this means that actual and proposed conventions of the awareness service should be visible and quotable, as should be the complete discourse around a convention. Since conventions are only valid for the work groups or organizational entities that decided for them, this structure also needs to be visualized in the tailoring tool. A feature we called “discourse awareness” helps users tracing discussions they participated in. E.g. whenever someone comments on a tailoring alternative they proposed or answers to one of their statements, they get notified.

5.2.6 End-User support for tailoring

The existing implementation of the awareness service allowed manipulating the event flow with a predicate language that referred to entities represented in the groupware (documents, containers, organizational entities, workflows, users, etc.) and their properties. We extended the language and provided a new visualization of the predicate rules. We undertook secondary analyses of protocols of interviews and workshops conducted in our application field (regarding different issues of Groupware usage) to see how users relate the groupware entities in their articulation. This work resulted in additional predicates on perceived relations between groupware entities (e.g. “X is owner of document Y”, “In workflow X, Y has worked on the document Z before”). We also developed a natural language representation for the conventions, respectively their realization as rules in a predicate language. This approach was inspired by similar work regarding the representation of access rights in groupware (Stiemerling et al. 1997). This relates to the idea of multiple representations (Mørch and Mehandijev 2000) as well as to the necessity of domain-oriented languages (Fischer and Scharff 2000).

5.2.7 Supporting design activities

Several ideas to ease the management of tailoring activities as a process have been implemented:

Tailoring awareness: Since tailoring usually does not belong to the users’ primary tasks, users cannot be expected to actively check whether there currently is a tailoring process that affects them. Instead, a tailoring awareness service notifies them and points them to the tailoring activity that might affect their work. This could be done e.g. when a tailoring activity would alter configurations a user explicitly agreed to within an earlier tailoring activity.

Another functionality for tailoring awareness is the detection of conflicts of the individual configurations of the privacy and interest filters with the configurations of the global filter can be used to detect the need for (re-) tailoring conventions. When a conflict has been detected, users can choose to start a tailoring process.

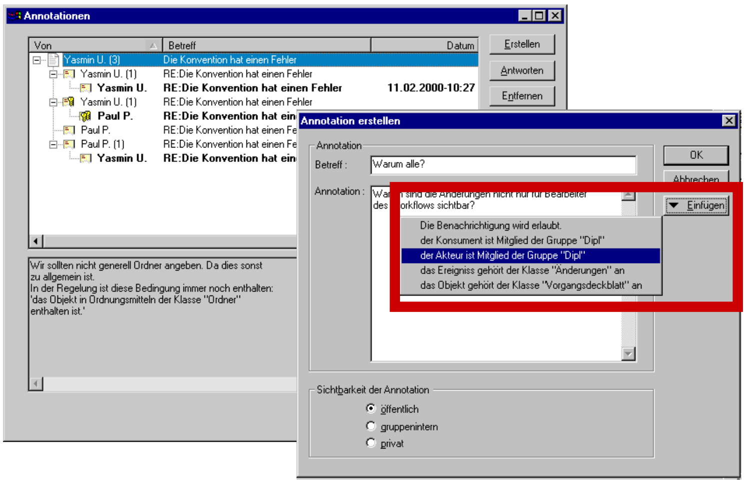

Figure 7. Discourse and Quotation in the Tailoring environment

Cooperative process administration: In deciding to provide means for a user-driven design process, it is also necessary to provide a user-driven process administration. Relevant administrative aspects of the tailoring process are the desired duration of the discourse and the decision procedure. Whether this “process tailoring” should be delegated to some kind of facilitator role, or whether tools should provide means for group decision (e.g. voting support) remained an open question. Although we regard this issue as important, in our case the groupware architecture, especially the inflexible access right model, restricted the implementation of these ideas. Certain functionalities necessary for this “meta-tailoring” required administrator privileges, and those could not be given to all users. Therefore we only implemented support for a privileged facilitator.

5.2.8 Implementation and Evaluation

In the context of the work of Fuchs (1998) an event-based Awareness Service had already been added to the groupware. It allowed the description of rules regarding the filtering of events using a predicate language. This awareness engine prototype was modified and extended. The filtering mechanism now bases on a PROLOG interpreter instead of a proprietary “hard-coded” implementation of the rules, and the user interface was redesigned completely to incorporate the new features. At the user interface, the annotations and the discourse on rules and conventions of the global filter are accessible at the same point where the other filters of the awareness pipeline is accessible (see Uyar 2002 for details). The implementation allows for a design of a conventions (convention proposals) guided by different menus concerning all kinds of groupware entities (folder/document types, user groups, workspaces, workflows, event types, temporal context, etc.). We now describe our environment by a “walk-through” of a tailoring process (see Pipek 2003 for additional details).

While configuring his privacy filter, John is being notified by the conflict detection feature that his adjustments contradict a convention of an organizational task force group on supply chain management he recently joined. He opens the window for the global filter, and navigates through the organizational entities modelled in the groupware to the new task force he just joined. There are two conventions, but one only addresses the workflow functionality. Reading the description of the other convention he finds out that the group inherited this awareness conventions from the configuration of the department of the groups' workspace creator. Their practice is that all events concerning the documents in a shared workspace of the type "strategic issues" should be observable by the whole department. It is being explained that this is appropriate because documents in these workspaces deal with important issues and that this convention is part of the implementation of the company's "Open Information Policy". However, John is used to a practice where not all events are observable, but only document creation and document completion events. He does not feel comfortable with his colleagues being able to observe each and every change he does to his documents. Additionally, since this is a group working across department boundaries, it does not make sense that all members of one department can observe the work while members from different departments can't. John tags the convention and clicks a button to add an annotation.

John writes a comment about his opinion, quoting the natural language representations of the two predicates he wants to be changed (see Fig. 7):

"Allow event forwarding
    if receiver belongs to "Marketing"
and
    if Event Type is anything
and
    …"

Then he opens the window to add a convention proposal and designs the convention, as he would like it to be. Other users may now add annotations to the new thread, quoting from the original or the proposed convention.

Depending on their individual configurations, all members of the task force are being notified that a tailoring process has begun that might affect them (e.g. a message in the status line of the groupware application). They may then open the tailoring environment to add own comments and proposals. Every time someone answers his comment, John is being notified, so that it is not necessary for him to monitor the activities in the tailoring environment by himself.

Every member involved in the discussion can call for a vote on a convention proposal. Depending on the configurations for the work group, the new convention is accepted as soon as an absolute majority, a two-third majority or all members vote for the proposal. If John does not succeed, he may delete his proposal anytime. After the decision for a convention all group members are being notified. This way, all of the tasks we describe in section three have been supported.

In our approach, aspects regarding visibility and accessibility of conventions can also be configured. Due to technical reasons, in our implementation only privileged users or administrators may alter the rights for modifying or deleting a valid convention, and for specifying the details of the decision procedure. For the same reason, the actual implementation of a convention after the groups' decision is also left to a privileged user. However, these aspects are still to be addressed in more detail in the future.

The design of the prototype was informed by about 25 interviews and workplace observations in the application field (e.g. the tailoring language). Due to re-organizations, at the time of the completion of our prototype our application field was not available anymore for a strong evaluation of our concepts in the work setting the prototype was designed for. We could evaluate the prototype only regarding the comprehensiveness of our concepts in a laboratory setting (using the "Heuristic Evaluation" method, Nielsen 1993) although with persons familiar with the original field of application). Some observations are:

- The representation of rules in natural language helps, although it does not significantly reduce the complexity of expressions necessary to achieve a desired result.
- The discourse environment for comments can be used not only strictly related to the design process, but also for more general questions regarding the awareness feature. It can also contribute to a qualification of users.
- The "Quoting" functionality is considered helpful, although it is still problematic to describe the actual problem with a (part of a) convention in words.

In general, the test users were able to orientate in the tailoring environment appropriately.

Of course, this concept and its implementation are just one other experiment in exploring possible solutions to support the continuous collaborative (re-) design of shared software artefacts. We will later describe a notion of opening the discussion of (collaborative) tailoring to support the appropriation of technologies in many different ways.

6 Collaborative Tailoring of and in "Shared Infrastructures"

We now move to areas of the discussion where we still find more challenges than solutions. In the preceding parts of this paper we could draw from a rich variety of studies and concepts that have been published in CSCW research. However, the technological solutions that have been developed share one aspect: a typical tailoring activity is supposed to cover *one* tool or service. But this is not an appropriate perspective on current work environments, today the different services and technologies needed and used to establish computer-supported environments for cooperation can be almost arbitrarily distributed between operating system platforms, middleware applications, groupware applications and single-user-applications. With the emergence of more and more fragmented work environments (Virtual Communities, Virtual Organizations, Freelancer Networks, etc.), and the development of new technologies "beyond the Desktop" (Personal Digital Assistants, Mobile Phones, Wearable Computing, Ubiquitous Computing, etc.) the complexity of the technological *infrastructure* used for intra- and inter-organizational collaboration is likely to further increase (see illustrating cases e.g. in Robertson 1998, Dittrich et al. 2002, Törpel et al. 2003). Regarding the scope of tailoring activities this means that there may be more than one tool or service to tailor to reach a desired state of the technological environment. But, as the example we used for "Shared Infrastructures" in the beginning of this paper illustrates, there are also more and hidden dependencies between different artefacts. We are aware that part of the approaches that we described before can contribute to support for collaborative tailoring for part of the scenario we now look at. But first we should look at some special aspects of the "Shared Infrastructure" scenario.

6.1 The notion of "Inner" and "Outer" Tailorability

Similar to the support concepts, also the general discussion of "tailorability" as a property of an artefact does not reflect the interdependencies with the (technological) context it is used in. It is useful to widen the perspective in the context of our discussion.

The considerations on tailoring described above assumed software artefacts (especially groupware tools) as the technologies to provide support for continuing design in use. In our setting we consider mixed reality environments that may provide a rich variety of hardware and software artefacts, input and output devices. We now try to frame our discussion on tailoring for this kind of settings.

From a user's perspective, all that matters for perceiving an IT artefact as helpful is whether or not the technology provides the use or functionality the user wanted. Maybe it is provided in a straightforward way, maybe the desired behaviour of the technology can be achieved through tailoring it. If this is not possible, there may be another technology that serves the user better. This competition of technologies leads to use environments that contain different technologies or even technology fragments (in terms of only partial use of the functionality available). A new, complex image manipulation software product may be only used to remove the "red eyes" effect off a digital photograph. Maybe adding an annotation to it will be done using a presentation software product, and the resulting file will then be sent out to a relative with an email client – all this in spite of the fact that the image manipulation software initially used would have been able to support all three tasks described. Established technology usages may be only partially abandoned in favour of new technologies. The use of different fragmented technologies has been found in several organizational settings (Robertson 1998, Dittrich et al. 2002, Törpel et al. 2003).

This use pattern lets us derive two kinds of tailorability. The "inner" tailorability addresses the flexibility and ease of manipulation of a tool or device itself to fit different use scenarios. This is the notion of tailorability that was part of most of the work described above. The use scenario with the image manipulation software requires something different that we refer to as "outer" tailorability: A technology has to be prepared for working together with other, even competing technologies to enable users to tailor their work environment by selecting and combining (partial) technologies. Technically technologies can achieve that by referring to a common, shared technological background that is – for desktop applications – usually provided by the operating system (e.g. providing a file system and a directory structure, or support for a software installation procedure), but also comprises standardized protocols (e.g. SQL, OLE), "common" hardware configurations (e.g. PCs being able to display graphics with a certain resolution) and a functionality composition and representation that shows the technology is aware of the fact that it may be used only partially or that it may be even replaced (Pipek and Wulf 1999 show an example for a lack of that awareness). Even more, a technology may have to refer to "softer" standards like usability standards (e.g. interface items like checkboxes or drop-down menus) or established use patterns (e.g. copy-paste-functionality) to achieve "outer" tailorability. Additionally, concepts for the support for collaborative tailoring have to become also aware of "outer" tailorability issues.

6.2 Infrastructure reconsidered

Traditionally, tailoring activities have been seen connected with the organizational change in a work setting. But our notion of infrastructures and technology interdependencies indicates that the complexity of the technological environment may also require tailoring activities to just maintain current work practice in changing technological contexts. This "Retrofitting" has previously been observed in workplace settings on an individual level, e.g. when users tried to re-configure an application after an update to provide the "old" application behaviour (Mackay 1991). But this kind of "Maintenance Tailoring" (how we'd like to call it) is not always directed backwards (as the term "Retrofitting" implies), but also towards the future, e.g. if a user updates an application because collaborating partners have done so, just to maintain an existing practice of document exchange.

In this context, it is helpful to take into account another notion of "infrastructure", beyond just being a set of interconnected tools, technologies and devices. Star and Bowker (2002) describe infrastructure as something "in the background", that "runs underneath other structures", something that we rely on without paying much attention to it. But in the case of a breakdown it comes "into the foreground", we suddenly notice its importance, and the correction of the breakdown may become a comparatively urgent issue. Star and Bowker also stress that there has to be some stability and standardization to allow a perception of an infrastructure as supportive. Some of the approaches we described above relied on this requirement, and were able to support the tailoring of sets of heterogeneous tools, but only if the tools have been developed within a specific framework (e.g. Wang and Haake 2000). However, cases from several virtual organizations showed there are also strong dynamics that work against standardization (Törpel et al. 2003, Karasti and Baker 2004). For current developments in the field of distributed computing, like application service providing, web services and grid computing (Foster and Kesselmann 1999), this observation should be taken into account.

For the field of collaborative tailoring, the hidden dependencies and the possible perceived urgency of detecting and correcting infrastructure breakdowns just to maintain the current work setting pose new challenges that can be illustrated by some questions:

- How do I know what part of my infrastructure did change and caused the breakdown? Who did the re-configuration that was responsible for that? How do I tell that it was someone else's tailoring activity, not an accident (e.g. power failure) that caused the breakdown? How can I react without negotiation in case the reconstitution of my infrastructure is urgent?
- How do I know who will be influenced by my tailoring activity? How do I negotiate about possible re-configurations of my tools/devices?

One possible way to answer these questions would be to use metadata on the applications and devices that reflect its capabilities regarding inner and outer tailorability. Once created, the resulting system of representations may be the foundation for new approaches to support collaborative tailoring.

6.3 Potentials for supporting collaborative tailoring

One of the key features of the approaches for collaborative tailoring we described in the preceding sections is the use of representations of applications and tailoring objectifications as a basis for computer-supported communication, distribution, discussion and negotiation processes. It was possible to develop and describe concrete approaches because the scope of developing the tailoring support was within *one* application, and so it was within the scope of the development of the application itself. The developer of the application had complete control over the functionality and the possible tailoring activities. Now, looking at tailoring "beyond one tool", the scope of possible application-relevant (and use-relevant) tailoring activities goes beyond the scope of development of the application itself. This situation still carries aspects of the "Shared Context" and "Shared Tool" scenarios, but a "targeted" design of support for collaborative tailoring is not possible here. But we can discuss the issue along these general notion of application representations on the one hand and communication and collaboration facilities on the other.

There are obvious and simple ways to create representations of applications and tailoring objectifications using the screen capturing features of modern operating systems to produce screenshots. However, depending on the interface design these can be more or less useless to describe tailoring issues, and some aspects of an application can't be represented accordingly (e.g. its architecture or menu structure). The use of video and/or animation would illustrate problems better (Baecker 2002), but the application itself may also provide more sophisticated representations of itself. It would be possible to draw on representations that have been made during application development (similar to those described by Mørch and Mehandijev 2000, or in the context of "design rationale" systems in Moran and Carroll 1996). Dourish (1997) discussed the idea of providing "computational representations" of the behaviour of application components for users at the interface as well as for other components to determine current activities of a component. Reaching the goal to enable every application (component) to exactly represent its "area of accountability" (e.g. a text processor is responsible for the spell check, but not for problems with the file system) would actually also help to manifest and visualize structure in "Shared Infrastructure" scenarios.

For communication and collaboration, levels of technology have to be used that are accessible for all the actors involved. As in the concepts developed by Kahler (2000), this is likely to be the traditional E-Mail infrastructure that can be found at almost every modern workplace. Another interesting idea would be the establishment of "Online Future Workshops" (following Jungk and Müllert 1996) for the communication regarding the setup of a shared infrastructure, where a web discussion forum could be used as a base technology.

7 From Collaborative Tailoring to Appropriation Support

For the final part of our discussion we want to take a step back and look at the broader picture. Several long-term studies (Mackay 1990, MacLean et al. 1990, Karsten and Jones 1998, Pipek and Wulf 1999, Törpel et al. 2003) pointed out that the technologies used in organizations are tailored again and again to match the changing user needs. Through the processes of familiarization, (re-) configuration and usage the technology is appropriated, that means it is being transformed and specialized from the more or less abstract notions of usage the technology designers once imagined to be possible with it, to the concrete interests, meanings and purposes of (a group of) users. In this process of appropriation, we see tailoring as the key activity, since every tailoring activity is in fact (at least partially) a re-discovery and re-invention of the practices that are possible with this technology. Exploring the collaborative dimensions of tailoring is – in our eyes - also a starting point for exploring the collaborative dimensions of technology appropriation, where the main point is to not only understand a technology and its affordances, but to also be able to develop and negotiate alternative scenarios in a group or organization. The provision of appropriate tailoring environments and technologies of flexibilisation (e.g. component-based systems) is a technological prerequisite of appropriation processes, but we now turn from a perspective of more or less "targeted" tailoring activities to build an appropriate collaborative environment to "softer" tasks like getting new ideas for a better practice, learning about technological options and possible usages, and discussing and deciding on alternative scenarios (with implementation being only a secondary interest). Can't we do more in technology design to support group appropriation processes?

Robinson (1993) described the "common artefact" and the characteristics that afford its successful appropriation in order to inform the design of collaborative software. For supporting the appropriation of technology in groups and organizations, the concept of "double level language" is particularly valuable. It distinguishes between two mutually supportive modalities of communication: "implicit communication" in which the artefact (e.g. a distributed spreadsheet, "Flight progress strips" used by air traffic controllers, a key rack at a hotel reception) is a medium for communication (by transmitting the current state of work activities or work results), and "explicit communication" (e.g. speech, ad-hoc notes), in which the artefact is providing representations of work-related issues that can be referred to. This suggests an understanding in which a collaborative technology or tool is as well a medium for implicit communication as something that provides reference points for explicit work-related communication. But in addition to that, the discussions of collaborative tailoring above show that a collaborative technology can also provide the means for explicit communication, can prepare to be the subject of explicit communication (being a "work-related issue") and can support implicit communication regarding technology usage.

We pick up some aspects of the experiences discussed before to show that there is more to think about along this line of thought. We do that together with developing our notion of "appropriation activities". We try to distinguish three purposes of activities within appropriation processes, and discuss related issues and concepts from the experiences we described.

7.1 Understanding technologies

By understanding we mean the discovery and aggregation of knowledge on the basic principles and scopes of technologies ("This software is a text processor.", "Before you can use software, you have to switch the computer on.", "If you can't save your text on a floppy disk, it may be write-protected, it may be not formatted, your operating system is maybe occupied with doing something else, but it is not the word processor that is causing the trouble.", etc.). This kind of knowledge is especially important for the case of breakdowns in the infrastructure. It is to some extent a prerequisite of technology usage, but it is also very general knowledge. Dourish (1997) pointed out one possible direction of technological improvements to support "understanding". He discussed the intransparency of current technological infrastructures and proposed to use behavioural descriptions of a technology (and its subparts) as a complement of current architectural descriptions. The general goal is to enable technologies to "understand" the behaviour of other technologies to be able to orientate in a rich infrastructural setting. As a consequence they would also be able to deliver a more precise picture of their capabilities and limitations to the user.

Regarding collaborative aspects, we see this kind of knowledge being gained mostly through "legitimate peripheral participation" in communities of people who have and use this technology (cf. Wenger 1998). Because of the weak shared interests of users, we consider similar to the situation in "Shared Use" scenarios, and see similar (limited) opportunities for technological support.

7.2 Learning about a technology

With learning we want to refer to the acquisition of knowledge necessary to use a technology according to the purposes that the technology designers embedded into it. It means to learn about the menu structure and other navigation instruments, and the functionality the technology offers. It also refers to learning about the means to modify and tailor the technology.

Currently, small tutorials, help systems and "application wizards" support users in learning about a software artefact together with non-computer-based means (books, courses, etc.). However, there could be even more and better representations of technologies using multimedia technology[4] and embedding it into the artefacts themselves. Another idea would be to offer more different

---

[4] It is interesting to observe that even in "Open Source" communities there are videos for explaining and training tool usage (for example for the Content Management System "Typo3": http://typo3.org/1407.0.html).

representations of the artefact (e.g. behavioural descriptions). But, since learning as a "side effect" of collaborative tailoring has been mentioned before (Wulf and Golombek 2001, Pipek 2003), we again regard the direct support of tool-related communication as a key feature; and we could well imagine using concepts and experiences from the field of Computer-Supported Collaborative Learning to further improve current tailoring environments to better support appropriation.

Another important idea for a more active learning process was presented by Wulf and Golombek (2001). They addressed the problem that the consequences of tailoring activities are often not visible to the tailor. They suggested integrating "exploration environments" into groupware tools that allow users to play with the tailoring interface of a tool without actually changing something in their organizational context. These exploration environments also presented the results of the exploratory tailoring activities from the perspective of the users affected by it. Again, we could well imagine additional benefit for appropriation processes if there would also be collaborative exploration environments where users share the experience of exploring the possible configurations of the technology they use.

7.3 Sensemaking of technologies

Maybe the most important step of appropriation happens when users start to answer the question: What can this technology do *for me (us)*? Usually this involves considering current abilities, tasks, technologies and their usages, but also the perceived value of the new technology at stake.

Several studies showed (Mackay 1990, Pipek and Wulf 1999) how a new technology "diffused" into an organization. In a department of a federal authority with approximately 25 staff members, the introduction of a groupware application took about 18 months from the first user working with it until the last computer was installed. Users discovered the value the new technology might have for them by observing other users in the same organizational context (Pipek and Wulf 1999). Making usages of a technology better observable, maybe within the technology itself, could support this mechanism. Linton (2003) described a system for sharing expertise on using a word processor. He tried to record and associate the "use traces" (consecutive interface actions) of different users to give recommendations on tool usage. This could also be interesting for just visualizing the practice of other users. Complemented with additional communication facilities this might help a new user to assess the perceived value of a technology for other users in similar use scenarios. With the similar intent to allow users to "leave their traces" by enhancing their information spaces, Dourish (2003) described a system that allowed flexible end-user extension of document metadata ("properties") as well as attaching dynamic behaviour through "active properties". Dittrich (1998) suggested to explicate the scenarios and assumptions that the designers had when developing the software in order to enable the users to understand what the intended use of the software was and what parts of these assumptions and scenarios apply to the current use situation the user is in. Similarly to Mørch and Mehandijev (2000) the idea relates to a designer-user dialogue for sensemaking, but can easily be expanded to support the user-user dialogue for sensemaking that we consider crucial to support technology appropriation in organizations.

Reconsidering the "exploration environments" (Wulf and Golombek 2001) mentioned above, we could also imagine using these environments to demonstrate alternative use scenarios or technologies to a group of users. As a "test simulation" they can be especially valuable if it is possible to map aspects of the real existing work setting of the potential users.

7.4 Building "Communities of Technology Practice"

Concluding the discussion above, we want to develop a notion of technical support for "Communities of technology practice" (following the concept of "Community of practice" of Wenger (1998)). The ideas of understanding, learning and observing technology-in-use can be consolidated in a concept of "inhabited technology", where environments for collaborative tailoring are extended to give users a permanent presence within the technology, making the "tool" also a "place". It is still an open question, for which (levels of) technologies this is possible and appropriate (see discussion on "Shared Infrastructures" above).

Again, some of the approaches we described above focus on a perspective of supporting "user-user" communication, which we consider most important for appropriation support. Törpel et al. (2003) and Hansson et al. (2003) show example settings in which that communication emerged, and changed technology design and usage. The concepts and experiences developed for virtual communities, especially regarding communication and the visualization of users and issues, can be exploited to further enhance tailoring environments in that line of thought. Twidale and Nichols (1998) showed an example for this support regarding the definition of successful search queries for a library database system, although because of the lack of persistence (of communication partners as well as "tailoring objects") in that solution their system follows neither the notion of "tailoring" nor the notion of "community" we would like to establish here. But in their system design as well as their evaluation they also stress the importance of adequate representations (which are, in their case, visualizations of a query and the related search process of the database system) of the system's behaviour not only for understanding, but also for communicating about a technology. Consequently, they also offer an embedded support for user-user communication.

That perspective above should be complemented by experiences and concepts published regarding the improvement of user-designer communication. Getting back on the ideas of Fischer and Scharff (2000) on the alternating phases of proliferation and consolidation (the "Seeding-Evolutionary Growth-Reseeding"-

Model), a concept that Törpel et al. (2003) also encountered in practice, and of Mørch and Mehandijev (2000) on the need for a solid long-term collaboration between technology designers, tailors and users, it would be also feasible to incorporate the designers into this community. Hansson et al. (2003) showed an example of combining methods from Participatory Design and Agile Software Development in order to form a (non-virtual) "Community of Technology Practice" of software providers and users. This would also change the context of tool development, and complement the "technology push" of the designers with the "demand pull" of the users. This is common practice in many Open Source Projects, where users can request new features of software artefacts (e.g. the "requested features" on http://sourceforge.net/), and has also be reported in commercial contexts (Dittrich et al. 2002, Hansson et al. 2003). In that concept, the domains of traditional software development, tailoring and Participatory Design, could find a shared manifestation in technologies and technology development processes (see also Dittrich et al. 2002).

8 Conclusion

In this paper we tried to capture the collaborative dimension of configuration activities, and explored opportunities for technological support of collaboration in tailoring activities. We distinguished four scenarios ("Shared Use", "Shared Context", "Shared Tool" and "Shared Infrastructure") with different intensities of ties between the potentially collaborating users, and described existing approaches as well as open research issues.

The general line of argumentation in the approaches was – in line with our own beliefs – that every technology should be aware of the fact that it is subject to communication and negotiation. The approaches tried to cover this requirement not only by offering the necessary granularity and flexibility to allow finely differentiated alternatives, but also by providing representations that are helpful in user-user communication and negotiation. Some also integrated communication facilities into tailoring interfaces.

We pointed out, that especially "Shared Infrastructure" scenarios with their heterogeneous, intertwined and interdependent mixes of technologies, still pose a major challenge to research on tailorable systems.

In the final part of our discussion we suggested to take these approaches to collaborative tailoring as a starting point for a broader technological support of processes of technology appropriation. We suggested improving the safe explorability of technologies and the visualizations of technology usage. But we consider it most important to further improve the means of communicating and negotiating (especially user-user contact, but also user-designer contact) on the basis of these new ways to represent technology structures and use.

9 References

Baecker, R. (2002): Showing instead of Telling. In: ACM SIGDOC'02, (Toronto, On, Canada, 2002), ACM Press. pp. 10-16.

Bellotti, V. and Sellen, A. (1993): Design for Privacy in Ubiquitous Computing Environments. in 3rd European Conf. on CSCW (ECSCW'93), (Milan, Italy, 1993), Kluwer, pp. 77-92.

Carter, K.; Henderson, A. (1990): Tailoring Culture. In: Reports on Computer Science and Mathematics no. 107, Åbo Akademi University 1990. Proceedings of 13th IRIS. pp. 103-116.

Churchill, E., Trevor, J., Nelson, L., Bly, S. and Cubranic, D. (2000): Anchored Conversations: Chatting in the Context of a Document. In: CHI'2000: The future is here, (The Hague, Netherlands, 2000), ACM Press, pp. 454-461.

Conklin, J. and Begemann, M.L. (1988): gIBIS: A Hypertext Tool for Exploratory Policy Discussion. In: Conference on Computer Supported Cooperative Work, (Portland, Oregon, USA, 1988), ACM, pp. 140-152.

Dellen, B. and Maurer, F. (1996): Integrating Planning and Execution in Software Development Processes. In: Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET-ICE), (Stanford, 1996)

Dittrich, Y. (1998): How to make Sense of Software - Interpretability as an Issue for Design, Dep. of Computer Science and Business Administration, University of Karlskrona/Ronneby, TR 98/19, Ronneby, Sweden, 9 p.

Dittrich, Y., Eriksen, S. and Hansson, C. (2002): PD in the Wild; Evolving Practices of Design in Use. in Participatory Design Conference, (Malmö, Sweden, 2002), CPSR, pp. 124-134.

Divitini, M., Jaccheri, L., Monteiro, E., Trætteberg, H. (2003): Open source processes: no place for politics? Proceedings of ICSE 2003 workshop on Open source, pp. 39-44

Dourish, P. (1997): Accounting for System Behaviour: Representation, Reflection and Resourceful Action. In: Kyng, M. and Mathiassen, L. (eds.): Computers and Design in Context, MIT Press, Cambridge, MA, USA, pp. 145-170.

Dourish, P. (1999): Software Infrastructures. In: Beaudouin-Lafon, M. (ed.): Computer Supported Co-operative Work, John Wiley & Sons, pp. 195-219.

Dourish, P. and Bellotti, V. (1992): Awareness and Coordination in Shared Workspaces. In: ACM Conf. on Computer Supported Cooperative Work (CSCW'92), (Toronto, Canada, 1992), ACM Press, pp. 107-114.

Dourish, P. and Edwards, W.K. (2000): A tale of two toolkits: Relating Infrastructure and Use in Flexible CSCW Toolkits. In: Computer-Supported Cooperative Work (CSCW), 9 (1). pp. 33-51.

Dourish, P. (2003): The Appropriation of Interactive Technologies: Some Lessons from Placeless Documents. Computer Supported Cooperative Work (CSCW) - The Journal of Collaborative Computing, 12 (4). pp. 465-490.

Foster, I. and Kesselmann, C. (eds.) (1999): The Grid: Blueprint for a new Computing Infrastructure. Morgan Kaufmann Publ. Inc., San Francisco, CA, USA.

Fischer, G., Lemke, A.C., McCall, R. and Mørch, A. (1996): Making Argumentation Serve Design. In: Moran, T.P. and Carroll, J.M. eds. Design Rationale - Concepts, Technology and Use, Lawrence Erlbaum Ass., Mahwah, NJ, USA, pp. 267-294.

Fischer, G. and Scharff, E. (2000): Meta-Design: Design for Designers. In: Int. Conf. on Designing Interactive Systems (DIS'00), (Brooklyn, New York, USA, 2000), ACM Press, pp. 396-405.

Fischer, G. (2002): Beyond 'Couch Potatoes': From Consumers to Designers and Active Contributors", Available at http://firstmonday.org/issues/issue7_12/fischer/, 2002. FirstMonday (Peer-Reviewed Journal on the Internet), 7 (12).

Fischer, G. and Ostwald, J. (2002): Seeding, Evolutionary Growth, and Reseeding: Enriching Participatory Design with Informed Participation. In: Participatory Design Conference, (Malmö, Sweden, 2002), CPSR, pp. 135-143.

Fischer, G. and Giaccardi, E. (2005): Meta-Design: A Framework for the Future of End User Development. In: Lieberman, H., Paternó, F. and Wulf, V. (eds.). End User Development. Springer, Berlin.

Fuchs, L., Sohlenkamp, M., Genau, A., Kahler, H., Pfeifer, A. and Wulf, V. (1996): Transparenz in kooperativen Prozessen: Der Ereignisdienst in POLITeam, ("Transparency in cooperative processes: The event notification service in POLITeam"). In: "Herausforderung Telekooperation", DCSCW'96, (Stuttgart, 1996), Springer.

Fuchs, L. (1998): Situationsorientierte Unterstützung von Gruppenwahrnehmung in CSCW-Systemen, („Situated Support for Group Awareness in CSCW-Systems“). PhD Thesis, FB Mathematik und Informatik, Uni-GHS Essen, Germany.

Gantt, M.; Nardi, B.A. (1992): Gardeners and Gurus: Patterns of Cooperation among CAD Users. In: Proceedings of CHI ‘92. pp. 107-117.

Garg, P.K. and Jazayeri, M. (eds.) (1996): Process-Centered Software Engineering Environments. IEEE Press, Los Alamitos, USA.

Gordon, T.F. and Karacapilidis, N. (1997): The ZENO Argumentation Framework. In: Int. Conf. on Artificial Intelligence and Law (ICAIL'97), (Melbourne, Australia, 1997).

Hansson, C., Dittrich, Y. and Randall, D. (2003): "The development is driven by our users, not by ourselves" - Including Users in the Development of Off-The-Shelf Software. In: 26th Information Systems Research Seminar in Scandinavia (IRIS 26), (Haikko Manor, Finland, 2003), IRIS Association.

Henderson, A. and Kyng, M. (1991): There's no place like home: Continuing Design in Use. In: Greenbaum, J. and Kyng, M. eds. Design at work: Cooperative Design of Computer Systems, Lawrence Erlbaum Ass., Hillsdale, NJ, pp. 219-240.

Isenmann, S. and Reuter, W.D. (1997): IBIS - a convincing concept ... But a lousy instrument. in Designing Interactive Systems: Processes, Practices, Methods and Techniques, (Amsterdam, The Netherlands, 1997), ACM Press, pp. 163-173.

Jungk, R. and Müllert, N.R. (1996): Future Workshops: How to Create Desirable Futures. Inst. for Social Inventions, UK.

Kahler, H. (2000): Constructive Interaction and Collaborative Work: Introducing a Method for Testing Collaborative Systems. In: acm interactions, Vol. VII (3) (May/June 2000). pp. 27-34.

Kahler, H. (2001a): More Than WORDs - Collaborative Tailoring of a Word Processor. In: Journal of Universal Computer Science (j.ucs), Vol. 7 (9). pp. 826-847.

Kahler, H. (2001b): Supporting Collaborative Tailoring. Department of Communication, Journalism and Computer Science, Roskilde University, Roskilde.

Karasti, H. and Baker, K.S. (2004): Infrastructuring for the Long-Term: Ecological Information Management. In: 37th Hawaii International Conference on System Sciences (HICSS 2004), http://csdl.computer.org/comp/proceedings/hicss/2004/2056/01/205610020c.pdf. (9.3.2004)

Karsten, H. and Jones, M. (1998): The long and winding road: Collaborative IT and organisational change. In: Int. Conference on Computer Supported Work (CSCW'98), (Seattle, WA, USA, 1998), ACM Press, pp. 29-38.

Linton, F. (2003): OWL: A system for the automated Sharing of Expertise. In: Ackerman, M.S., Pipek, V. and Wulf, V. eds. Sharing Expertise: Beyond Knowledge Management, MIT Press, Cambridge, MA, USA, pp. 383-401.

Mackay, W.E. (1990): Patterns of Sharing Customizable Software. In: Proceedings of CSCW ´90. pp. 209-221.

Mackay, W.E. (1991): Triggers and Barriers to Customizing Software. In: Proceedings of CHI ´91. pp. 153-160.

MacLean, A.; Carter, K.; Lövstrand, L.; Moran, T. (1990): User-Tailorable Systems: Pressing the Issues with Buttons. In: Proceedings of CHI 90. pp. 175-182.

Malone, T.W.; Grant, K.R.; Lai, K.-Y.; Rao, R.; Rosenblitt, D. (1988): Semistructured Messages are Surprisingly Useful for Computer-Supported Coordination. In: Proceedings of CSCW 88. Morgan-Kaufmann Publishers. pp. 311-334.

Malone, T.W., Lai, K.-Y. and Fry, C. (1992): Experiments with Oval: A Radically Tailorable Tool for Cooperative Work. In: Int. Conference on CSCW (CSCW'92), (Toronto, Canada, 1992), ACM Press, pp. 289-297.

Moran, T.P. and Carroll, J.M. (eds.) (1996): Design Rationale: Concepts, Techniques and Use. Lawrence Erlbaum Assoc., Mahwah, NJ, USA.

Mørch, A. (1997): Three Levels of End-user Tailoring: Customization, Integration, and Extension. in Kyng, M. and Mathiassen, L. (eds.): Computers and Design in Context, MIT Press, Cambridge, MA, USA, pp. 51-76.

Mørch, A. and Mehandjiev, N. (2000): Tailoring as Collaboration: Mediated by Multiple Representations and Application Units. In: Computer Supported Cooperative Work: The Journal of Collaborative Computing, Special issue on “Tailorable Systems and Cooperative Work”, 9 (1). pp. 75-100.

Nardi, B.M. (1993): A Small Matter of Programming. Cambridge, Massachusetts, MIT Press.

Nardi, B.A.; Miller, J.R. (1991): Twinkling lights and nested loops: distributed problem solving and spreadsheet development. In: Int. J. Man-Machine Studies, Vol. 34. pp. 161-184.

Nielsen, J. (1993): Usability Engineering. Academic Press, Boston, MA, USA.

Oberquelle, H. (1994): Situationsbedingte und benutzerorientierte Anpaßbarkeit von Groupware. (“Situation-dependent and user-oriented Tailorablilty of Groupware”). In: Hartmann, A., Herrmann, T., Rohde, M. and Wulf, V. (eds.): Menschengerechte Groupware - Software-ergonomische Gestaltung und partizipative Umsetzung, Teubner, Stuttgart, pp. 31-50.

Okamura, K., Fujimoto, M., Orlikowski, W.J. and Yates, J. (1994): Helping CSCW Applications succeed: The role of Mediators in the context of Use. In: Int. Conf. on CSCW, (1994), ACM Press, pp. 55-65.

Pankoke, U.; Syri, A. (1997): Collaborative Workspaces for Time deferred Electronic Cooperation. In: Proceedings of GROUP ´97. pp. 187-196.

Pipek, V.(2003): An Integrated Design Environment for Collaborative Tailoring. In: ACIS Int. Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD'03), (Lübeck, Germany, 2003), ACIS, 430-438.

Pipek, V. and Wulf, V. (1999): A Groupware's Life. in European Conference on Computer Supported Cooperative Work (ECSCW'99), (Copenhagen, Denmark, 1999), Kluwer, Dordrecht, Netherlands, pp. 199-218.

Reeves, B. and Shipman, F.M.I. (1992): Supporting Communication between Designers with Artifact-Centered Evolving Information Spaces. In: Int. Conference on CSCW (CSCW'92), (1992), ACM Press, pp. 394-401.

Rittel, H.W.J. (1973): On the Planning Crisis: Systems Analysis of the First and the Second Generation. bedriftsokonomen (8). pp. 390-396.

Robertson, T. (1998): Shoppers and Tailors: Participative Practices in Small Australian Design Companies. In: Computer Supported Cooperative Work (CSCW), 7 (3-4), 1998. pp. 205-221.

Robinson, M. (1993): Design for unanticipated use..., In: European Conference on CSCW (ECSCW'93), (Milan, Italy, 1993), Kluwer, Dordrecht, NL, pp. 187-202.

Shipman, F.M.I. and Marshall, C.C. (1999): Formality Considered Harmful: Experiences, Emerging Themes, and Directions on the Use of Formal Representations in Interactive Systems. Journal on Computer Supported Cooperative Work, 8. pp. 333-352.

Sommerville, I. (2000): Software Engineering. Addison Wesley.

Stallman, R. (1981): EMACS: The Extensible, Customizable, Self-Documenting Display Editor. In: Proc. of the ACM SIGPLAN SIGOA, (1981), Mass. Institute of Technology, pp. 301-323.

Star, S.L. and Bowker, G.C. (2002): How to infrastructure. In: Lievrouw, L.A. and Livingstone, S. (eds.) Handbook of New Media - Social Shaping and Consequences of ICTs, SAGE Pub., London, UK, pp. 151-162.

Stevens, G. and Wulf, V. (2002): A new dimension in access control: Studying maintenance engineering across organizational boundaries. In: Int. Conference on CSCW, (New Orleans, 2002), ACM Press, pp. 196-205.

Stiemerling, O. and Cremers, A.B. (2000): The EVOLVE Project: Component-Based Tailorability for CSCW Applications. AI & Society, 14. 120-141.

Stiemerling, O., Kahler, H. and Wulf, V. (1997): How to Make Software Softer - Designing Tailorable Applications. In: DIS '97, (Amsterdam, 1997), ACM Press, pp. 365-376.

Sumner, T. and Buckingham Shum, S. (1998): From Documents to Discourse: Shifting Conceptions of Scholarly Publishing. in CHI 1998: Human Factors in Computing Systems, (Los Angeles, CA, USA, 1998), ACM Press, New York, pp. 95-102.

Törpel, B., Pipek, V. and Rittenbruch, M. (2003): Creating Heterogeneity - Evolving Use of Groupware in a Network of Freelancers. Special Issue of the Int. Journal on CSCW on "Evolving Use of Groupware", 12 (4). pp. 381-409.

Trigg, R., Moran, T.P. and Halasz, F.G. (1987): Adaptability and Tailorability in NoteCards. In: INTERACT'87, (Stuttgart, Germany, 1987), pp. 723-728.

Trigg, R.; Bødker, S. (1994): From Implementation to Design: Tailoring and the Emergence of Systematization in CSCW. In: Proceedings of CSCW ‘94. pp. 45-54.

Twidale, M. and Nichols, D. (1998): Designing interfaces to support collaboration in information retrieval. Interacting with computers, Vol. 10. pp. 177-193.

Uyar, S. (2002): Unterstützung kollaborativen Anpassens in Groupwareanwendungen am Beispiel der Konfiguration eines Awarenessdienstes. (“Support of collaborative Tailoring in Groupware applications: The example of the configuration of an Awareness Service”), Master Thesis (in German), Institut für Informatik, Universität Bonn, Bonn.

Wang, W. and Haake, J.M. (2000): Tailoring Groupware: The Cooperative Hypermedia Approach. International Journal of Computer-Supported Cooperative Work, 9 (1), 2000.

Wasserschaff, M.; Bentley, R. (1997): Supporting Cooperation through Customisation: The Tviews Approach. In: Computer Supported Cooperative Work: The Journal of Collaborative Computing (JCSCW), Vol. 6. pp. 305-325.

Wenger, E. (1998): Communities of Practice - Learning, Meaning and Identity. Cambridge University Press, Cambridge.

Wulf, V. (1997): Konfliktmanagement bei Groupware (“Conflict Management in Groupware applications”). Vieweg, Braunschweig.

Wulf, V. (1999): Evolving Cooperation when Introducing Groupware-A Self-Organization Perspective. In: Cybernetics and Human Knowing, Vol. 6 (2). pp. 55-75.

Wulf, V. (1999): "Let's see your Search-Tool!" - On the Collaborative use of Tailored Artifacts. In: Proceedings of GROUP ’99, ACM-Press, New York, pp. 50 - 60

Wulf, V. and Golombek, B. (2001): Exploration environments: concept and empirical evaluation. In: International ACM SIGGROUP Conference on Supporting Group Work, (Boulder, Colorado, USA, 2001), ACM Press, pp. 107-116.

Wulf, V., Pipek, V. and Pfeifer, A. (2001): Resolving function-based conflicts in groupware systems. AI & Society, 15. 233-262.