Putting Objects to Work: Hypermedia as the Subject Matter and the Medium for Computer-Supported Cooperative Work

Norbert A. Streitz

GMD-IPSI
Integrated Publication and Information Systems Institute (IPSI)
German National Research Center for Computer Science (GMD)
Dolivostr. 15
D – 64293 Darmstadt, Germany
Email: streitz@darmstadt.gmd.de

Abstract of invited talk. One can observe the following parallel developments: an increasing merge of computer, network and telecommunication technology, new needs and markets in the information and media industry, indications of changes in the way people use information in their work and in their home environments. A common factor is the digitalization of information at the time it is processed or – resulting in more possibilities – when it is created. But the progress in networks and basic technology is not paralleled to the same degree by advances in the development of corresponding applications which – in the end – are necessary to justify the immense investments, e.g. in information super highways. One important class of applications is support for the cooperation of spatially distributed people working with shared information objects. We propose that "hypermedia" serve not only as the "subject matter" of cooperation but also as a "medium" for coordination, communication, and cooperation by using specific object types and exploiting their properties. In order to provide examples of how hypermedia can support telecooperation, we will present the design and implementation of two group aware applications – SEPIA and DOLPHIN – which were developed at GMD-IPSI.

1 Introduction

One can observe the following parallel developments: an increasing merge of computer, network and telecommunication technology, new needs and markets in the information and media industry, indications of changes in the way people use information in their work and in their home environments. A common factor is the digitalization of information at the time it is processed or – resulting in even more possibilities – when it is created. But the progress in networks and basic technology is not paralleled to the same degree by advances in the development of corresponding applications which – in the end – are necessary to justify the immense investments, e.g. in information super highways.

One important class of applications is the area of synchronous and asynchronous cooperation of geographically distributed people who although in remote locations want to work with information objects to be shared in various ways. This requires support for "telecooperation" complemented by audio and video communication channels providing "telepresence". In order to facilitate cooperative work over distances using computers, an appropriate application framework and a corresponding system architecture for distributed applications is needed. First, the approach has to accommodate the specific requirements deriving from support for cooperative work of distributed users. Second, the approach must be able to provide multiple ways of handling information and this in a very flexible fashion with respect to structural as well as media-type properties.

We propose that an object-oriented and hypermedia-based approach has the potential to provide answers meeting these general requirements which will be more detailed in section 2. *Hypermedia* are to be considered as "networked multimedia documents" in the following two interpretations. Hypermedia consist of networks of information objects represented by typed nodes and links with multimedia content. At the same time, this collection of information objects can also be geographically distributed in many locations and the connections are made over a physical (e.g., fiber-based) or radio-based network. Beyond this, in our specific approach, hypermedia serve not only as a means to represent the "subject matter" when handling information, e.g., planning, editing, reviewing, and revising a large and highly structured document, but also as a "medium" for the processes of coordination, communication, and cooperation of a group of people by using specific object types and exploiting their properties.

In order to provide examples of how one can support telecooperation, we will present in section 3 the design and implementation of two group aware applications which were developed using an object-oriented approach: SEPIA – a cooperative hypermedia authoring environment for remote collaboration, and DOLPHIN – an electronic meeting room support system using an interactive electronic whiteboard coupled with networked local notebook computers and remote desktop workstations at the same time.

2 Next Generation Information Systems: Flexible and Situation-Aware Cooperative Hypermedia Systems

In order to meet the requirements of future applications, e.g., to be used on informations highways, hypermedia systems have to scale up on various dimensions. The traditional view of hypertext and hypermedia serves as a good starting point. It is closely related to — by some authors even identified with — the simple but powerful idea of providing freedom to associate and to relate information objects to each other. On this view, hypertext provides opportunities to deviate from linear, prescribed structures and to use more informal structures in communicating information [9]. For authors, hypertext systems provide a large degree of flexibility for connecting pieces of information and presenting it as an assembled collection in an information network. For readers, hypertext provides tools for navigating in these information networks and for exploring them freely. This way of presenting and exploring information was considered a new and promising development, especially contrasted with traditional information retrieval paradigms with their highly structured storage requirements and their formal query languages. It was soon realized, however, that the hypertext approach had problems as well: "getting lost in (hyper)space", "cognitive overhead" [1],, and "premature organization" [6] [7] are examples.

This first phase was followed by a time of providing additional information to the reader/user by expressing more "semantics" via the notion of typed nodes and links. Representing more semantic knowledge explicitly, allowed the system also to process the information in ways not possible before. At the same time, it related hypertext structures more to knowledge representations as they are being developed in the AI-community. But the increasing potential of expressing a more detailed and varied view of the content domain led – in some cases – also to more cognitive overhead. As a result, the two phases, i.e., first emphasizung very informal and less structured information and, second, presenting and communicating more detailed structures were followed by new proposals arguing again for less formal approaches and raising, e.g., the issue of graphical layout as a means of expressing relationships instead of typed links [8]. Thus, the next generation of hypermedia systems has to address this range of different requirements. In subsection 2.2, it will be discussed how to reconcile the different possibilities by arguing for more flexible systems.

In parallel to the issue of how much structure to provide and in which way, another issue was raised in the discussion of important properties of hypermedia systems. It became obvious very soon that large and complex hyperdocuments, e.g., technical documentation or manuals for aircrafts cannot and are not created by a single author but usually by a team of technical writers, or the documents created for planning, identifying, and deciding on new products and markets for a company are the result of the cooperation of a business team assigned to this task. Thus, the issue of how to support the cooperation and coordination of groups became critical when discussing design requirements for a new generation of hypermedia systems. In subsection 2.1, this will be translated in our requirement of situation-aware cooperative systems.

2.1 Requirements for *Situation Aware* Cooperative Systems

Investigating samples and different conditions of cooperative work shows that the cooperation of people can take various forms. Accordingly, a number of classification schemes have been proposed [2]. The most obvious dimensions are time of cooperation (same, different) and location of cooperation (same, different). This results in the well-known matrix of four conditions for different cooperation situations (see fig.1). The following instances provide examples for the different combinations. Situation 1 is the standard face-to-face meeting while a phone conversation between people working in two different cities is an everyday example for situation 2. Leaving a written note for a currently not present team member on the table in the jointly shared office and getting a comment back on the next day, is an example for situation 3. Sending a document by postal or electronic mail to another city at one point in time which is read some time later and answered subseqently are examples for situation 4. Research in computer-supported cooperative work (CSCW) investigates how the use of computers can help to facilitate various forms of cooperation. Although the variety of situations presented in fig.1 is widely acknowledged, most systems address only one or two combinations and consider them as more or less static situations. This restriction is a real shortcoming when it comes to real-life situations and task contexts where all the situations play a role.

On this basis, we argue for a new class of CSCW-systems which are not situation-restricted but are "*situation-aware*", i.e., they can accommodate the range of situa-

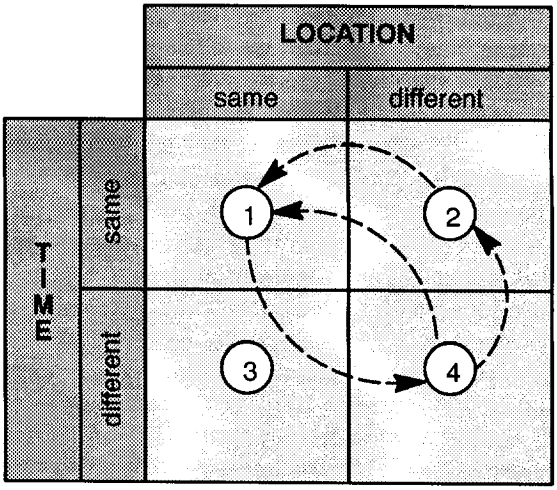

Fig. 1. Classification of cooperation situations representing an example for transitions between situations (see text for details)

tions of cooperative work. Differences caused, e.g., by the characteristics of synchronous or asynchronous work should not interfere but taken into account by smooth and seamless transitions between these situations in the course of the cooperation activity.

An example for this variety is the following sequence of events/ processes as part of a larger and more comprehensive cooperative effort represented in fig. 1. Of course, it has also to be accounted for that a number of processes are going on in parallel. The example includes the remote distribution of the agenda for a meeting, the individual preparation of the meeting by each participant, a synchronous coordination of, e.g., two members in remote offices before the meeting, the face-to-face meeting itself, the subsequent postprocessing of the meeting by each individual according to decisions and task assignments during the meeting, a follow-up meeting discussing the results of the individual work. This example includes transitions from situation 4 to 2 to 1 to 4 and returns to 1 for a follow-up meeting. It is this variety and not always predictable course of events and interactions which next generation CSCW-systems have to accommodate.

The overall goal is that there is one system meeting the requirements of each situation so that the users are not required to switch between different systems. This implies a set of requirements which includes tailored distribution lists with action items, joint workspaces shared between remote partners, group awareness in terms of dynamic indicators about who is doing what, reuse and annotations of material previously or concurrently created by other group members, versioning of objects within and between users, locking of objects in a multi-user database – to name a few. A more comprehensive discussion of specific requirements can be found in [4],[14].

Beyond this, a "*situation-aware*" CSCW-system should also be able to adapt to the critical conditions of the physical and social situation at hand. It should provide a range of structures appropriate for the identified group situation and task context and offer corresponding interaction mechanisms for creating, presenting, and revising content in formats compatible to the different situations. In a face-to-face meeting, for example, there should be the possibility to use the same software on a large interactive whiteboard operated with a cordless pen including gesture recognition while other members of the team interact with it via the mouse and keyboard of their personal computer situated on the meeting room table. Different social situations prompt also different types of material to be created: from handwritten scribbles to formal argumentation structures or organization charts. These situations translate into a set of requirements for "flexible" hypermedia systems (see section 2.2).

2.2 Requirements for *Flexible* Hypermedia Systems

We start from the observation-based assumption that people use a variety of ways to express their ideas and to communicate them to other people. The means of communication can vary depending on tasks, contexts, situations, and, of course, on individual differences and preferences. They range from very informal elements for structuring information (e.g. scribbles, sketches, handwriting, etc.) to very formal structures communicating explicitly properties and relationships of information objects. In the context of computer-based information systems, this raises the question of which role computers can play in these activities. Taking into account that differences in appearance may indicate differences in meaning, one has to ask "how much of the structure provided by a human can be interpreted and processed by the computer and how much can only be perceived and interpreted by another human ?"

Addressing this question, we have developed a classification of information structuring systems [5]. There are two dimensions:

- The degree to which a user exlicitly identifies object types to the system (**u** dimension).
- The degree to which the system represents object types internally (s dimension).

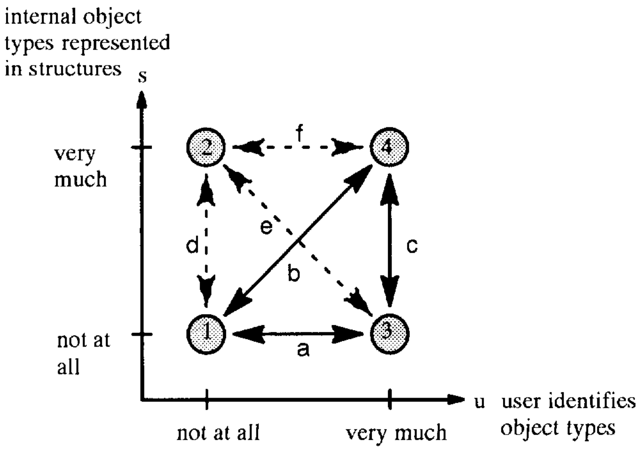

Fig. 2. Design space and possible transformations

Together, these dimensions span a design space of information structuring systems. Both dimensions have a scale ranging from "not at all" to "all". Since systems occupying the extreme points of "all" will be very rare, we indicate four points by using "not at all" and "very much" which are presented in fig. 2.

We have used this design space in order to locate existing hypertext systems. Because of the lack of space in this paper, we summarize from [5]. We found one cluster of systems in the neighborhood of area 4 and another one to the left of area 3 with some elevation on the s dimension. It is important to note that there are almost no examples covering more than one combination of our two dimensions. This raises the issue of how to provide possibilities of coexistence of different structures accommodated by the representational scope of one system. Assuming that systems can accommodate more than one type of structure, one has to ask the question if the system accommodates also transformations between these different types of structures as they are indicated by the arrows in fig.2. It is obvious from this discussion that there are still not yet explored "white areas" on this map of systems and it is our task to fill them with concrete implementations. The DOLPHIN system presented in subsection 3.2 provides some first answers to these open problems.

Summarizing these investigations [5], we arrived at the following general requirements for *flexible* hypermedia systems:

1) The system should provide objects with different degrees of formality to users. The objects should differ in their degree of expressiveness for explicit type information.
2) The user should be able to determine which degree of formality is needed in a given application situation.
3) It should be possible for objects with different degrees of formality to co-exist and be related to each other.
4) The system should be able to build up automatically a semantically valid internal representation of the information objects created by the users.
5) Users should be able to transform different types of structures into each other in a flexible way. The different types of transformations are indicated by a,b,.. in fig. 2. We come back to this figure when we explain the position of SEPIA and DOLPHIN with respect to this design space.

Flexible hypermedia systems provide the necessary prerequisites so that the following user activities can be performed and are under full control of the enduser: structuring and restructuring, aggregating and segmenting, annotating and modifying, relating and separating information objects. This includes also that there very few restrictions on the format of these information objects. They can range from informal (e.g., scribbles and sketches, handwritten text) to formal structures (e.g., argumentation schemata constructed from a collection of typed nodes and links) and allow multiple media types (text, graphics, pictures, sound, video).

3 Two Hypermedia Applications Providing Situation Awareness and Flexible Information Structures

In this section, we describe two examples of how to meet the requirements developed in the previous sections. The sequence of presentation – first SEPIA then DOLPHIN –

corresponds to the development of our thinking about these issues and the actual system development extending the scope of covering multiple cooperation situations and providing more flexibility by a wider range of available structures.

3.1 SEPIA: A Cooperative Hypermedia Authoring Environment

SEPIA provides a comprehensive authoring functionality for the creation and processing of complex hyperdocuments. Its design is based on a cognitive model of authoring [12]. This includes the planning of documents, developing argumentation structures, investigating personal archives, and creating the final structure of the hyperdocument from a rhetorical perspective. Support is facilitated by four corresponding 'activity spaces' which provide dedicated functionality in terms of task specific operations and objects. The four activity spaces are: planning space, content space, argumentation space, and rhetorical space. This implies a corresponding hypermedia data model with typed links and nodes, and composites for aggregations [11],[13].

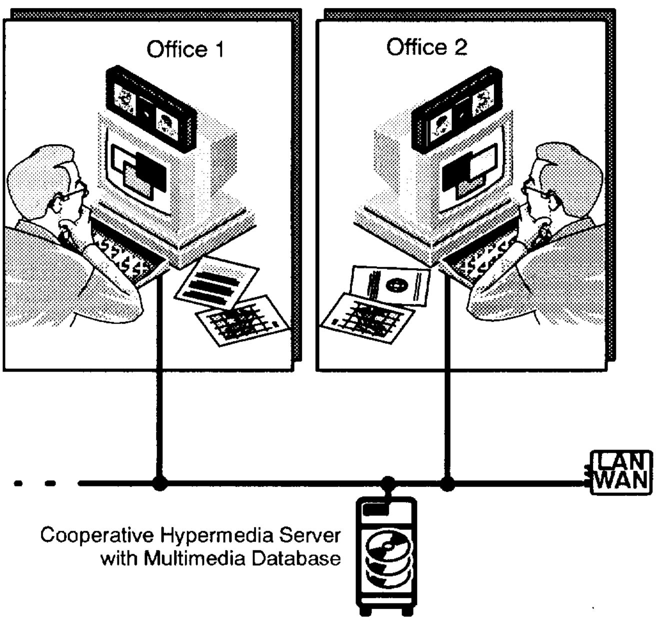

Fig. 3. Desktop-based collaboration with cooperative SEPIA

In order to support cooperative work of a group of authors, there is a persistent and shared data storage system allowing different kinds of databases as backends [10]. Support for cooperative work allows synchronous and asynchronous cooperation of physically distributed groups of authors by sharing objects in loosely and tightly coupled modes, smooth transitions between individual work phases and different degrees of coupling for collaborative activities. The system provides group awareness by indicators on who is working in which activity space, on which composite and even on which individual node or link within a collection of information objects [4]. In tightly coupled mode for very focussed cooperation of two or more participants, telepointers with user names allow pointing at objects and areas while at the same an audio and/or a video connection provides additional communication possibilities. Creating and cooperatively modifying objects is facilitated by the WYSIWIS (What You See Is What I See) property for synchronous cooperation displaying all actions, e.g. in a shared drawing tool WSCRAWL, on all screens of those group members which are currently in tightly coupled mode.

The system architecture of SEPIA is presented in fig. 4. The cooperative SEPIA clients are implemented using the object-oriented Smalltalk-80 environment, Unix, and OpenWindows on SUN Sparc-10 workstations. One version of CHS is implemented in C on top of the relational DBMS Sybase. There is also a version for other storage engines (e.g., the gdbm server). Current efforts include to use the object-oriented data base system Vodak also developed at IPSI. The audio communication fea-

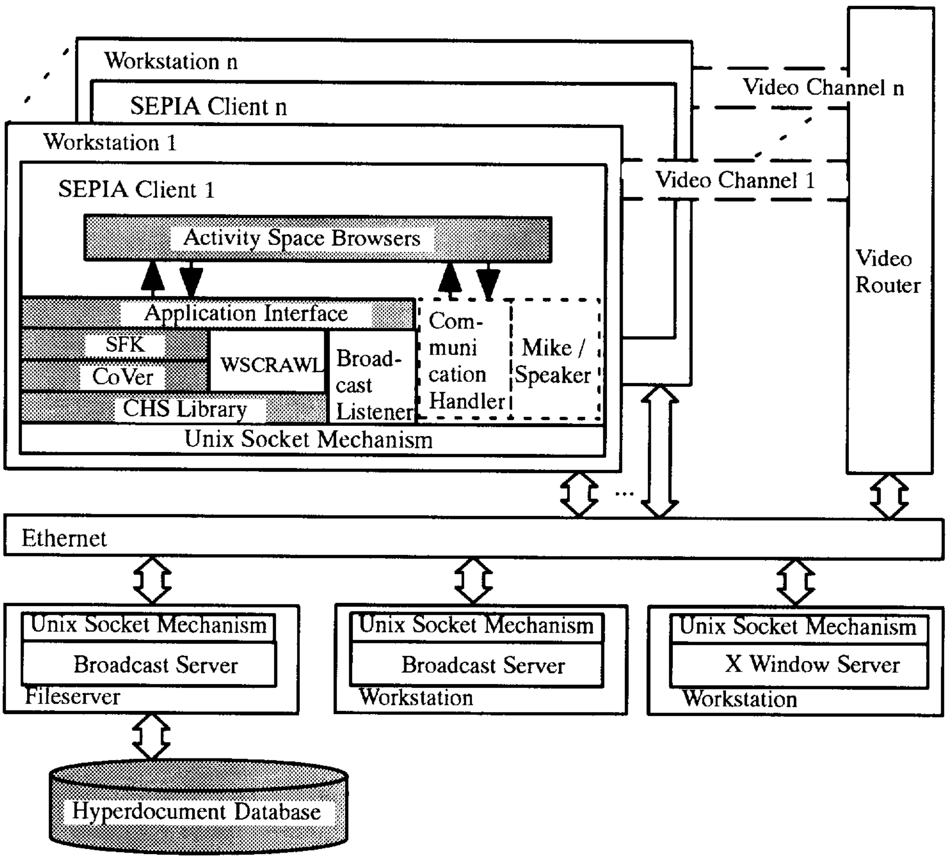

Fig. 4. System Architecture for Cooperative SEPIA

Shaded areas represent components which already existed in the SEPIA system for single authors. The communication handler and the mike / speaker processes are represented as dotted lines because they are created on demand only.

ture (using Netfone), the video server, and WSCRAWL, i.e., the integrated shared drawing tool are implemented in C. WSCRAWL uses an X server to broadcast information. Interprocess communication is implemented using standard UNIX sockets and TCP/IP.

3.2 DOLPHIN: An Electronic Meeting Room Support System

Putting our previous work and experiences with the cooperative hypermedia system SEPIA into a more global perspective, we came to the conclusion that we have to include and integrate support for face-to-face meeetings because they are an integral part of the overall cooperation activity of groups. Our new approach stresses therefore the fact that there is a range of collaboration situations which depend on each other and partially even exist in parallel. Considering evidence from observations of comprehensive real life applications, practical experience, and theoretical considerations, we built a new system called DOLPHIN in order to implement the above conclusions.

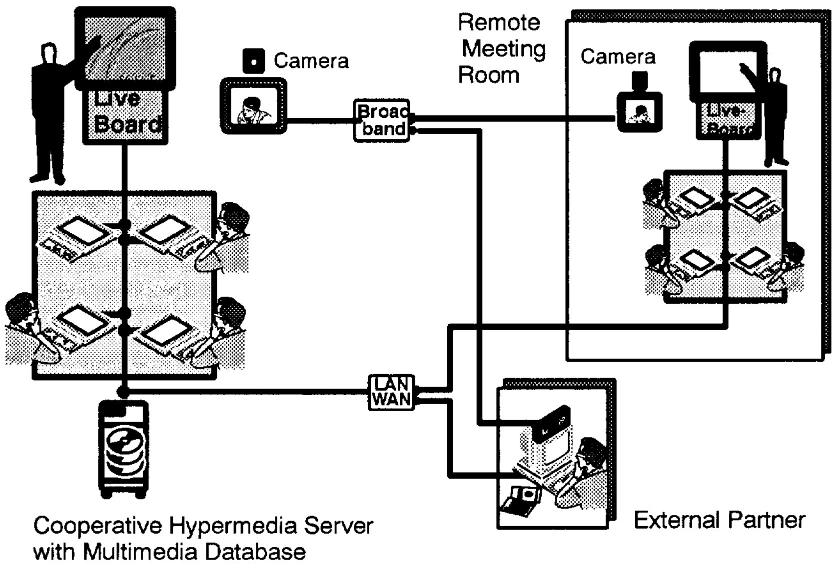

Fig. 5. Range of application scenarios currently supported by DOLPHIN

The basic scenario of DOLPHIN [14] assumes a group of meeting participants sitting around a table and a large interactive display in front of them. But the goal and the functionality of DOLPHIN are not restricted to this setting. The current implementation of DOLPHIN supports the following four application scenarios (see figure 5)

It can be used

1. as a pen-based single-user system on a Liveboard [3].
2. as a multi-user application shared between the Liveboard and a number of local networked personal computers situated in the meeting room (e.g. on the table)

3. as a multi-user application shared between the Liveboard, the local networked computers, and remote desktop-based workstations connected by Ethernet and/or FDDI.
4. as a multi-user application shared between one meeting room and a corresponding setup of Liveboard and computers in a second meeting room.

In order to support this range of application scenarios we had to address the following issues: content and structures used in meetings, interaction modes on large interactive displays, connectivity between public display and local computers, connectivity between meeting room and remote participants.

To provide as much freedom as possible to create different types of content / documents on the one hand and to create explicit relationships between them and between parts of them on the other, we adopted again a general hypermedia approach building on our experience with SEPIA. This implies that all objects and relationships are realized as hypermedia objects and links with varying degrees of complexity (e.g., layered and embedded composite objects) as well as multimedia content of all objects. Given the increased scope of requirements for DOLPHIN, the new nested hypermedia document model includes the following object types: nodes (objects consisting of other objects), content of nodes (there is always a compound content which can consist of all other object types), links (connecting nodes), scribbles (can have multiple forms: handwritten text, free hand drawn figures, etc.), text (as ascii objects), and images (pixmap drawing objects). In addition, we provided new interaction modes necessary for the large interactive display of the Liveboards. We developed our own gesture recognition software for a number of operations (e.g., creating, moving, opening, deleting of nodes, links) to be performed with the cordless pen which is also used for creating the scribble objects. It is important to note, that the different object types can be created in parallel in the same window where they coexist and that there are operations to transform the different structures into each other. In terms of support for cooperative work, DOLPHIN provides not only the features of SEPIA but supports additional concepts as, e.g., the notion of public and private workspaces (see [14] for more details).

4 Conclusions

We described two cooperative hypermedia systems which are both fully group aware applications providing a range of different structures to a group of users. Moreover, DOLPHIN is also a "situation-aware" cooperative system by adapting to the different situations and hardware plattforms it is used on. While SEPIA emphasizes more formal structures, DOLPHIN extends these by more informal structures. With respect to the design space presented in figure 2, SEPIA is located around area 4. DOLPHIN matches more of the requirements of "flexible" hypermedia systems and covers the areas 1, 3 and 4 supporting also the transformations a, b, and c. Although this is more than comparable systems have to offer, there is still progress to be made in reaching the overall goal of flexible and situation aware cooperative information systems.

Acknowledgements

The work and especially the systems presented here are the result of a cooperative effort in the division "Cooperative Hypermedia Systems" at GMD-IPSI. I would like to thank Jörg Haake, Jörg Geißler, Jeroen Hol, Thomas Knopik, Wolfgang Schuler, and Ajit Bapat for their valuable contributions to different parts of the work involved. Special thanks are also due to Boris Bokowski, Lutz Kirchner, Jan Schümmer, and Christian Schuckmann who helped to implement SEPIA and DOLPHIN. Finally, I like to thank Chris Neuwirth for her stimulating contributions to our work during her stay as a visiting professor at IPSI.

References

1. J. Conklin (1987). Hypertext: An introduction and survey. *Computer*, 20, 17-41.
2. C. Ellis, J Gibbs, G. Rein, (1991). Groupware – some issues and experiences. *Communications of the ACM*, Vol. 34 No.1 (January 1991), 38–58.
3. S. Elrod, et al. (1992). Liveboard: a large interactive display supporting group meetings, presentations and remote collaboration. *Proceedings of the CHI'92 Conference, Monterey, CA, USA (May 3–7, 1992)*, 599–607.
4. J. Haake & B. Wilson (1992). Supporting Collaborative Writing of Hyperdocuments in SEPIA. I*n Proceedings of the ACM 1992 Conference on Computer-Supported Cooperative Work (CSCW'92)* (pp. 138 – 146) Toronto, Ontario, November 1–4, 1992. New York: ACM Press.
5. J. Haake, C. Neuwirth, & N. Streitz (1994). *Coexistence and Transformations of Informal and Formal Structures: Requirements for More Flexible Hypermedia Systems*. GMD-IPSI Technical Report, April 1994.
6. F. Halasz (1988). Reflections on Notecards: Seven issues for the next generation of hypermedia systems. *Communications of the ACM*, 31, 7, 836 – 852.
7. F. Halasz (1991). *"Seven issues": Revisited.* Final Keynote Talk at the 3rd ACM Conference on Hypertext, San Antonio, TX, December 15 –18, 1991.
8. C. Marshall & A. Rogers (1992). Two years before the mist: Experiences with Aquanet. *Proceedings of the ACM European Conference on Hypertext (ECHT'92)*, Milano, Italy, Nov. 30 – Dec. 4, 1992, 53–62.
9. T. Nelson (1987). *Literary machines*. Edition 87.1
10. H. Schütt & J. Haake (1993). Server Support for Cooperative Hypermedia Environments. In H. Frei & P. Schäuble (eds.), *Hypermedia'93*. (pp. 45–56) Springer.
11. H. Schütt & N. Streitz (1990). *HyperBase*: A Hypermedia Engine Based on a Relational Database Management System. In A. Rizk, N. Streitz, & J. André (eds.), *Hypertext: Concepts, Systems, and Applications (ECHT'90)* (pp. 95–108). Cambridge: Cambridge University Press.
12. N. Streitz, J. Hannemann & M. Thüring (1989). From Ideas and Arguments to Hyperdocuments: Travelling through Activity Spaces. In *Proceedings of the 2nd ACM Conference on Hypertext (Hypertext'89)* (pp. 343 – 364). ACM Press.
13. N. Streitz, J. Haake, J. Hannemann, A. Lemke, W. Schuler, H. Schütt & M. Thüring (1992). SEPIA: A Cooperative Hypermedia Authoring Environment. In *Proceedings of the fourth International ACM Conference on Hypertext (ECHT '92)* (pp. 11 – 22) Milano, Italy, November 30 – December 4, 1992. ACM Press.
14. N. Streitz, J. Geißler, J. Haake, & J. Hol (1994). *DOLPHIN: Integrated Meeting Support across Liveboards, Local and Remote Desktop Environments.* Technical Report (Arbeitspapiere der GMD) No. 828, February 1994.