About Representational Artifacts and Their Role in Engineering

Hilda Tellioğlu

Vienna University of Technology, Faculty of Informatics

Argentinierstrasse 8/187, A-1040, Vienna, Austria

hilda.tellioglu@tuwien.ac.at

Abstract

This chapter is about showing how artifacts impact engineering work processes by representing important issues of individual and collaborative design work. After summarizing the state of the art of engineering as a design process, artifacts and their representational role in design and engineering, a selection of rich descriptions of artifacts' creation and use in engineering work including team-based coordination and decision activities will be presented. The studies are based on ethnographic research carried out for several years in different design and engineering companies. Artifacts used in these studies will be analyzed from their representational point of view to discuss their important role in design and engineering, by considering users' motivation to use and sometimes adapt them as well as internal and external constraints given in work settings which call for improvisations, before concluding this paper.

Introduction

Engineering is about bringing together technologies to meet human needs or to solve problems (Khandani, 2005). Considering engineers as problem solvers, engineering design is about problem solving, which defers from other types of problem solving, in the nature of the problem and the solution. Design problems are open; they might have more than one correct solution. The solution to a design problem is normally a system that possesses specified properties, and the process of solving a design problem is usually iterative and cooperative, depending on its complexity.

A basic process of an engineering design can be described in five steps (Khandani, 2005, p.5ff): defining the problem, gathering pertinent information, generating multiple solutions, analyzing and selecting a solution, testing and implementing the solution. In all these steps, several tasks are evolved; several actors are involved; several sub-problems occur. Interactions for actions, possibilities, clarification, and orientation (Tellioğlu, 2003), needed for communication, coordination, decision-making, and improvisation, make use of representations of problems and solutions on the one hand, and of tasks, responsibilities, restrictions, and dependencies on the other.

Artifacts in general have qualities to help representing problems, possible solutions, and the state of a process. Within the CSCW research, the role of artifacts in everyday work of professionals was studied from several perspectives (Haas, 1996; Henderson, 1999; Nardi, 1993; Sellen, 2001; Schmidt and Wagner, 2004). Frameworks like organizational memory (Conklin et al., 1988; Conklin, 1989; Ackermann and Malone, 1990; Ackermann, 1997; Conklin, 1993), common information spaces (Bannon, 1997; Schmidt and Bannon, 1992), workflow systems (Grinter, 1996; Bardram, 1997), coordination mechanisms (Carstensen, 1996; Divitini et al., 1996; Schmidt and Simone, 1996; Simone and Divitini, 1997) or boundary objects (Star, 1989; Star and Griesemer, 1989; Bowker and Star, 1999; Lutters and Ackerman, 2002) have been developed to address specific aspects of coordinative practices. For instance, Bardram and Bossen studied non-electronic coordination artifacts at a hospital ward (Bardram and Bossen, 2005). They focused on “1) the material characteristics of these artifacts, 2) on how they order the world by providing templates, 3) on how they provide overview and signal ad hoc status, 4) on the importance of acknowledging the importance of meaning in collaboration, and finally 5) on the importance of supporting second order articulation work” (p.168). The network of these highly interwoven artifacts operates as a resource for action. It supports information flow, status overviews, synchronous and fine-grained coordination, and articulation of status.

Artifacts used in collaborative engineering processes can be identified and their use can be supported by well-designed computer systems. Our ethnographic studies, we have carried out for almost last 10 years, show that these tools are not really addressing the issues, which are important for designers. Designers do not want to give up their conventions and standards they have established so far when new tools are introduced into their work environments. What they usually do is to adapt their work habits to make them keepable with the systems they have to use, or to find a work around in order to meet their goals quantitatively and qualitatively. In fact, they want to continue working with their old known artifacts they invented, modified, or composed. How artifacts are shaped is a result of experiences, cooperative work, and conventions established in work groups. It takes time to create and adapt especially common artifacts to meet differing needs and procedures in work groups and to avoid misunderstandings and frustration of users.

To design more suitable IT-tools for designers and engineers, we have to understand their work practices. Several ethnographic studies (Tellioğlu, 2010a, 2010b, 2009, 2007) in manufacturing, electronic engineering, system design, and especially multimedia production show that knowledge workers overcome the complexity of their engineering work by using different types of artifacts in different formats and views depending on the setting they are currently in. For instance, multimedia designers have developed a particular visual culture, which allows them create and read design representations in different media. Most of these visual artifacts are produced as part of discussions, as an integral part of explanations, developments, and arguments, and they are re-used in follow-up meetings and sometimes annotated with supplements, modifications, and comments (Tellioğlu and Wagner, 2005).

Another example is the use of models by engineers. Models help engineers in overcoming complexities and to create a common understanding about processes and products. Organizational, commercial, technical, and process-based circumstances have impact on models and modeling practices. In a previous work (Tellioğlu, 2009), several modeling practices are identified and described: 1) modeling to visualize several important issues in a cooperative project, 2) modeling to support collaboration and coordination among members, 3) modeling to support system engineering for individual and group work, and 4) models triggering automated actions in workgroups. Different types of models are created and used in these different modeling activities. Each type consists of different information, has a different format, and normally a different role in the course of work processes. Analyzing how models are visualized and how they are used in which context, informs us about the work habits of engineers, about the problems they try to solve, about the circumstances under which they work and cooperate.

Analyzing artifacts and their representational roles in design-based engineering can help to create an approach for studying engineers' work processes and, in this way, design systems to meet their requirements. The main purpose of this chapter is to study, on the one hand, how artifacts are constructed and shaped and how they evolve during a design process, and on the other hand, in which ways artifacts' presence and use have impact on design decisions and practices. Besides the theoretical background, studies of work practices in two engineering teams are presented and analyzed to show empirical evidence to the results derived.

The chapter is structured as follows: First of all, the state of the art about engineering as a design process, different types of artifacts, and their representational role in design and engineering will be summarized. Then, a selection of rich descriptions of artifacts' creation and use in engineering work including team-based coordination and decision activities will be presented. Artifacts used in these studies will be analyzed from their representational point of view to discuss users' motivation to use and sometimes adapt them as well as internal and external constraints given in work settings, before concluding this paper.

Engineering as a Design Process

In engineering, designer is an engineer who designs and develops systems, and design is an engineering activity affecting human life, using laws and insights of science, building upon special experience, and providing the prerequisites for the physical realization of solution ideas (Pahl et al., 2007, p.1). Engineering requires systematic procedures combined with so-called heuristic principles or creativity techniques. Certain conditions must be given for an engineer to provide the possibility to apply such systematic approaches, like definition of an overall goal and individual sub goals, clarification of conditions by defined initial and boundary constraints, dispelling of prejudice to ensure the wide-ranging search for solutions possible, search for variants to find solutions or combinations of solutions, evaluation based on goals and conditions, decisions made built upon objective evaluations to enable progress (ibid, p.53).

A typical engineering design process consists of several steps, which can occur iteratively[1]: problem identification, research, requirements specification, concept generation, design, prototyping, system integration, and maintenance. So far several approaches for design are established in engineering: top down methodologies like functional decomposition, top-down iterative refinements of roughly finished designs to make it finer and more exact until the complete design is done, case-based methodologies focusing on similar design cases to model the own process, incremental re-design practices by unraveling an existing design from the bottom up, by modifying as required. Sometimes there is a need to combine both the top-down and bottom-up approaches to so called hybrid design processes. In case of creating design ideas, explorative methodologies may be very useful especially in the initial design and specification phases.

[1] Scott Umbaugh, Textbook: Design for ECE Engineers, Ford & Coulston.

Representational Artifacts

No matter individual or cooperative, actors usually need to use *artifacts* to carry out their work. They produce, read, annotate, modify, check, evaluate, communicate, and delete artifacts in the course of their individual or cooperative work. Besides mediating contacts among people by helping to search for people with specific competencies (Hertzum, 1999), artifacts mediate articulation work by acting as an intermediary with a specific material format between actors. For Kuutti "instruments, signs, procedures, machines, methods, laws, forms of work organization" are examples of artifacts (1996, p.26). Furthermore "an object can be a material thing, but it can also be less tangible (such as a plan) or totally intangible (such as a common idea) as long as it can be shared for manipulation and transformation by the participants of the activity" (Kuutti, 1996, p.27). In a usual work environment actors interact by using several artifacts. Where an artifact is placed, when it is accessed, by whom it is modified, what the exact modifications are, how these modifications are represented, etc. are important questions related to cooperation and especially coordination issues among the parties involved.

As being permanent symbolic constructs artifacts act as mediators of the coordination. At the same time, they are used to clarify ambiguities and to settle disputes. Mediating requires accurate representation of a certain level of details, which need to be exchanged. Representations are not real (Suchman, 1987), are local and temporary (Gerson and Star, 1986) and are "conventionalized practices based on rules of mapping and translation between representation and the object that is represented" (Schmidt and Wagner, 2004, p.15). Representational artifacts are immediate objects of the work (Schmidt and Wagner, 2004). They are objectifications of things-to-come and of things-in-the-process-of-becoming.

To understand the impact of artifacts being as representational objects we have to study artifacts used in work settings. Professionals interact usually with three types of artifacts in their current work environment (Cole, 1982; Hertzum, 1993). Some artifacts are readily at hand and often piled up on the desk. They mainly consist of action information. Others are within reach but put on shelves or binders. These artifacts can be considered as personal work files. Some artifacts are kept away from the office. They are used to archive relevant information for eventual future use. Hertzum focused on the relations “between the use of documents in the individual professionals’ execution of their current activities and the use of documents for information sharing among professionals who are separated in time or space” (1999, p.57). In this sense, artifacts as personal work files are mainly accessed by an individual person and through her/him by other actors within the work environment. These artifacts are arranged and maintained personally and not organized by categories defined in the work setting.

Another distinction is made by artifacts’ properties of mediating. Wartofsky defined three types of mediating artifacts [255], which has been further developed by Engeström (1990) into a three-level hierarchy using Leont’ev’s hierarchy of activity (1981). “Primary artifacts are tools used directly in production to mediate the relationship between the subject and object of activity; secondary artifacts are representations of modes of action – models – used to preserve and transmit skills in the production and use of primary artifacts; tertiary artifacts are imaginative or visionary and give ‘identity and overarching perspective to collective activity systems’ ” (Guy, 2003, p.3). This scheme has been further developed within activity theory (Engeström, 1999; Collins et al., 2002) and includes what-artifacts, which contribute a means of achieving the object, how-artifacts that contribute to understanding how to achieve the object, why-artifacts motivating achievement of the object, and where-to-artifacts motivating evolution of all elements in the activity system (Guy, 2003, p.3).

Lundberg and Sandahl have investigated artifacts’ properties in work by analyzing and identifying how artifacts’ peripheral properties become common resources within a community of practice and how artifacts are active within a work environment (1998). Actors need to see any time what others are doing, which is normally provided by a common artifact. Besides facilitating implicit communication within a team (Berlage and Sohlenkamp, 1999), a common artifact serves as a template representing a limited model of the work to be done by leaving enough space to be filled by its users. If this template is filled in then it becomes a record of the work. Common artifacts may coordinate activities by their visibility or by their arrangements. “Documents’ material and visible presence, in a shelf or on a table according to their structured trajectory in the radiology department and news agency afford the linking of actions and events over different sites and times without personal interaction between staff. ... the coordination of work is indicated by ‘who is holding the document’. The paper acts as a token and the shelf in which the documents are placed represents the state of work” (Lundberg and Sandahl, 1998, p.7). These artifacts are reminders of things to do and help manage one’s work activities (Hertzum, 1999). They are arranged spatially in particular ways. Some artifacts are formatted in ways that enable coordinating activities by tracking the work between cooperating actors within a work group. They are used then as tokens related to material objects, which are accessible to all involved in a shared work process (Schmidt and Wagner, 2002). The location of material artifacts includes some relevant information. Some of the actors can probably make sense of it and some others not. One can also see the history on a material artifact, the history of past work as well as the contributions of different actors. The materiality of artifacts is also referred to affordances (Gibson, 1979; Selen and Harper, 2001) or to immutability of inscriptions on paper and their mobility (Latour, 1986).

On the other hand, there are coordinative artifacts, which are material artifacts that have a coordinative role in carrying out work practices. In this sense they are communication objects and persuasive (Wagner, 2000). They help actors in several ways in their daily work: They create a common understanding of a task. They enable talking about a task in a rich way. They can be used consequently to share information with some, yet withhold it from others, e.g., for the reason to avoid affecting the power structures and privileges in a work environment (Hertzum, 1999). Sometimes it is important to reduce the multiple voices of an artifact. This makes them to convey meaning among the people sharing them. For using them, no continuous interpretation of their meaning is necessary.

Coordinative artifacts also remind principles, approaches and methods applied, questions that are still open. They also help keeping track of activities and materials (Schmidt and Wagner, 2002). Some artifacts usually used for coordination purposes contain work plans. They include work to do, project phases, how to proceed in a specific project phase, material to collect or create which is necessary to represent the work, methods defining rules and conventions within the work group, illustrations like sketches, images or photos to explain things or to generate new meaning, references to material to look for, names of actors responsible for certain tasks. As boundary objects (Star, 1989; Star and Griesemer, 1989) coordinative artifacts are accessed and modified by all responsible actors. They enable crossing organizational and professional boundaries many times. All decisions made in a project can be recorded and available in a common design artifact. Actors annotate their artifacts by circling certain areas, adding notes, or marking certain parts. This type of annotations makes artifacts multi layered. Multi layered artifacts “facilitate coordination between activities (and the people who are responsible for them). They, for instance, provide a collective or individual space for experimentation and change” (Schmidt and Wagner, 2002, p.10). That implies that artifacts are interrelated to work activities.

There are several types of coordinative artifacts (Schmidt, 2001, p.7):

- Traces are artifacts that indicate aspects of past activities of coordinating actors.
- Templates are artifacts that specify the properties of the result of individual contributions, like product standards, drawings, style sheets, etc.
- Maps are artifacts that specify interdependencies of tasks or objects in a cooperative work setting like organizational charts, classification schemes, taxonomies, etc.
- Scripts are artifacts that specify a protocol of interaction in view of task interdependencies in a cooperative work setting, like checklists, production schedules, office procedures, bug report forms, etc.

The changing of static state of all types of coordinative artifacts offers cues and an array of signals to other actors as to the intentions, challenges, and problems of the actor carrying out the changes (Schmidt and Wagner, 2002).

In the next section, we will present two cases of design-based engineering projects to illustrate different ways of working with representational artifacts in design processes. This depends on the type of the product, on conventions and standards established in the project team, on project circumstances being planned, foreseen or situated ad-hoc. We selected one multimedia company and one engineering company to show differences in approaches to design and engineering in the course of the project evolvement.

Case *Archcom*: A Multimedia Company

Archcom[2] is a multimedia company providing web applications with database integration for contemporary architecture, by combining web engineering with architectural design. It is founded in 1996. The main product of this company is an Internet forum for contemporary architecture. Its manager is an architect with a strong interest and advanced skills in multimedia production and use. He, in parallel, works on small architectural design and building projects. Four professionals with different backgrounds, like web production, designing multimedia architecture and web information, developing web content strategies, web programming, including two free lancers with skills in HTML, Java Script, Microsoft Access, Perl programming and image processing, and external programmers for the development of database integration, and graphic designers with experiences in design and implementation of Flash animations build the team of the project that we[3] were studying in the scope of an international project[4].

[2] The name is modified.

[3] The project team in Austria consisted of Hilda Tellioğlu, Ina Wagner, and Friedrich Glock.

[4] Our study was conducted as part of research project "Systemic Integration of Production and Services. Case Studies in the Software and Multimedia Industries", in cooperation with the Brandenburg University of Technology Cottbus and DJI Munich (2001-2003).

The Internet forum builds on a central database managing architectural data, containing images, journal articles, as well as technical data about architects and buildings. Its technical implementation is complex because it gets the data from distributed sources, each of them using slightly different technology. Interfaces to these sources need to be flexible enough to enable regular data transfer without failure. The multi-media character of the data adds to the complexity of the system.

We characterized this design process as best practice. This applies to planning activities, to the use of methods, and the documentation. The project manager uses a project plan in form of a spreadsheet for the management of deadlines, responsibilities, and workflows, which he updates regularly. During the weekly project meetings detailed to-do-lists are generated, which afterwards are used to organize the cooperative work within the team (Figure 1). These to-do-lists are essential for continuous project work. They contain a lot of data representing the status of the project, the distribution of work between project members, decisions made so far or added newly to the open issues, the importance of certain issues by explicitly showing their due date, status of some issues (either done, cancelled, or scheduled to the next milestone). The owner of these to-do lists is the project manager. He updates the actual list, which contains only open issues, during the regular project meetings by discussing the single issues with all team members. These to-do lists are also used as meeting minutes. They provide crucial coordinative data for team members to follow all decisions made about the functionality and user interface of the system, about priorities made to issues including certain tasks, about changes and enhancements of the underlying database model, about modifications of the web pages, about the business logic, about open questions and the responsibilities of team members for different tasks.

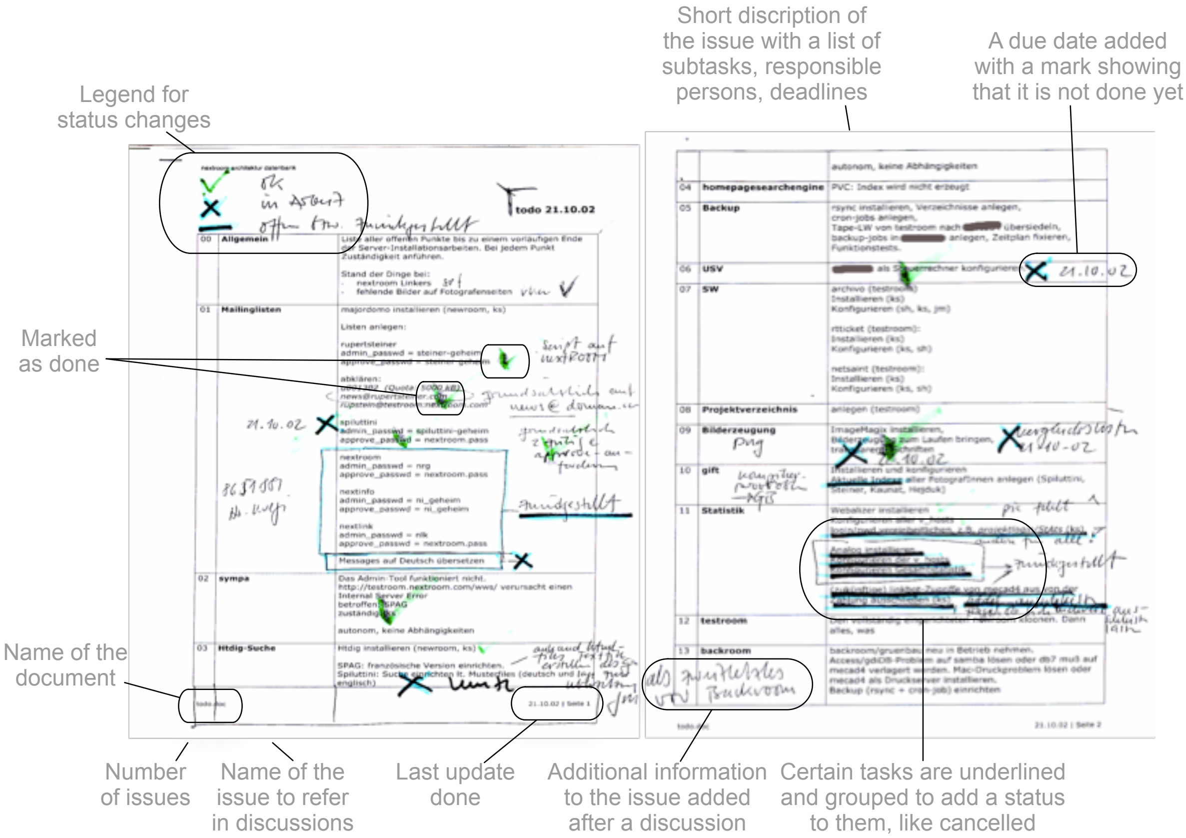

Figure 1: Part of a to-do-list created and updated regularly after weekly meetings showing the dynamic character of a coordinative artifact (*Archcom*).

With regard to important issues, for example concerning the design of the search function of the forum, thorough research is conducted in order to prepare a solid base for decision-making. This ended in detailed documents evaluating different approaches and technologies for searching. To integrate a search facility into the website, detailed sketches of navigation and graphic layout of the pages are produced, partly before and partly during the meetings cooperatively (Figure 2).

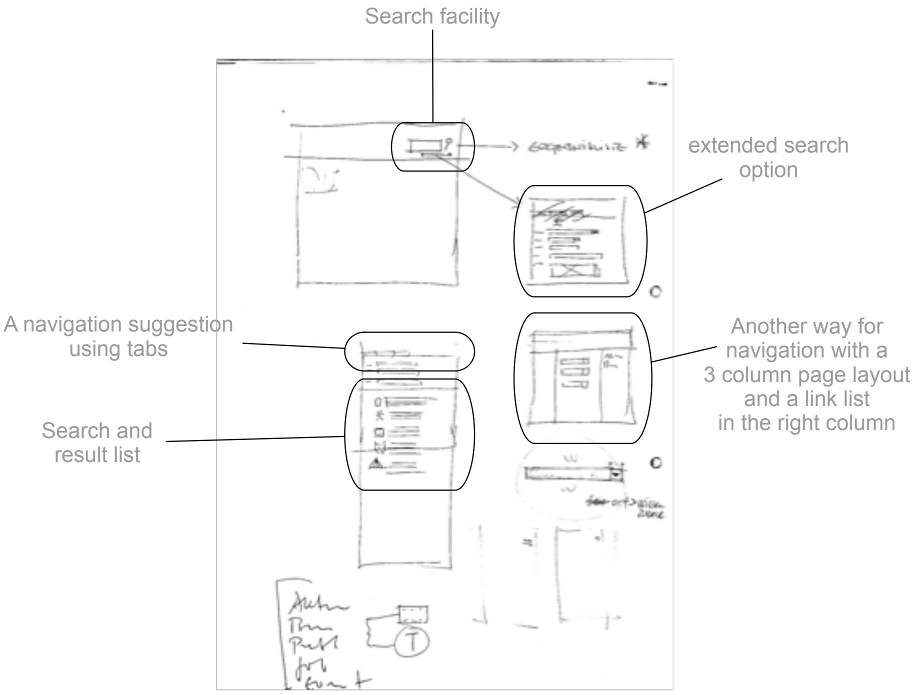

Figure 2: Linking and navigation designed for the web site by *Archcom*.

Equally detailed and thorough are the descriptions of the data for the persistence layer of the system, which are discussed before integrating the database components (Figure 3).

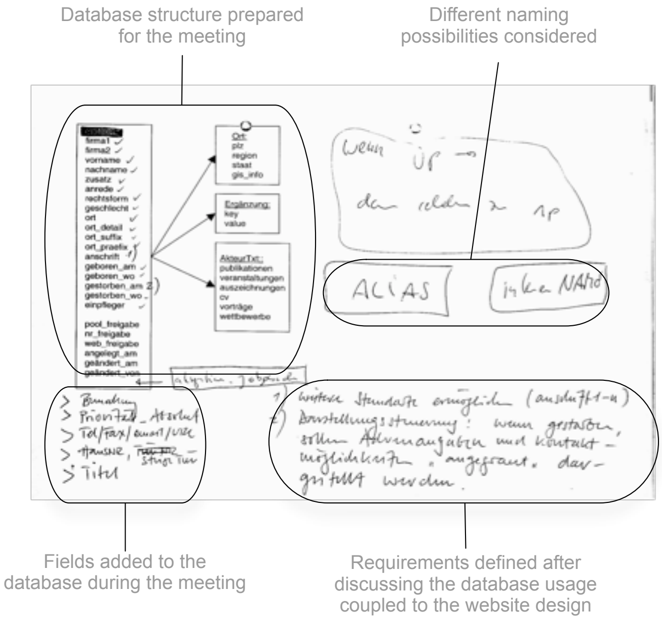

Figure 3: The database model with additional data attached after a discussion in a meeting (*Archcom*).

Besides the multi layered sketches of the database, the final data structure is represented by a UML class diagram. UML is also used for modeling the system and to describe the use cases. The team considers representations of the system-in-development with UML particularly useful in convincing the customer of the technical design and of its quality. The team member responsible for the technical concept tries to follow a ‘best practice example’ in software engineering. He applies coding conventions and tries to keep the code simple, readable, understandable, and easy to maintain. At any possibility, he shows the insights of his code and models to convince others of his professional approach to design and development.

Design representations including the layout and functionality issues are prepared in different media: sketches on paper, computer-based drawings and lists, storyboards, videos and HTML prototypes. In meetings these are used to communicate the design-in-progress, considering constraints, such as deadlines, available resources, and user requirements (Figure 4).

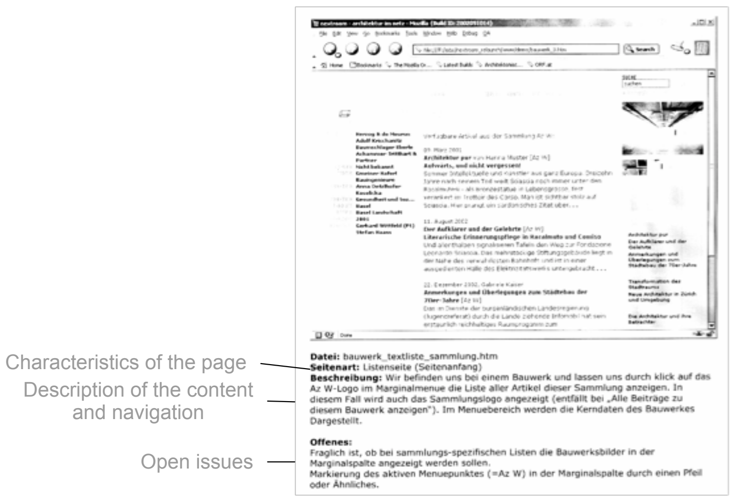

Figure 4: A page printout of the HTML prototype with a description of design steps and open questions added on bottom of the page (*Archcom*).

Printouts of the website (representing a part of the HTML prototype) are created as part of the design process and annotated with a short description and open questions. These composite visualizations represent a design step together with all the issues that need to be resolved in a way that is visible and understandable to all. The collection of printouts serves as an important base for planning of further activities in the project.

As has also been observed by Newman and Landay (2000), team members use all these artifacts simultaneously, switching between formats and media with ease. For the presentation of the whole system to the customer, a HTML prototype is built together with a video movie illustrating the use of the system.

Analysis and Discussion of Representational Artifacts at *Archcom*

Multimedia production performed by *Archcom* combines design with web engineering. Behind the goal of creating attractive multimedia interfaces for WWW, highly technological engineering decisions are made in an interdisciplinary team. Decisions are embedded in processes that are carried out very professionally to be examples of 'best practice'.

The to-do list used at *Archcom* (Figure 1) is, on the one hand, the personal work file of the project manager having it readily at hand in project meetings, and on the other, the coordinative artifact used in the project. It is the primary artifact of the manager, which makes it to the what-artifact showing coordination issues in the whole project. It partly contains the evolution of the issues listed, and is then a where-to artifact. It is material, because it represents the contributions of actors and history of past work, e.g., by status marks, due dates, names of responsible persons attached to single issues and single sub-tasks listed by the issues. As a common artifact the to-do list makes coordination arrangements in the team visible. Acting as a token, it is a resource for action by facilitating implicit communication between project members. At the same time, if it is added marks like done, cancelled, or postponed with a specific due date, it becomes a record of work and articulates the state of the work in the group, and so synchronizes the coordination within the team.

Coordination of the design and production of the multimedia platform is mainly carried out by the manager. He uses his to-do list that he shares with the other team members to define (sometimes synchronous) tasks and dependencies in work activities before they collaboratively assign team members and deadlines to them. This shared artifact is then referred to by all in case of misunderstandings, troubles to keep the deadline, contingencies occur unexpectedly and call for improvisations, etc. The to-do list shows at the same time the status of the work progress in the project. It is not only triggering actions, it keeps also track of work done, cancelled, or postponed. It grows through the project and helps remember decisions made, alternatives considered, and actions taken. Without a to-do list *Archcom* would not be able to coordinate and document its projects.

Detailed sketches of layout and navigation (Figure 2) of the web pages are what-artifacts crated cooperatively during project meetings at *Archcom*. Based on skills and experiences of team members, different suggestions are discussed to fulfill the requirements defined for the product. Ideas are visualized easily on a piece of paper, making their communication ad-hoc and simple for others. Alternatives can be represented one by one put next to each other to make variants available to all involved. Design ideas become more tangible for the team of designers, and this way a common understanding can be created for the decision of choice. Such a sketch can structure and guide the individual design work of team members by being a template and specifying the properties of the results of each.

Sketches support the creativity of the *Archcom* team as a whole. Each team member can present his design idea easily in form of a sketch usually in regular meetings. This helps not only understanding each other's solution to a current problem but also creating alternatives and makes them visible for decision-making. Ad-hoc exchange between team members is also made possible by sketches between the meetings, just to find a solution to a design problem cooperatively. As templates, sketches structure the work all by guiding the design of each member and providing a common approach to design in the team.

The database model (Figure 3) is a primary artifact of the database engineer. The objects in the database representing the primary what-artifacts used in the Internet forum are at the same time common objects enabling crossing boundaries between professionals, like the graphic designer, who designs the presentation of these objects in the web site, or the database engineer, who takes care of the storage, retrieval, and consistency of these objects at the backend, and the web programmer, who accesses database services to retrieve and display database objects at the frontend.

The database model is a typical boundary object enabling communication between different disciplines in the design team. The common problem how to make web data persistence becomes technical. This interface needs to be presented and negotiated between the database engineer and the graphic designer, because the implementation in the database can have impact on the graphic design and vice versa. The model is an abstract representation of a possible solution to agree on in the team. At the same time it protects the competence areas of different professionals because it does not show the complexity and the details of the implementation.

The page printout of the HTML prototype (Figure 4) has again a coordinative character like the to-do list, by grouping open issues by the structure of the web site pages. At the same time, it provides an overview of all data composed at one page, but managed and maintained separately in different system components. Such a page represents the integration of the data processed with different mechanisms in the same system. It is the only interface to the user and visible for evaluation and further negotiation. So, it supports data exchange between the design team and the customer by enabling implicit articulation of the work progress.

Multimedia producers use a variety of techniques, from sketches on paper to mock-ups, prototypes, and flow charts. Newman and Landay (2000) have described some of these artifacts in detail. High fidelity mock-ups for instance “contain(ed) images, icons, rich typography, and sophisticated color schemes, and these details of the visual presentation were meant to be taken literally” (p.266). This type of visualizations is typically the ones used to present the user interface of a product as they can be easily understood by the client. In *Archcom*, linking and navigation are usually presented as sketches on paper or in the form of site maps, as shown in Figure 4. These are “high level visualizations of site structure in which web pages or entire subsections of the site are represented by textual labels” (Newman and Landay 2000, p.268) and are created and readable only by professionals.

Chipcom: An Engineering Company

In a European STREP project called MAPPER[5] (Model-based Adaptive Product and Process Engineering) (IST-016527) we[6] investigated technological and methodological possibilities to support designers in collaborative engineering, which involved participative engineering methodologies to enable joint product and process design, interdisciplinary and inter-organizational collaboration throughout multiple product life cycles. We carried out an ethnographic study (Jordan, 1993) in one of the industrial partners we call *Chipcom*[7] in this paper. We visited *Chipcom* three times to observe participatory the workplaces, meetings, and design sessions, what we recorded as video and audio files. We interviewed in-depth some of the key actors. Additionally, we gathered and analyzed *Chimcom*'s artifacts like documents, source codes, test reports, simulation results, project plans, charts, organization diagrams, and models of different kinds. This qualitative study resulted among others in field visit and validation reports (Jacucci, 2005).

Chipcom provides IP (Intellectual Property) cores since 1997, VHDL and Verilog model development services as well as hardware and software development for microcontroller-based systems. In this paper we are referring to the production of the USB High Speed OTG Transceiver (PHY, physical layer) IP core with UTMI+ interface, shown in Figure 5. Analog block is implemented by the group A, digital by B, where as the prototype of the whole product is prepared by the group C. The arrows between analog and digital blocks represent the data flow between the components and by this the interfaces, which must be defined and agreed on by both groups A and B.

[5] http://mapper.eu.org/

[6] The project group consisted of Hilda Tellioğlu and Ina Wagner from Vienna University of Technology as well as of Gianni Jacucci and Gian Marco Campagnolo from University of Trento.

[7] The name is modified.

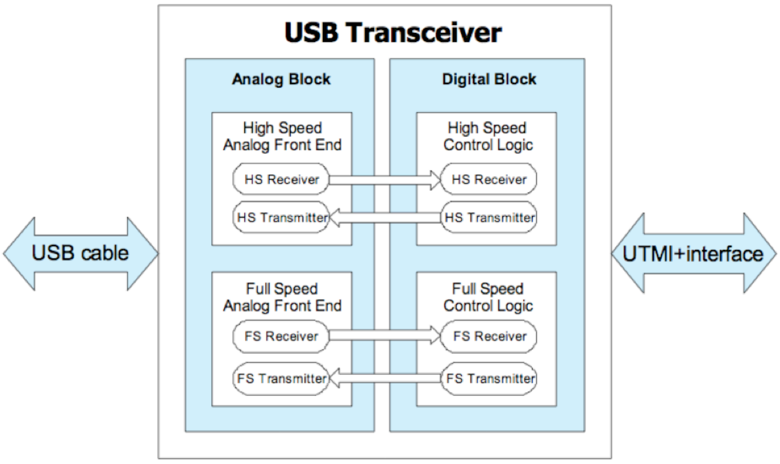

Figure 5: *Chipcom* presents the company's product and its parts, which at the same time represents implicitly for the insiders of the project the organizational structure around the production.

At *Chipcom* we could find other types of presentations as well. Several models are created to capture the as-is work processes and then to define the to-be processes. All three groups (A, B, C) generated their internal design workflows as Active Knowledge Models (AKMs) with IDEF (ICAM Definition Language) notation. Besides showing the interfaces to other workflows, this AKM-based workflow spanned over specification, development, verification, and product preparatory phases in all groups. It was a result of numerous consultations between managers and engineers of all groups. Next, actors responsible for particular design phases in the design flow were identified, technologies for components manufacturing and tools to be used were decided.

The design flow represented as a visual model (Figure 6) comprises sets of design tasks performed at these three geographical locations. The design object is a mixed-mode component. Design processes are split among the team members, taking the engineering competences into account, in such way that an analog design takes place at A, a digital design at B, and a board-level testing at C. This model represents details of the process needed to create the product USB Transceiver shown previously (Figure 5). "Design and Verification" is a part of a larger engineering process, starting with design objectives, and resulting in delivering the final product (Figure 6). After the process "Specification", which is the co-responsibility of the groups A and B, the process "Design and Verification" can be started. It contains ten tasks, which are connected to each other. Some tasks can be carried out parallel; some are sequential and depending on the outcome of a previous task. The model shows the logical and temporal dependencies between all tasks. Additionally it represents the assignment of tasks to the groups. Six of the tasks are assigned to B, four to A, marked by different colors. How the IT infrastructure is associated to the process "Design and Verification" is not presented in this model. It shows also a wide spectrum of other information related to the current shared product, namely information on: the internal organization of involved companies (e.g., company structure, locations, human resources, staff competence skills), the available IT infrastructure (e.g., design automation, administration, and office tools), the current project organization (e.g., project responsibilities), the detailed structure of the common product, and the project plan (e.g., management and design workflows). These details are not illustrated in the figure.

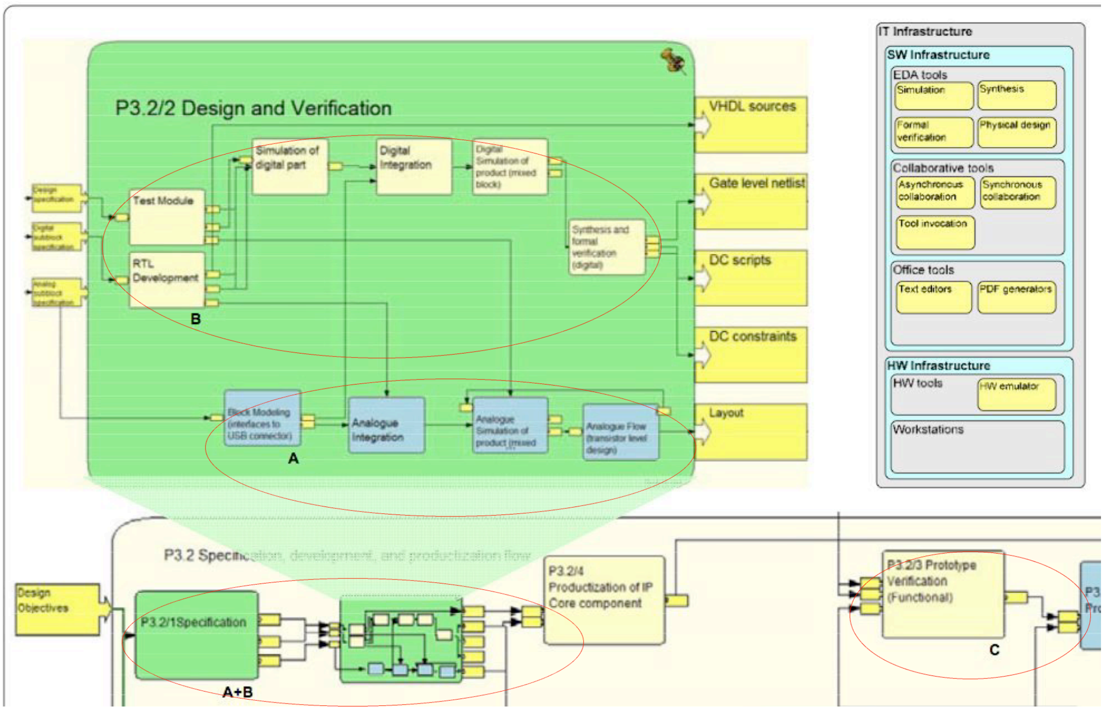

Figure 6: A view of a part of the design processes distributed into three locations (A, B, C): AKM in IDEF notation of "Design and Verification" process.

As one of the research results in MAPPER, *Chipcom* has installed a system called TRMS (Tool Registration and Management Services) to support distributed design. The workflow used represents the involvement of different persons working on distributed teams implicitly. Multiple users can access TRMS at the same time, and long jobs can be invoked and reported. A sequential workflow is integrated by automating some steps in simulation and evaluation. The aim was to support distributed design between the location A, B, and C.

TRMS can be started by anyone remotely, which displays then the different steps in the workflow in their sequential order (Figure 7). If a task has been successfully finished, its status is visualized green and a textual report is displayed in the appropriate task tab. If there was an error, this is marked red to get the attention of the engineers. Also a detailed list of execution steps is provided to follow the process invoked by that erroneous task.

A scenario illustrates how TRMS is used between different teams. One of the engineers at B working at the USB Transceiver performs a synthesis with support of TRMS. After the synthesis has been completed, TRMS sends the results to the CVS repository, which is also accessible from C, where the engineer at C can download the file and start FPGA prototyping. One of the advantages is that the synthesis can be done on one machine (with one single license); the other benefit is that if the engineer at C finds a small error, he can fix it on his computer and invoke the synthesis again remotely.

The engineer at B has found a bug, which he needs to correct in the source file. He updates the actual source file locally, uploads the modification to the repository, updates the file in TRMS, and starts the synthesis again.

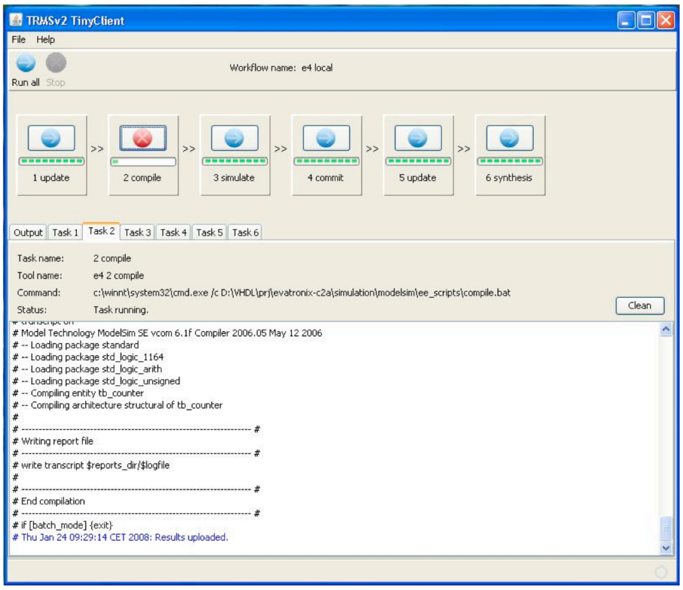

Figure 7: The TRMS workflow system used at *Chipcom*.

Different artifacts are created in this process: log files from the synthesis, where engineers can look for warnings; the result, which is a binary file in MCS format, used for FPGA prototyping, with the software only being available at C; and the scripts, which are evoked by TRMS, which are shared with A in the form of a Netlist.

When the synthesis has been completed, which is hard to predict, the engineer at B switches to the internal chat forum GaduGadu to inform his colleague at C that the update has been completed. He does not chat in the system attached to TRMS “because it does not work".

The engineer at B, who has used TRMS for a few weeks for hardware verification together with his colleague at C, comments: “This is a good thing that everything is automated. There is no possibility to run the wrong script, you are sure that the actual sources are used, you don’t have to think about which file to update, it is simplified now, TRMS does the CVS update”.

TRMS does not provide awareness information like who else is currently accessing it or any notification when a task or workflow is completed. The feedback delivered about the state of the synthesis is in most cases not sufficient. The processes invoked by different actors are not synchronized. Engineers must synchronize their invocations manually.

Analysis and Discussion of Representational Artifacts at *Chipcom*

The visualization in the Figure 5 shows a model of the product USB Transceiver, what different teams create cooperatively. This is an image presented inside and outside the company. It is used not only to communicate the product's components within the company between cooperating branches, but also to represent the product to customers, external partners, and other potential consumers or supporters. At the same time it is a representation of the actual distribution of work and interfaces between cooperating teams. One can see easily the areas of responsibilities and who to blame if there are errors or feature lacks. It shows also, no matter who is responsible for the single components of the product, the product must provide two interfaces, one via a USB cable and one via UTMI. This is a concern of all groups participating in the design of this device. In case of interface definition or enhancements to the product, this representation helps talking about possibilities, alternatives for solutions, and impacts among components.

Besides the practical and work-related issues, this representation of the USB Transceiver (Figure 5) serves as a tertiary artifact illustrating the vision of a transceiver implemented by *Chipcom*. It is a goal, a product, which has not been implemented yet, a challenge, embedding an evolution of a new product. At the same time, it is common, which makes it also a where-to artifact used by all three groups, to arrange their work by keeping interfaces to others transparent and visible all the time. The visionary character of this product representation supports focusing on the product by providing a common artifact to refer to.

The AKM model of work processes, including single tasks in their relation to each other and showing human and non-human resources, is the major secondary artifact shaping work arrangements around the product (Figure 6). Not only in planning processes to produce a product but also in reorganizing existing design and production for collaboration can be modeled and communicated in such a secondary artifact. It is created cooperatively and shared among groups involved. It can also be used to identify problems in the process, like deadlocks, lack of integration or interfaces, not considered interdependencies between tasks and subtasks and furthermore to interfere in processes by, e.g., assigning other persons or groups to certain tasks or moving non human resources to support the performance of certain activities. Besides facilitating orientation, the process and resource model in AKM format engages all project members, who have access to it, just by making them aware of work processes planned, optimized, or rearranged. It makes process management transparent for all, which invites people to think about it and perhaps to make suggestions for improvement.

Having a common workflow system implemented in TRMS (Figure 7) makes it for all engineers at all locations possible to run and debug others' code. The code becomes a primary artifact of the person using it. This makes collaborative engineering possible, especially by communicating explicitly via other channels like instant messaging. So, TRMS is a common artifact. It is a well-accepted tool, which standardize and automate scripts for synthesis or compiling, linking and versioning. Unfortunately, it was not coordinative at the time of our study. All coordinative features were replaced by the engineers with additional tools.

TRMS impacts the work processes at *Chipcom*. As a secondary artifact, it shows a model of engineering work in temporal and logical order. Furthermore, it forces engineers to run scripts in a very certain predefined way. It not only shows a map how to run, compile, and link code simultaneously, it also acts as a script. By means of TRMS, it is possible to access the code of others and debug and upload them to the common server. However, due to ownership reasons no one tries to fix a bug in the code of a colleague. The convention established at *Chipcom* is to report or discuss bugs found in others' code. Fixing it is the responsibility of the owner of the code. Even the technologies make the handling otherwise possible, the norms and work protocols agreed upon established in the company determine how to act and setup the constraints for work processes. This shows that artifacts and the technology cannot determine how to act at work places. At the end, it is the human who decides what to do and what to avoid.

So, synchronization, awareness, and configurability are features needed by engineers at *Chipcom*, which is barely provided by the system used. To overcome this gap, engineers use texting to communicate with each other, for notification but also for exchange of pieces of code or makefiles to decide how to change them. Not only in the process of simulation, also in design and development phases, engineers at *Chipcom* are used to contact their colleagues of other branches to ask them questions about coding (if there is a special case, which must be considered by all), compiling (the exact compiler options), or debugging (when they debug each others' code) issues in developing electronic devices.

At *Chipcom*, a kind of representation to show the state of a task across the organization is needed. Erickson et al. called this "task proxy", who not only considered visualization on technical level, but also focused on task information in a way intended to support social inference (2004). Making task proxy visible to participants causes emerging of social dynamics, like peer pressure or proactively helping those who need support. This type of visualization of the task proxy and through this the social proxy is one of the examples showing the interest on social visualization in the literature. Shared representations and body lists, profiles, and changes of status of people as we are familiar with in social software are further developments of social visualizations playing a very important role in social interaction and collaboration in teams. Halverson et al. (2006) explored this in change management.

Conclusions

In this paper, we presented first a trajectory of artifacts by focusing on their representational role in cooperative design-based engineering. After illustrating two cases from real work environments we described some of important artifacts we could identify in our ethnographic studies. Using the concept of artifacts we analyzed the artifacts used in these cases by stressing their types, roles in individual and collaborative design work, and impacts on work practices. We mainly focused on their representational role as visualizations of different kinds. Henderson (1995) looks at visualizations as 'network-organizing devices'. They support individual and cooperative thinking and organizing. Visualizations help to keep the design concept present in teams and to coordinate the work around it. They are used to illustrate the design ideas to different actors involved in the project, such as the clients, external professionals, or partners, convincing them of the design idea and mobilizing their cooperation (Schmidt and Wagner 2002).

In their study of designers in a software company, Muller and Carey describe all these visualization strategies and tools as designers' artifacts and typical of their techniques of creating representations of the various aspects of their work and communicating them. They identified prototypes (which may take a number of forms) and graphics as one of the strongest media, arguing: "Some highly influential designers use very low-technology prototyping tools, such as basic painting programs. Others dig deeply into what appears to be simple business software, reinventing presentation systems into elaborate prototyping environments. Still others develop competences that are very similar to those of programmers, albeit in environments that support rapid idea expression rather than production software performance" (2002, p.387). This is an example of style that is representative of a particular occupational community, which members preserve and culture even if they work in other contexts, e.g., as a minority discipline in a software company.

In this paper, we showed how representational artifacts could be identified and analyzed in cooperative engineering environments. Studying real work environments by means of ethnographic methods provides rich empirical material, which is not easy to analyze, compare, and use to develop design implications. Applying the corpus of different types of artifacts to ethnographic study material results in better understanding of work processes, tasks, and artifacts used, and furthermore cooperation practices between actors and groups. This is a good start to design or redesign systems, to introduce new artifacts of any kind to support work processes.

Acknowledgements

We are grateful to project partners for their cooperation in MAPPER, especially to our users at *Chipcom*. Many thanks to the company *Archcom*, where I could spend a lot of time with observing their meetings and design sessions.

References

Ackerman, M. S. (1997). Augmenting the organizational memory: A field study of Answer Garden. In T. Malone, (Ed.), *CSCW'94: Proceedings of the Conference on Computer-Supported Cooperative Work* (pp. 243-252). Chapel Hill, North Carolina, USA: ACM Press.

Ackerman, M. S., & Malone, T. W. (1990). Plans and situated actions. The problem of human-machine communication. *Proceedings of the Conference on Office Information Systems*. Cambridge, Massachusset: ACM Press, 31-49.

Bannon, L. (1997). Constructing common information spaces. *Proceedings of the Fifth European Conference on Computer Supported Cooperative Work*, 81-96.

Bardram, J. E. (1997). Plans as situated action: An activity theory approach to workflow systems. In *Proceedings of the Fifth European Conference on Computer Supported Cooperative Work*, September 7-11: 17–32.

Bardram, J. E., & Bossen, C. (2005). A web of coordinative artifacts: Collaborative work at a hospital ward. *GROUP'05*, Sanibel Island, Florida, USA, 168-176.

Berlage, T., & Sohlenkamp, M. (1999). Visualizing common artefacts to support awareness in computer-mediated cooperation. *Computer Supported Cooperative Work, 8*: 207-238.

Bowker, G. C., & Star, S. L. (1999). *Sorting Things Out: Classification and Its Consequences*. Cambridge, Massachussetts, USA: MIT Press.

Carstensen, P. (1996). *Computer Supported Coordination*. Risø National Laboratory, Roskilde, Denmark.

Cole, I. (1982). Human aspects of office filing: Implications for the electronic office. *Proceedings of the Human Factors Society 26th Annual Meeting*. Santa Monica, California, USA: 59-63.

Collins, P., Shukla, S., & Redmiles, D. (2002). Activity theory and system design: A view from the trenches. *Computer Supported Cooperative Work: The Journal of Collaborative Computing, 11*(1-2): 55-80.

Conklin, J. (1993). Capturing organizational knowledge. In *Readings in Groupware and Computer-Supported Cooperative Work: Assisting Human-Human Collaboration*. San Mateo, California, USA: Morgan Kaufmann Publishers: 561-565.

Conklin, J. (1989). Design rationale and maintainability. In B. Shriver (Ed.), *Proceedings of 22nd Annual Hawaii International Conference on System Sciences Vol. II* (pp. 533-539). IEEE Computer Society.

Conklin, J., & Begeman, M. L. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *CSCW'88: Proceedings of the Conference on Computer-Supported Cooperative Work*. Portland, Oregon, USA: ACM Press, 140-152.

Divitini, M., Simone, C., & Schmidt, K. (1996). ABACO: Coordination mechanisms in a multi-agent perspective. *COOP'96: Second International Conference on the Design of Cooperative Systems*, Antibes-Juan- les-Pins, France, INRIA Sophia Antipolis: 103-122.

Engeström, Y. (1990). When is a tool? Multiple meanings of artifacts in human activity. In *Learning, Working and Imagining: Twelve Studies in Activity Theory* (pp. 171-195). Helsinki: Orienta-Konsultit Oy.

Engeström, Y. (1999). Innovative learning in work teams: Analyzing cycles of knowledge creation in practice. In *Perspectives on Activity Theory* (pp. 377-404). Cambridge: Cambridge University Press.

Erickson, T., Huang, W., Danis, C., & Kellogg, W.A. (2004). A social proxy for distributed tasks: Design and evaluation of a prototype. *Proceedings of the CHI 2004*, ACM Press: 559-566.

Gerson, E. M., & Star, S. L. (1986). Analyzing due process in the work place. *ACM Transactions on Office Information Systems 4*(3), 257-270.

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Boston, USA: Houghton-Mifflin.

Grinter, R. E. (1996). Supporting articulation work using software configuration management systems. *Computer Supported Cooperative Work. The Journal of Collaborative Computing 5* (4), 447-465.

Guy, E. S. (2003). Patterns as artifacts in user-developer collaborative design. *Proceedings of ECSCW 2003*.

Haas, C. (1996). *Writing Technology: Studies on the Materiality of Literacy*. Mahwah, New Jersey, USA: Lawrence Erlbaum.

Halverson, C. A., Ellis, J. B., Danis, C., & Kellogg, W. A. (2006). Designing task visualizations to support the coordination of work in software development. *Proceedings of the 20th Anniversary Conference on Computer Supported Cooperative Work.* New York, NY: ACM Press, 39-48.

Henderson, K. (1999). *On line and on paper: visual representations, visual culture, and computer graphics in design engineering*, MIT Press.

Hertzum, M. (1993). Information retrieval in a work setting: A case study of the documentation part of chemists' work. In J. P. Bansler et al. (Eds.), *Proceedings of the 16th IRIS In- formation Systems Research Seminar in Scandinavia* (pp. 786-798). Copenhagen, Denmark: DIKU Report 93/16, University of Copenhagen.

Hertzum, M. (1999). Six roles of documents in professional's work. In S. Bœdker, M. Kyng, & K. Schmidt (Eds.), *Proceedings of the Sixth European Conference on Computer-Supported Cooperative Work* (pp. 41-60). Netherland: Kluwer Academic Publishers.

Khandani, S. (2005). *Engineering Design Process. Education transfer plan*.

Kuutti, K. (1996). Activity Theory as Potential Framework for Human-Computer Interaction Research. In *Activity Theory and Human-Computer Interaction* (pp. 17-44). Cambridge, Massachusetts, USA: The MIT Press.

Latour, B. (1986). Visualization and cognition: Thinking with eyes and hands. *Knowledge and Society: Studies in the Sociology of Culture Past and Present*: 1-40.

Leont'ev, A. N. (1981). *Problems of the Development of the Mind*. Moscow: Progress Publishers.

Lundberg, N., & Sandahl, T. I. (1998). Artifacts in work practice. *IRIS 1998.*

Lutters, W. G., & Ackerman, M. S. (2002). Achieving safety: A field study of boundary objects in aircraft technical support. *CSCW'2002: ACM Conference on Computer Supported Cooperative Work*. New Orleans, Louisiana, USA: ACM Press: 266-275.

Muller, M., & Carey, K. (2002). Design as a Minority Discipline in a Software Company: Toward Requirements for a Community of Practice. *Proceedings of CHI 2002*, Minneapolis, Minnesota, USA.

Nardi, B. A. (1993). *A Small Matter of Programming, Perspectives on End User Computing*. Cambridge, Massachusetts, London, England: MIT Press.

Newman, M. W., & Landay, J. A. (2000). Sitemaps, Story Boards, and Specifications: A Sketch of Web Site Design Practice, *Proceedings of DIS'00*, Brooklyn, New York: ACM Press.

Pahl, G., Beitz, W., Feldhusen, J., & Grote, K-H. (2007). *Engineering Design: A Systematic Approach*, Springer, 3rd Edition.

Schmidt, K. (2001). Classification schemes as material practice. *Proceedings of the Second CISCPH workshops on Cooperative Organization of Common Information Spaces*, IT University of Copenhagen, Denmark.

Schmidt, K., & Bannon, L. (1992). Taking CSCW seriously. Supporting articulation work. *Computer Supported Cooperative Work (CSCW) 1*: 7-40.

Schmidt, K., & Simone, C. (1996). Coordination mechanisms: Towards a conceptual foundation of CSCW systems design. *Computer Supported Cooperative Work 5* (2-3): 155-200.

Schmidt, K., & Wagner, I. (2002). Coordinative Artefacts in Architectural Practice. In M. Blay-Fornarino et al. (Eds.), *Cooperative Systems Design: A Challenge of the Mobility Age. Proceedings of the Fifth International Conference on the Design of Cooperative Systems (COOP 2002)* (pp. 257-274). Amsterdam et al.: IOS Press.

Schmidt, K., & Wagner, I. (2004). Ordering systems. Coordinative practices and artifacts in architectural design and planning. *OrdSys Journal of Computer Supported Cooperative Work*.

Sellen, A., & Harper, R. H. R. (2001). *The Myth of the Paperless Office*. Cambridge, Mass: MIT Press.

Simone, C., & Divitini, M. (1997). Ariadne: Supporting coordination through a flexible use of the knowledge on work processes. *Journal of Universal Computer Science 3* (8): 865-898.

Star, S. L. (1989). The structure of ill-structured solutions: Boundary objects and heterogeneous distributed problem solving. In *Distributed Artificial Intelligence* (pp. 37–54). Vol. 2. London, UK: Pitman.

Star, S. L., & Griesemer, J. R. (1989). Institutional ecology, ‘translations’ and boundary objects: Amateurs and professionals in Berkeley’s museum of vertebrate zoology, 1907-39. *Social Studies of Science 19*: 387-420.

Suchman, L. (1987). *Plans and Situated Actions. The problem of human-machine communication*. Cambridge et al.: Cambridge University Press.

Tellioğlu, H. (2010a). Coordination 2.0. Using Web-based Technologies for Coordination Support. *Proceedings of the International Workshop on Web Intelligence and Virtual Enterprises 2 (WIVE 2010)*, Saint-Etienne, France.

Tellioğlu, H. (2010b). Coordination of Work by Using To-do Lists: Practices and Requirements. *Proceedings of the International Conference on Information Technologies (InfoTech-2010)*, St. Constantine and Elena, Bulgaria.

Tellioğlu, H. (2009). Model-Based Collaborative Design in Engineering. In L. Yuhua (Ed.), *Lecture Notes in Computer Science, Information Systems and Applications. Proceedings of the 6th International Conference CDVE 2009* (pp. 85-92), Vol. 5738.

Tellioğlu, H. (2007). *Cooperative Work*. Habilitationsschrift, Technische Universität Wien, Fakultät für Informatik, Wien.

Tellioğlu, H., & Wagner, I. (2005). Work cultures in multimedia production. *Proceedings of the ALP_IS Alpine Information Systems Seminar*: 143-161.

Tellioğlu, H. (2003). Modeling Coordinated Work: Definition and Application of the Model “Coordinated Work Environment”, *The Journal of Supercomputing. An International Journal of High-Performance Computer Design, Analysis, and Use 24*(2): 161-171.

Wagner, I. (2000). Persuasive artefacts in architectural design and planning. *Proceedings of CoDesigning 2000*: 379-390.

Keywords: Artifacts, representations, design, engineering, ethnography, coordination, workflow.