Conceptualization and Appropriation: The Evolving Use of a Collaborative Knowledge Management System

Claus Bossen & Peter Dalsgaard
Information and Media Studies
University of Aarhus, Denmark
bossen@imv.au.dk; pdal@imv.au.dk

ABSTRACT
Zephyr is an expanding software company that developed a knowledge management system designed to support the work of employees and provide management overview. Despite strong management support the system was not much used and instead employees themselves developed a competing and much used parasitic system. First, we argue that the failure of the management's system is caused by the concept of knowledge upon which the system was built. Hence, design of computer systems is as much a question of critical conceptual understanding of its application domain as a question of doing ethnography and system development. Second, we argue that the process of design extends far into the process of use and that much can be learned by looking at the process of appropriation of a new system. The problems of conceptualisation and appropriation point towards the need to critically examine the mangle of practice in which artefacts, actors and organizations intertwine.

Keywords
Appropriation, conceptualisation, design, knowledge management, participatory design, system development, user involvement, tailorability.

INTRODUCTION
"Free your mind!" Neo says to himself in the movie *The Matrix* and tries to jump from the roof of one skyscraper to the next while being in virtual space. He cannot let go of his conceptualizations of the world, however, and hits the (virtual) concrete street. Having returned to the real world a bit later he has to struggle with the autonomous technological artefacts in the shape of octopus-like robots together with the crew of the Nebuchadnezzar.

It is this tension and interaction between conceptualizing the world and material technological artefacts that we want to explore in this paper. We base our argument on the case on the corporation of Zephyr (a fictional name) that invested heavily in a knowledge management system in order to provide support for its employees at a time when the corporation had expanded from 150 to 300 employees within a few years. However, after 4 years of development and considerable investments, the Zephyr management had to accept defeat of their knowledge management system, which was not used. Instead they adopted a parasitic system that had been developed in its place.

In the case of Zephyr, a crucial question was the conceptualization of knowledge and how this was incorporated into the knowledge management system. The system was in part based on studies of the use domain and the involvement of users, but still knowledge was conceptualized in a way that did not fit the work practices of the employees. Another crucial issue was the ability of employees to adapt and redesign system components and ultimately develop the parasitic system that did support their work. The case of adoption and adaptation of the knowledge management system of Zephyr thus points towards the importance of critical conceptualizations, competent actors and malleable technology for an agreeable relationship between actors and technology.

BACKGROUND
The dynamic interaction of actors and technology in which technology enables and restrains action and actors delegate competencies to technology has been described by various authors [11, 22, 30]. Actors design technologies based on their understanding of the field of use and the intentional changes they want to achieve, and artefacts in turn have structural features that enable some forms of action more than others. New actions may form the basis for new conceptualizations and artefacts, and a continuous evolving exchange of conceptualization, action and artefact is going on: "…technologies comprise the objectification of knowledges and practices in new material forms" [37: p1]. Actors may, of course, use technologies in unintended ways and contrary to design intentions, but it is nevertheless easier to cut meat with a knife than with a spoon.

This dynamic interaction has led, on the one hand, to the development of a number of design practices in order to tie into the work practices in which the envisioned artefact will be used – Participatory design [16, 35], user-centred design etc. – and, on the other hand, a number of studies of how actors adopt new artefacts [9, 27, 34].

While we are definitely in favour of engagement with the actors and practices for which artefacts are made, we also like to point to the importance of sensible concepts and

AARHUS'05 8/21-8/25/05 Århus, Denmark

appropriable technology, i.e. technology that can be decomposed and deconstructed, in furthering critical computing.

Knowledge Management

From the middle of the 1990s, corporate firms turned to the notion of 'knowledge management' in attempts to gain an edge in marketplaces with ever-increasing competition (For a critical overview, see [46]). Amongst the aims were the wish to secure critical knowledge in cases where employees left the corporation by having this knowledge shared by several people or stored somewhere in corporation; a wish to accelerate innovation and creativity and to use better the allegedly inactive or under-used knowledge resources of employees building on the statement by Michel Polanyi [32] "we know more than we can tell".

The field of knowledge management can be divided into two tracks: an IT track and a people track [42]. The IT track tends to regard knowledge as an object that can be captured, transmitted and handled by information systems. According to Fahay and Prusak [14], organizations wanting to promote 'knowledge management' often have inadequate conceptualisations of knowledge, which they equate with information. In this line of thinking knowledge management is rather information management. The people track, on the other hand, emphasizes the central role of individuals in the process of knowledge creation and sharing. Knowledge is seen as the human faculty of sense-making (e.g. [24, 43]. Parallel distinctions see organizational memory as either storable and retrievable or as enacted by people [5], as either object or as process [2]

There is no agreement upon how to define knowledge, but a common way of making distinctions is to distinguish between three forms (See e.g. [26]): First, *tacit knowledge* (or *procedural knowledge)* I; s what actors rely on while acting, but which cannot be made explicit, e.g. face recognition. Second, *implicit knowledge* is knowledge that is not, but can be articulated and made explicit, e.g. implicit categorizations or unarticulated aspects of work practice. Third, *explicit knowledge* is that which can be articulated, also often called *declarative knowledge*. Explicit knowledge may be expressed in spoken language or written down.

According to one position, mainly found in the people track of the knowledge management field, knowledge is a state that only exists in the individual who has made meaning of something external to her/him, i.e. information [24, 46]. "*Information*, it turns out, is simply the vehicle by which we attempt to provoke - or evoke - a human response. Information on its own is quite static and lifeless. It simply exists - on multimedia computer screens, in text books, magazines, movies, TV, CDs, reports, letters, emails, faxes, memos and so on - all waiting to be interpreted, all waiting to have meaning attached - by people." [24]. Following this understanding of knowledge means that most of what commonly is labelled 'knowledge management' is really about 'information management'.

Making sense of information (explicit knowledge) requires that the recipient of information can connect his or her horizon of meaning to the information (i.e. a hermeneutical circle), which can be achieved either through ongoing dialogue with the sender of information (e.g. when we make bullet points understandable during presentations), or by bringing sufficient contextual information into the message to enable the sense-making process (which is why we write articles and not just present our knowledge in bullet points). Knowledge sharing can thus be furthered by supporting dialogue between actors or by producing sufficiently contextualised information. Finding the right granularity of contextualisation may, however, be difficult [1, 3],

Even within a meaning-oriented view of knowledge as above [24], IT can play an important role, since tacit, implicit and explicit knowledge can be transmitted from one actor to another different ways: writing, visual representations, video and sound can all be 'vehicles' that evoke meaning making in different and more or less effective ways. Knowledge creating and sharing may happen within or between people, but IT mediated interaction may still further such processes (e.g. [20]). However, if a system excludes e.g. visual information, knowledge sharing may not be possible (e.g. [31]).

As we will show, the difference between an IT and a people approach was crucial in the case of Zephyr, as was the conceptualization of knowledge work. Knowledge was equated with implicit and explicit knowledge (e.g. 'information' in a meaning-oriented approach) which could be explicated and stored.

Appropriation

In a certain sense, knowledge can be said to become objectified as embedded in artefacts and organizations [45] [37]. Such objectification may happen in tacit, implicit or explicit ways and over time artefacts are designed and redesigned to suit the practice in which they are applied.

Within participatory design and user-centred approaches to development, the assumption is that it is essential to connect to the knowledge that actors apply in order to design artefacts appropriate for the task at hand. The three different forms of knowledge, however, present considerable challenges for developers of new artefacts, since the artefacts and work practices may incorporate tacit, implicit and explicit knowledge at the same time – hence, the knowledge embedded in actors and artefacts may not be explicable in total.

There are at least two challenges: First, the explicit knowledge that actors have or that they can produce of their practice may not be accurate, just as the explication by observers may not be accurate. A difference may exist between what actors say and what they do, and observers'

conceptualisation of the practice of others may equally not be adequate. Representation of practice is not easy [36, 38]. Second, tacit knowledge cannot be articulated and hence only shown or learned through ongoing interaction.

The first point implies that developers have to make critical judgements upon how to conceptualise practice and whose representation they want to rely upon. Even if developers and users agree upon a representation, subsequent implementation of the new artefact may prove them wrong. The second point implies either that developers engage in the time consuming task of learning the tacit knowledge applied by users in practice through ongoing participation in that practice, or that the design process is seen as extending beyond the introduction of the new artefact: only through ongoing use and redesign will it be possible to design the artefact to meet the tacit requirements.

Both points make the process of appropriation a crucial test to the design of an artefact. Only through ongoing use will it become clear whether the conceptualisation of work practice and requirements for the artefact have been right. Processes of appropriation, however, are inherently complex and involve the organizational context, the change of work practices, and the acceptance by actors. All affect the process of appropriation and make an assessment of the design choices difficult.

Several studies have described processes of appropriation at the levels of organization, work practice and actors [6, 21, 28, 44], while the literature around tailorability, customization and end-user computing [15, 25, 40] can be seen as attempts to facilitate flexible appropriation at the level of the artefact. In contrast to these 'weak' forms of flexible adaptation, others conduct investigations into 'stronger' forms of adaptation where the technology is changed in ways beyond the intentions of the designers (e.g. [12]).

As we will show, the case of Zephyr shows a 'strong' adaptation of the artefact, which was change to support knowledge management in a way radically different from that intended by the designers of the system.

THE CASE: KNOWLEDGE MANAGEMENT IN ZEPHYR

Zephyr develops and implements large-scale IT systems for the news industry and has approximately 400 employees. The main part of the employees work at the company headquarters, while the remaining part works in sales offices across the globe.

During a period of two years, the company doubled the number of employees. This rapid growth put a strain on organizational processes and routines pertaining to learning, cooperation and knowledge sharing among employees. To cope with these challenges, Zephyr's management launched a number of knowledge management initiatives aimed at improving knowledge sharing and innovation. These initiatives resulted in the creation of a collaborative knowledge management system, Knowledge Repository.

Research Setting and Methods

The analysis of knowledge management initiatives in Zephyr is based on the active participation of the second author maintenance, support and development of Knowledge Repository. This work was carried out in a task force, the so-called Knowledge Repository group, which was comprised of 7 Zephyr employees from different departments. The findings in this paper is based on two years of participatory observation captured in frequent field note entries, a series of user studies including 12 interviews with Zephyr employees from various departments and a member of the Zephyr board, interface usability testing of Knowledge Repository with 12 users, and 5 workshops for development of new knowledge sharing concepts.

The studies started when Knowledge Repository had been in use for two years and were carried out over a period of two years. Insights into the actual use of the system and its appropriation were thus gained.

The studies of the initial development is based on interviews with members of the Knowledge Repository group and the Zephyr board and on studies of documents from the development process, e.g. internal memos describing concept and vision of the project, the functional specifications, presentations of the project, etc.

Knowledge Sharing Problems at Zephyr

The Knowledge Repository system was developed in-house at Zephyr and was intended as a cross-organizational system for capturing and making available employees' knowledge and experiences.

During Zephyr's rapid expansion, a number of problems related to the dissemination of information and knowledge throughout the organization became evident:

First of all, it took a long time to train new employees. Depending on the job, Zephyr managers estimated that it would take as long as a year-and-a-half for new employees to be "worth their salary". Prior to the expansion, new employees could work in teams with more experienced peers who monitored and helped them. This was no longer feasible, as roughly half the employees were new.

Second, a number of managers as well as employees were complaining about loss of knowledge in project hand-over processes. The systems that Zephyr develops are highly complex and many specialists take part in their creation, implementation and support. A typical systems development project starts out in the development department, and then moves on to the documentation department, upon which the sales department takes over. When systems are sold, the implementation department implements the system at the customer's site, and the customer later on contacts the support department in case of problems. During this process, hand-over meetings are held when the system moves from one phase to another.

However, employees often complain that they do not have adequate information about the project status, or that they work on a problem for a period of time only to discover that someone in a different department has carried out the same work unbeknownst to them.

Third, Zephyr employees rely on a large number of collaborative systems to carry out their work. Some of these systems are used throughout the organization, such as e-mail and calendar applications, whereas other systems, including software bug report generators, work assignments etc., are used only in one or few departments. A number of the smaller systems have been developed and maintained by the employees of these departments in response to local needs and practices. Only intended for intra-departmental use, many of these systems are not documented and the information within them is not generally accessible. For new employees, learning to use these systems relies on competent colleagues demonstrating and explaining what they are intended for and how they are to be used, i.e. on being taught how to make sense of the mesh of systems.

The Development of the Knowledge Repository System

These issues, combined with an expressed need from Zephyr's management to have an overview of the organizations knowledge and competencies in order to better plan future initiatives and allocate resources, comprised the main incentives for the development of the Knowledge Repository system. The development of the system was initiated by Zephyr's management, and the expressed vision was to create a central system for the capture, storage and dissemination of employees' knowledge and experiences. The system should in turn make the organization less dependent on individual employees, since their knowledge could be stored in the system and made easily accessible for co-workers.

In order to determine how to enable this, a number of studies were carried out to make clear what information was stored in the intra-departmental systems, how employees stored information on their computers, and what kind of documents employees requested in regards to project handovers etc.

Based on these studies, Knowledge Repository was constructed to be first-and-foremost a *document management system.* The backbone of Knowledge Repository was a collection of documents written by and uploaded by Zephyr employees. A document classification scheme was established by the Knowledge Repository group to ensure standardized meta-data and to solve the problem of contextualization of information. Upon uploading a document, employees were thus requested to fill out a range of meta-data, including an abstract, keywords, expiry date and a list of departments and/or co-workers for which the document was relevant. Using advanced search features employees could later filter and retrieve documents by selecting the appropriate meta-data. Knowledge Repository contained a number of minor features to supplement the document management-features. Among these was the *Employee Information.* Every Zephyr employee was listed in Knowledge Repository with data such as a photo, job title, office location, phone number, e-mail and a list of documents uploaded to Knowledge Repository by the employee in question. Other features of the system included links to a number of legacy intra-departmental systems, employee handbooks etc.

The Knowledge Repository system was introduced to the organization at a large company presentation. Training sessions in which the various features were described in detail and Zephyr employees tried out the system hands-on followed this. Furthermore, it was made clear that all employees were responsible for documenting what they deemed relevant knowledge and making it accessible through Knowledge Repository. In order to ensure that the system contained up-to-date and relevant knowledge, each document was to be marked with an expiry date upon which the author was to update the document, renew the expiry date or remove the document.

The crisis of Knowledge Repository

Despite managerial support, extensive training and ongoing refinements of features and user interface, Knowledge Repository suffered from low acceptance in Zephyr. When the Knowledge Repository Group (including the second author) started studying the system after it had been in use for two years, the use of the document management features was very limited. A small number of employees had uploaded most of the documents in the system, and realizing that this was the case, these employees felt less than compelled to keep updating their old documents, let alone write and upload new ones. Many employees stated that time could be better spent "doing actual work" than updating the system. The only feature that had gained acceptance was Employee Information which was used on a daily basis by many employees to put a face on the co-workers they communicated with on telephone or via e-mail.

Summing up, Knowledge Repository never fulfilled the expectations placed upon it. It was only used sparsely in parts of the organization, and most of the time, employees would disregard the system and use the old, intra-departmental systems. Studies of Knowledge Repository revealed a number of concrete and tangible problems, including usability problems related to the user interface and problems in the technical implementation of the system, as well as user acceptance problems, have been identified. Despite a great deal of resources having been allocated to solve these issues, the use of Knowledge Repository remained unacceptable in the light of the management's initial vision for the system.

The Emergence of Parasitic Systems

Independently of the Zephyr management and the Knowledge Repository group, however, emerged a number

of systems so-called *parasitic systems*. The systems were labelled parasitic by the management and the Knowledge Repository group because they drew upon the Employee Information from the Knowledge Repository database, but bypassed the main system's user interface and instead appeared to users as simple stand-alone applications. The first of the parasitic systems was conceived and developed by a Zephyr developer tired with the rather complex and cumbersome interface of Knowledge Repository system. The developer only used the Employee Information features of Knowledge Repository, and instead of launching the system and clicking through a number of screens, he instead devised a small application that consisted of a search field and a results page. Users could enter initials or names into the search field and were then presented with some of the Employee Information from Knowledge Repository on the results page. Launching this application was a lot faster than launching the main system and the speed combined with the simplicity of the interface appealed to the developer's colleagues who started using the system. Through positive word-of-mouth and via e-mail the system spread throughout Zephyr's development department. After a few months of use, another developer refined the system and added additional information from Knowledge Repository to the search results. The effect was increased use and further adoption of the modified system by Zephyr employees outside of the development department. Both the initial and the refined version of the parasitic system had been developed in the developers' own time or in between work assignments. They were in no way condoned by the Zephyr management or the Knowledge Repository group, who in fact regarded the use of the systems as detrimental to the Knowledge Repository project. The Knowledge Repository group discussed whether the parasitic system should be banned, but keeping the system's widespread use in mind, this was thought to cause negative outbursts and ill will towards the Knowledge Repository project.

In the end, a developer was eventually granted resources by the development department management to further refine the system and integrate it into the Zephyr Intranet, a fairly simple intranet website that contained organizational news stories, employee manuals etc. as well as links to the independent intra-departmental systems. Most employees used Zephyr Intranet on a daily basis out of curiosity for news stories or because of the links to the many intra-departmental systems – in fact, many employees had set their browsers to display Zephyr Intranet upon launch.

The third iteration of the parasitic system was expanded to include a map of the Zephyr headquarters and contained further employee information from the Knowledge Repository database. Upon search, users of the system could now also see the location of the office in which co-workers were placed. In case of multiple employees sharing an office, the search results would also contain links to information about office-sharing employees. This version of the system was integrated into Zephyr Intranet via a search bar on the front page. Zephyr employees responded very positively to this version, dubbed *Employee Search*, and it quickly became one of the most used features on the intranet. This affected the Knowledge Repository system in that the only feature of the system that had been used on a frequent basis, the Employee Information, was now very rarely used.

The Knowledge Repository group launched a number of initiatives to boost the use of Knowledge Repository, including extra training and support in uploading and attaching meta-data to documents, but to little avail. When the studies at Zephyr ended, the Knowledge Repository system was in a state of coma: seeing that its use was dwindling, the employees who had been using it were now looking elsewhere to more well-used systems for finding and sharing documents and information.

CONCEPTUALIZATION OF KNOWLEDGE

The pertinent question remains: Why did Knowledge Repository fail to the expectations placed upon it? Finding the one true answer to this question is not feasible since a number of aspects influenced the development and acceptance of the system. This includes usability issues and problems in the technical implementation of the system. However, although these issues were solved along the way, the system remained largely unused.

We think that problems of a more fundamental order were central: The conceptualization of knowledge put forward in the vision driving the development of Knowledge Repository was inadequate and did not match the complexity of knowledge processes that occur in Zephyr. This inadequate conceptualization of knowledge was made manifest in the creation of Knowledge Repository. This was highly critical since the system was intended not only to support but ultimately to transform the knowledge sharing processes in the organization. Thus, the fairly abstract vision and concepts had real consequences for the Zephyr employees in that the actual construction of Knowledge Repository was faithful to the vision.

A Document-centric Approach to Knowledge Sharing

The primary vision behind the Knowledge Repository system was one of capturing the knowledge of Zephyr employees, storing it in a central repository in a fixed form, and making it available to everyone in the organization. This may be construed as a document- or information-centric approach, in that it relies on the transformation of knowledge in the heads of employees (i.e. implicit or explicit knowledge) into information put down in writing in documents and the subsequent technologically supported management of these documents.

However, as pointed out by Wilson [46], tacit and implicit knowledge may not easily be documented, if at all – and even if it were possible, it is seldom worth the effort. This is made clear by the case at hand for several reasons:

First, going from knowledge-in-the-head to articulate information-in-writing is a *transformation process* of which the result is different from the "source material". This transformation process implies *reduction*, *decontextualization*, and possibly *recontextualisation* in order to produce information that makes sense to people other than the author. In other words, when implicit knowledge is articulated and codified, it is somehow different from the author's personal knowledge.

Second, *it takes an effort* to document implicit and explicit knowledge: It is time-consuming to reflect upon which knowledge it is relevant to document, to contextualise it, and subsequently write it [18].

Third, given the fact that Zephyr operates in a highly innovative market and is continuously developing or reshaping products, much documented knowledge might soon be rendered *irrelevant* or *out-dated*.

Pipek and Wulf experienced similar problems when trying to implement the knowledge management system Answer Garden in a steel mill [31]. Contextualising information was difficult since people did not understand the meta-information used to contextualise, and because of the long history and different systems of categorisation applied. Sharing information could not be based on documents, since asking the right question required extensive dialogue and visual information as well [31: p13].

Dynamic Knowledge Processes at Zephyr

Our observations of actual work processes in Zephyr and interviews with employees point towards a more diverse conceptualization of knowledge than the one expressed in the Knowledge Repository vision.

Zephyr is continuously developing new products and updating old ones. For employees, this implies constantly keeping up with what colleagues are doing. To exemplify this, we may focus for a moment on the product managers who oversee the development of specific products. Product managers have to stay up-to-date with the ongoing development of certain products as well as the tasks of the people working on the specific products. In interviews, product managers described that they spent most of their time talking to people, either on the phone or by walking around the Zephyr headquarters to seek them out (See e.g. [7, 8, 23]) on mobility and work processes). Things would change at such a rapid pace in product development that the only way for product managers to keep up was to talk to the project workers doing the actual changes. The product manager then disseminated the knowledge of ongoing work processes to other project workers for them to work in tune to the latest changes. A number of intra-departmental legacy systems helped in this ongoing coordination, but the product managers were instrumental in keeping an overview of entire development processes and keeping them on track. Not delving into the same level of detail, similar situations played out for most Zephyr employees: Developers would talk to each other about plausible, but yet unrealized solutions to various problems, software testers would call developers to ask what might have caused specific bugs to occur, documentation specialists would talk to software trainers about educational strategies, sales people would consult implementation specialists to inquire about deadlines and user acceptance issues etc.

As pointed out by Grinter [17], some aspects of collaborative work are made easier by (or are indeed only possible because of) workflow systems that structure recurring processes. We do not disagree in these findings – many Zephyr employees used legacy intra-departmental workflow systems – but the case suggests that such systems were not adequate and that much knowledge does not easily fit into them.

Supporting Different Forms of Knowledge

The success of the parasitic systems and their ongoing refinements, in spite of the lack of managerial support, indicate that these systems support the knowledge sharing needs of Zephyr employees in a more adequate way than Knowledge Repository. Although the parasitic systems are very simple compared to Knowledge Repository and contain no document management features, only employee information, they facilitate contact between employees and thus support the inter-personal dissemination of fluid knowledge. The option of seeing where co-workers are located in the Zephyr headquarters and with whom they share offices supports an awareness of probable social ties not otherwise afforded by Knowledge Repository.

The developers of Knowledge Repository opted for the IT-track of knowledge management by supporting information sharing, which for employees implied the extra work of contextualising their information sufficiently to enable sense-making. The parasitic system is more in line with the people-track of knowledge management, in that it supported knowledge sharing and creation by way of supporting dialogue.

ISSUES OF APPROPRIATION AND EVOLVING USE

Drawing upon the "situated action" perspective introduced by Suchman [39], Dourish [12] proposes that "the ongoing, incremental adaptation of interactive technologies is inherent to the emergence of practice" and that the solutions to problems of adaptation of technologies "must involve some kind of transformation of the technology out of which software systems are constructed" [12: p467]. The case of Knowledge Repository and the parasitic systems highlights a number of issues vis-à-vis the ongoing appropriation and adaptation of collaborative systems: First of all, it brings into focus the skills that are required from end users in order for them to appropriate the systems and make them fit into domain practice. Second, it invites a discussion of the intertwined nature of appropriation and conceptualization and the possible consequences of inadequate conceptualizations in the development process. Third, it points to the ways in which designers and formal

decision-makers deal with appropriation and evolving use of collaborative systems.

Hacking and Redefining a Collaborative System: Weak and Strong Appropriation

CSCW research often focuses on the social transformations brought about by the introduction of collaborative systems (e.g. [10, 18, 29]).

However, as the case of Knowledge Repository shows, the appropriation of a collaborative system is a reciprocal process in which both social and technological transformations take place. Many systems are designed for a certain degree of tailoring or customization ([4, 19]) to allow users to modify the system to fit into their work practices.

This allows for a certain degree of appropriation. However advanced such customization features may be, they remain part of the system and as such are the result of developers' conceptualization of what might be purposeful for users.

The parasitic systems of Zephyr, however, represent a radical change from the original Knowledge Repository system and are not merely a question of tailoring. They are instead stand-alone systems that hack into the original system. This brings into view the question of what kinds of technological change appropriation covers and what kinds of technological competencies are required of users who wish to adapt a system. As such, one may argue that the case of the parasitic systems tapping into Knowledge Repository is not one of appropriation in the typical sense of the word, since it was not the original system that was adopted and adapted for use. The parasitic systems rather represent an evolutionary adaptation that was not feasible by way of customizing Knowledge Repository on its own terms – that is, the new and widely accepted systems could not have been brought about were it not for the technological insight and prowess of the systems developers who devised them. Had the original system been successful, "typical" users might have appropriated it in ways described by Suchman as "small acts of subversion taken up in the name of getting things done" [37]. Such forms of appropriation require much less technological insight than what was required to construct the parasitic systems. Thus, it makes sense to make a distinction between what technical skills are required for users to appropriate systems, as is done in recent research in End-User Development and Component-Based Development [41].

At one end of the spectrum, there are users who have just enough technological insight to use a system. Over time, such users may appropriate a system in a way that incurs no actual changes to the system itself but rather consists of new ways of using the system as it is (creative misuse) or new ways of ascribing sense to the system and/or the information it contains.

More advanced users may have the skills to tailor systems to their existing work practices by using built-in features such as customization, macros etc.

Users with programming skills may furthermore be able to create plug-ins that enhance the features of existing systems, given that these systems support external plug-ins.

Finally, expert users such as the developers responsible for the parasitic systems may go one step further and reshape existing systems or ultimately create new systems that may draw upon information from legacy systems.

Although we have focused on the latter group of users and the rather radical form of appropriation they have brought about, other forms of appropriation did take place to a limited degree regarding Knowledge Repository and to a more extensive degree regarding the parasitic systems. The adoption of the parasitic systems, evidenced by the ongoing iterative refinements and their proliferation from the initial user group in the development department to the whole of the organization, leads us to argue that appropriation may consist of big acts of subversion taken up in the name of getting things done – acts that may undermine or topple efforts at creating collaborative systems if these do not match users' needs and practices. We thus argue for a distinction between *weak appropriation* (encompassing different ways of ascribing sense to and using existing systems and using built-in features to customize the system to use domains) and *strong appropriation* (encompassing technical alterations of existing systems and creation of new systems in place of original systems). The fact that appropriation occurs, be it weak or strong, is not in itself an indicator of the success or failure of a collaborative system. Ways of weakly appropriating a system may increase the usefulness of the system by making it malleable in response to contingencies on the use context. It may, however, also lead to confusion and incongruence of use, thus hindering collaboration and coordination. In the same way, strong appropriation may occur as a testament to the fact that the system has great potential for supporting work. It may, however, also occur as way of creating alternatives to inappropriate systems.

Conceptualization and Appropriation Intertwined

The development of Knowledge Repository was a carefully planned process that included elements of user participation: User studies and interviews were conducted to determine how Zephyr employees stored and exchanged documents. The strong appropriation process and the emergence of the parasitic systems serve to highlight the inadequacy of the initial vision and conceptualization of the use domain. We thus argue that there exists an intertwined and reciprocal relationship between supporting a tenable conceptualization of the use domain and the subsequent appropriation of the system into that domain.

This finding is somewhat problematic in that it points to the fact that it may not always be possible to determine whether correct conceptualizations about the use domain

have been made in the initial development process – only by studying the appropriation of collaborative systems after their deployment is it possible to determine whether the conceptualizations are adequate for supporting the use domain.

A further problematic aspect of the case is the fact that the parasitic systems, conceived of and devised as they were by individual developers acting on their own, perhaps indeliberately struck the right conceptualization of knowledge work processes in Zephyr,. This serves to illustrate that user studies and involvement may not always be sufficient in creating systems that fit into use domains. If inadequate concepts underlie the development process, designers might not ask the "right" questions when studying users or inviting them to participate in the development process. If the "wrong" questions are asked and answered, it could serve to embed and manifest the inadequate conceptualizations in the systems developed.

Dealing with Evolving Use and Adaptation: learning from the parasitic systems

In her study of a collaborative system, Grinter [17] points out that managerial support is an important factor for workflow systems to succeed. The Zephyr case points to the fact that collaborative systems may succeed not because of but in spite of initial managerial support. Knowledge Repository had the support of the Zephyr management and the dedicated Knowledge Repository group, whereas the use of the parasitic systems was discouraged. The right conceptualization of the use domain may have been a more important factor than managerial support.

As mentioned the Knowledge Repository Group labelled the independently developed systems "*parasites*", and as the negative connotations of the label indicate, the Knowledge Repository group was distressed to discover the widespread use of the systems that was perceived as detrimental to the acceptance of Knowledge Repository. This initial reaction is understandable, bearing in mind that many resources had been allocated to the Knowledge Repository project and that the Zephyr management as well as the Knowledge Repository group had invested a great deal of prestige in it.

There are thus lessons to be learned from studying acts of subversion, be they small or large. Instead of branding such efforts as illegitimate, we rather suggest that they be regarded as invitations to discuss the concepts underlying the system and the ways in which the system may be improved to better fit the domain.

CONCLUSIONS AND PERSPECTIVES

It is of course a complex task to point out the multiple reasons as to why the processes of design and appropriation of Knowledge Repository in Zephyr developed as they did. We have tried to show that the conceptualization of knowledge and how this was embedded in the system was a crucial factor and that ability to appropriate Knowledge Repository in a 'strong' sense was another. Other factors, such as those mentioned in [18], may have affected the use of Knowledge Repository. We think, however, that the case of Zephyr raises three critical points.

First, Zephyr and the development of Knowledge Repository points to the importance of conceptualizing the application domain of an artefact appropriately. User-centred design and participatory design are strong steps in the right direction for getting a design right, but do not necessarily overcome these contingencies. As we have argued, the three different kinds of knowledge (tacit, implicit, explicit) and the possibility that neither actors themselves nor observers appropriately conceptualize work practice and the objectified embedded knowledge in existing artefacts appropriately, make the development of new artefacts (and by implication new work practices) a challenge. In addition to technical abilities and close connection to work practice, a design process requires critical conceptualizations and extended periods of use of the new artefact. Iterative design processes are an option, but it may take considerable time – possibly years – before actors, work practices and organizations have stabilized around a new artefact (see e.g. [33]). In the case of Zephyr, it became clear rather quickly that employees did not regard the way knowledge sharing had become conceptualized as appropriate.

Second, the question of which conceptualization is appropriate is not only a question of accuracy, since we all engage in situated practices. What different actors and observers regard as appropriate is very much a question of their position. A classical case is the example of a workflow system for a printing company, which was not appropriate to a flexible and effective work practice, because it was design to make bills for tasks handled [10]. Seen from the workers perspective the system was inappropriate, whereas the opposite was true from the perspective of management. Many workflow systems are simultaneously accountancy systems as Dourish [13] argues, and Zephyr's Knowledge Repository shows the same ambivalence: it is supposed to simultaneously support knowledge sharing between employees and provide management an overview of what is going on. Neither perspective is wrong, but as the case shows they were not both supported appropriately. While in theory it might be possible to meet the requirements of multiple parties, it is in practice difficult. Hence, not only does the design process entail critical conceptualisations of the application domain, but also making critical design choices which handles different and possibly incompatible requirements.

Third, it is only in practice that it will become clear whether an artefact has been conceptualized and designed appropriately and since there is some chance that something has gone askew, the design process should be seen as extending into the phase where a new artefact is appropriated and used in everyday context. This is rarely

the case with knowledge management systems where the emphasis of the seller is on the advantages and so-called quick-wins that will arise once the system has been introduced. A prolonged phase of redesign after implementation is seldom envisaged. We suspect this to be the case with most systems. While we would argue for as much tailorability, customization, and end-user computing as possible in order to strengthen the processes of appropriation in the direction of maximising appropriateness, there are two problems with this approach.

One is how to organize the different technical and organizational knowledges and the practical experiences with the artefact to ensure that the design process continues. Most users do not have the competencies required to handle the complexities of adaptation, but get lost in the many options in e.g. a word-processing or photo-editing system. A high degree of technical appropriation demands high skills from users and can in many cases only be handled by experts of some kind. In the case of Zephyr, 'strong' appropriation was possible because the users were themselves developers, but this is an exceptional case.

Another is that there may be good reasons to limit the scope and depth of changing the system. One strong position in the field of computer-supported cooperative work has been to design in ways that interrupt as little as possible the practices of actors. There are good reasons not just to design artefacts that help actors to orientate themselves in their interactions, but also to design them as artefacts that ensure that practice follows a prescribed sequence of action are followed. E.g. the checklist that a pilot making ready for take-off makes use of, or the instruction cards physicians at some hospitals carry with them that prescribe how to act in case of heart strokes. In some situations, e.g. safety-critical practices, there should be strict limits to user-adaptation of systems.

The complexity of the processes of design and appropriation preclude any easy answers, but as the case of Zephyr shows critical concepts and critical analysis of the interrelationships between various forms of knowledge, work practices and artefacts may lead us towards asking the right questions.

Artefacts, actors and organizations intertwine in complex ways in the mangle of practice [30]. "What is the Matrix?" Neo asks in the movie *The Matrix*. "It cannot be explained, but only shown", answers Morpheus. What applies to the virtual world of the Matrix, also applies to the relationship between conceptualizations and appropriation: it can only be shown by critically examining conceptualizations and evolving use.

ACKNOWLEDGMENTS

The authors would like to extend a warm thank to Zephyr management and employees for their helpful cooperation.

REFERENCES

1. Ackerman, M.S. Augmenting Organizational Memory: A Field Study of Answer Garden. *ACM Transactions on Information Systems, 16* (3). 203-224.
2. Ackerman, M.S. and Halvorsen, C. Organizational Memory: Processes, Boundary Objects, and Trajectories *Proc. of HICSS´99*, 1999, http://www.computer.org/ proceedings/ hicss/0001/00011/00001toc.htm.
3. Ackerman, M.S. and McDonald, D. Answer Garden 2: Merging Organizational Memory with Collaborative Help. *Proc. of CSCW´98*. 97-105.
4. Andriessen, J.H.E., Hettinga, M. and Wulf, V. Evolving Use of Groupware. *Special issue of Computer Supported Cooperative Work 12(2)*, 2003.
5. Bannon, L. and Kuutti, K. Shifting Perspectives on Organizational Memory: From Storage to Active Remembering. *Proc. of HICSS'96*. 156-167.
6. Barley, S.R. Technology as an Occasion for Structuring. *Administration Science Quarterly, 31*. 78-108.
7. Bellotti, V. and Bly, S. Walking away from the Desktop Computer. *Computer Supported Cooperative Work*. 209-218.
8. Bertelsen, O.W. and Bødker, S. Cooperation in massively distributed Information Spaces. *Proc. of ECSCW´01*, Klüver, Netherlands, 2001, 1-17.
9. Bijker, W.E. *Bicycles, Bakelites and Bulbs. Towards a Theory of Sociotechnical Change*. MIT Press, Cambridge (Mass) & London, 1995.
10. Bowers, J., Button, G. and Sharrock, W. Workflow from Within and Without *Proc. of ECSCW´95*, Klüwer, Dordrecht, 1995, 51-66.
11. DeSanctis, G. and Poole, M.S. Capturing the Complexity in Advanced Technology Use: Adaptive Structuration Theory. *Organization Science, 5* (2). 121-147.
12. Dourish, P. The Appropriation of Interactive Technologies. *Computer Supported Cooperative Work, 12* (4). 465-490.
13. Dourish, P. Process Description as Organisational Accounting Devices: the Dual use of Workflow Technologies. *Proc. of GROUP´01,* ACM Press, New York, 2001, 52-60.
14. Fahay, L. and Prusak, L. The Eleven Deadliest Sins of Knowledge Management. *California Management Review, 40* (3). 265-276.
15. Fischer, G. and Giaccardi, E. Meta-Design: A Framework for the Future of End User Development. in Lieberman, H., Paternò, F. and Wulf, V. eds. *End User Development - Empowering People to Flexibly Employ Advanced Information and Communication Technology*, Klüwer, Dordrecht, 2005, in press.

16. Greenbaum, J. and Kyng, M. *Design at Work. Cooperative Design of Computer Systems*. Lawrence Earlbaum Associates, London, 1991.

17. Grinter, R.E.q. Workflow Systems: Occasions for Success and Failure. *Computer Supported Cooperative Work, 9*. 189-214.

18. Grudin, J. Why CSCW Applications Fail: Problems in the Design and Evaluation of Organizational Interfaces. *Proc. of CSCW´88*. 85-93.

19. Kahler, H., Mørch, A., Stiemerling, O. and Wulf, V. Tailorable Systems and Cooperative Work. *Special Issue of Computer Supported Cooperative Work, vol. 9(1)*, 2000.

20. Karsten, H. Constructing Interdependencies with Collaborative Information Technology. *Computer Supported Cooperative Work, 12* (4). 437-464.

21. Karsten, H. and Jones, M. The Long and Winding Road: Collaborative IT and Organisational Change. *Proc. of CSCW'98*, ACM, Seattle, 1998, 29-38.

22. Latour, B. Technology is Society made Durable. in Law, J. ed. *A Sociology of Monsters: Essays on Power, Technology and Domination*, Routledge, London & New York, 1991, 103-131.

23. Luff, P. and Heath, C. Mobility in Collaboration. *Computer Supported Cooperative Work*. 305-314.

24. Miller, F.J. I = 0 (Information Has No Intrinsic Meaning). *at http://www.fernstar.com.au/publications/papers/i=o.htm*

25. Mørch, A., Stevens, G., Won, M., Klann, M., Dittrich, Y. and Wulf, V. Component-based technologies for end-user development. *Communications of the ACM, 47* (9). 59-62.

26. Nichols, F. The Knowledge in Knowledge Management. in Cortada, J.W. and Woods, J.A. eds. *The Knowledge Management Yearbook 2000-2001*, Butterworth-Heinemann., Boston, 2000, 12-21.

27. Orlikowski, W.J. Action and Artifact: the Structuring of Technologies-in-Use, Sloan School of Management, Massachusetts Institute of Technology, 1995.

28. Orlikowski, W.J. Evolving with Notes. in Ciborra, C. ed. *Groupware and Teamwork*, Wiley, Chichester, 1996, 23-60.

29. Orlikowski, W.J. Improvising Organizational Transformation over Time: A Situated Change Perspective. *Information Systems Research 7(1)*, 63-92.

30. Pickering, A. *The Mangle of Practice: Time, Agency and Science*. University of Chicago Press, Chicago, 1995.

31. Pipek, V. and Wulf, V. Pruning the Answer Garden: Knowledge Sharing in Maintenance Engineering. *Proc. of ECSCW´03*, Klüver, Netherlands, 2003, 1-20.

32. Polanyi, M. *Personal Knowledge: Towards a Post-Critical Philosophy*. University of Chicago Press, Chicago, 1958.

33. Prinz, W., Mark, G. and Pankoke-Babatz, U. Designing Groupware for Congruency in Use. in *Proc.of CSCW'98*, 1998, 373-382.

34. Rogers, Y. Exploring Obstacles: Integrating CSCW in Evolving Organisations. *Proc. of CSCW´94*. 67-77.

35. Schuler, D. and Namioka, A. (eds.). *Participatory design: Principles and practices*. Lawrence Erlbaum Associates., Hillsdale, NJ, 1993.

36. Star, S.L. The Sociology of the Invisible. in Maines, D. ed. *Social Organization and Social Process*, Aldine de Gruyter., Hawthorne, NY, 1991, 265-283.

37. Suchman, L. Located Accountabilities in Technology Production. Department of Sociology, Lancaster University, at *http://www.comp.lancs.ac.uk/sociology/soc039ls.html*.

38. Suchman, L. Making Work Visible. *Communications of the ACM, 38* (9). 56-63.

39. Suchman, L. *Plans and Situated Actions. The Problem of Human-Machine Communication*. Cambridge University Press, Cambridge, 1987.

40. Sutcliffe, A. Scenario-based Requirements Engineering. in *RE'03 Proceedings*, IEEE, Monterey Bay, California, 2003, 320-329.

41. Sutcliffe, A. and Mehandjiev, N. Special Issue on End-User Development: Tools that Empower Users to Create their Own Software Solutions. *Communications of the ACM, 47* (9).

42. Sveiby, K.E. What is knowledge management? Brisbane, Sveiby Knowledge Associates. *at: http://www.sveiby.com/faq.html#Whatis*.

43. Takeuchi, H. and Nonaka, I. Theory of Organizational Knowledge Creation. in Morey, D., Maybuny, M. and Thuraisingham, B. eds. *Knowledge Management: Classic and Contemporary Works*, MIT Press, Cambridge (Mass) & London, 2000, 139-182.

44. Törpel, B., Pipek, V. and Rittenbruch, M. Creating Heterogeneity. Evolving Use of Groupware in a Network of Freelancers. *Computer Supported Cooperative Work, 12*. 381-409.

45. Walsh, J.P. and Ungson, G.R. Organizational Memory. *Academy of Management Review, 16* (1). 57-91.

46. Wilson, T.D. The Nonsense of 'Knowledge Management'. *Information Research, 8* (1). paper no. 144. [Available at http://InformationR.net/ir/148 141/paper144.html].