Learning in Dynamic Synchronous Information Spaces

Simon Schwantzer, Dirk Henrici, Paul Müller

Integrated Communication Systems
University of Kaiserslautern
67653 Kaiserslautern
{schwantzer, henrici, pmueller}@informatik.uni-kl.de

Abstract: Working in unmoderated small groups is a common approach to explore complex fields of knowledge. This form of collaboration is based on the tight conjunction of communication, cooperation, and coordination. Taking a look at the portfolio of environments for distributed learning, most of them are trying to realize synchronous collaboration just with communication tools. The aim of this work is the design of a dynamic information space for synchronous cooperation and coordination. Based on this model, an environment is created that fulfils the requirements to enable members of a small group working together efficiently via computer networks.

1 Introduction

A common approach to handle even sophisticated learning scenarios is collaborative work in small groups. In these scenarios, learning is based on different persons having a comparable level of knowledge but different views on the subject. The required information is available and provided in different forms such as books, scripts, and exercises. The information is transformed into knowledge by close collaboration of the group.

If one examines the portfolio of environments for distributed learning, one will find a large number of platforms for interaction in medium or big groups like classes or lectures. The interaction is usually realized asynchronously and therefore not applicable to the given scenario.

The aim of this work is the conceptualization and design of a distributed environment that allows a close, synchronous collaboration amongst a small group of learners. It supports a constructive approach to discover complex fields of knowledge in the area of higher education. The therefore related work is addressed in the next section. Based on this work, a requirement analysis is performed in Section 3. A model with focus on cooperation is then designed in Section 4. Furthermore, aspects of navigation, interaction, and cooperation are considered. A sample environment, so-called ’Synchronous Learning Environment’, is introduced and evaluated to demonstrate the applicability of the concept.

2 Related Work

This work follows two different approaches to determine the requirements for a synchronous learning environment: an application-oriented and a problem-oriented approach. In the application-oriented approach, relevant elements of existing tools are analyzed for their value in the given scenario. The problem-oriented approach tries to construe the essential elements of collaboration. This allows the development of new elements and the valuation of existing elements extracted in the application-oriented approach.

The analyzed tools for distributed and synchronous cooperative work divide into two classes: shared text editors and shared whiteboards.
Shared text editors came up at the beginning of the 1990s. One of the first was *GROVE*. The underlying work [EG91] introduced several important basics of groupware like concurrency issues and views. In [DB92] the term ’awareness’ was coined and the concurrency handling in GROVE classified as an active awareness strategy. In the same publication, another strategy - the use of passive awareness mechanisms - was introduced and implemented in an editor called *ShrEdit*. A newer representative of shared text editors is *NetEdit* [ZSEP01], which implements a number of more recently developed improvements concerning awareness mechanisms and application design.
Solutions which are not restricted to simple text creation and manipulation exist in the form of shared whiteboards. An early representative of these is *wb*, part of the MBone groupware architecture [Er94]. Aiming to share and annotate presentations, it offers simple drawing routines for all users. This concept was extended by *MediaBoard* [Tu98], which is also based on MBone. In MediaBoard the available tools were extended, the workspace was no longer bounded, and awareness mechanisms were implemented. A step away from working with fixed graphics was taken in *MMwb* [ZCG96]. It allowed to share and to synchronize audio and video data via a shared workspace. Based on the possibilities for interactive presentations, shared whiteboards where developed towards interactive conferencing systems and virtual classrooms.

Another approach for synchronous collaboration are realized with mind mapping tools like MindManager [MM08]. While the previously presented solutions were created for synchronous collaboration, mind mapping tools were designed for single-user information management and later on extended with multi-user functionality.

Different forms of communication in small group collaboration were examined and evaluated in [OOM95]. A comprehensive part of this work deals with the cooperation between users. It uses the structure of an information space as outlined in [Kr05]. Furthermore the work of [DS06] provides experience of artifact-mediated cooperation.

3 Requirement Analysis

Figure 1: Classification according to the 3C Collaboration Model

The well known 3C collaboration model allows us to categorize the different elements of collaboration (see [FPL06]). Each collaborative technology can be divided into three parts: communication, cooperation, and coordination (Figure 1). For the requirement analysis, we look at the parts separately and classify the elements accordingly.

In order to realize a distributed **communication**, we must be aware that communication happens through different communication channels. The available channels are voice, mimics, and gestures. In a distributed communication, the channels are often restricted or disrupted. The more communication channels are missing, the more prone to errors the communication will be.
Common types of computer supported distributed communication are textual, audio and video communication. Each type provides more communication channels than the previous one. In [OOM95] the different

types were examined for their value in small group collaboration. According to this study, high quality voice communication enables a distributed group to achieve the same results in collaboration as a local group. While a video communication is only of little additional value for the result, it is beneficial to the acceptance of a collaboration environment.

Coordination refers to the control of multiple users and the organization between them. Being more or less restrictive, the main issue of user control is concurrency handling. The advantage of a more restrictive system is providing clearer role definitions and requiring less organization between the users. In the given scenario, roles are surely existent, but the role allocation is very dynamic. A member's role often changes based on the individual level of knowledge in the topics addressed.
With a less restrictive concurrency handling, the cost for organization between the users increases. This leads to a greater operative overhead, which distracts the users from the cooperation. Aiming to reduce this operative overhead, [DB92] introduced passive awareness mechanisms. While active awareness mechanisms, like explicit role definitions and edit logs, are directly controlled by the users, passive mechanisms collect awareness information in the background and present them directly on the workspace. Focusing on passive mechanisms also has advantages with respect to cooperation: Apart from the possibility of dynamic role changes, the users can easily switch between close and loose cooperation.

In groupware systems, **cooperation** is defined by the interaction of multiple users with a 'shared material'. The shared material defines the usage and functional range of a groupware. To find an appropriate material, we have to adhere to the given problem: The goal is to reclaim knowledge by structuring information. A common definition of 'knowledge' is 'networked information'. 'Information' is interpretable data. In the given scenario, we can work directly with information because it is already available.
According to this definition, the shared material is a workspace which allows to place and link information. A simple example for such a workspace is a whiteboard: information can be placed in the form of text and simple graphics. The information units can be linked with lines, arrows, or other linking symbols. The combination of a workspace, information objects, and links can be named 'information space'.
There are two main requirements for an information space to be an appropriate material for the given scenario: On the one hand, the information space should provide a wide variety of information object types and good possibilities to structure information objects. On the other hand, the information space structure should be as simple as possible. Overly complex structures distract the user from the content. A concept for an

information space has to consider both requirements and therefore needs to find an adequate balance between them.

4 Model of a Structured Information Space

First of all, we have to define the basic structure of the information space. The simplest structure for an information space is a linear array of information objects. An example for a linear information space is pure text: the information objects are words, sentences, or sections. The linkage is given by their order. Linear information spaces are easy to access, but aside from the order of objects, there is no possibility of structuring information. The second simplest structure is a two-dimensional information space. One representative of this class is a whiteboard. As shown in the section above, a whiteboard can be used to structure information in the desired way. A two-dimensional information space therefore satisfies both given requirements. This structure can also be named 'information graph'.

The whiteboard lends itself to serve as a starting point for a model of the information graph. Owing to virtualization, many of the restrictions of classic whiteboards can be removed:

- Workspace: A virtual workspace needs not to be bounded. There are no restrictions to the level of detail a workspace can have.
- Objects: Information objects can be of any digitally representable form and are not restricted to text and graphic types. Objects can easily be manipulated, moved and removed.
- Links: Links are not restricted to symbols. They can be moved easily with the linked objects.

To allow the positioning of objects, the workspace needs a coordinate plane. Real-valued coordinates provide a high level of detail. The system should be origin-centered to handle an unbounded workspace.

It should be possible to place any type of data as an information object. This allows a universal application. Looking at whiteboards and hyper-medial information spaces (i.e. web pages), text and graphic objects play a decisive role in information spaces because they are easily accessible to the user. Further common object types in virtual environments are audio/video data, documents, and internal/external links. While external links represent URLs, internal links refer to other information graphs.

The missing final element required to complete the definition of the information space is the linkage. Links can only be created by specifying two or more information objects. Based on the number and arrangement of the linked object, a link can have different forms: One-to-one link, fork (1:n), join (n:1) or n:m mapping.

While the information space model is synchronized between all users, the aspects of visualization are not. This allows a much more flexible handling and dynamic changes between close and loose collaboration. With the exception of some basic object handling, the interaction with the workspace can also differ among the users.

The information space model itself does not contain any concrete visualization information, so the appearance of workspace, objects and links has to be defined separately. Two main aspects have to be considered when visualizing the workspace: First, while the local workspace is bounded, the global (information space) workspace is not. Second, users can have different views on the (global) workspace.

Both points require a differentiation between the global coordinate system of the information space and the local coordinate system of the local workspace. While the global coordinate system is part of the model and therefore the same for all users, the local one varies.[1] The visualization therefore is a projection from the information space to the local workspace.

So far only the placement of objects on the local workspace was considered, not their visual appearance. The objects are also being projected which allows assigning multiple representations for a single object type. Two representations are essential: the workspace representation and the detail representation. The detail representation displays all object information and makes it accessible. The workspace representation should save space but still provide the most important object information.

The visualization of links strongly depends on the form and complexity of the chosen link model. All link information should be visible directly on the workspace, so there is no need for different link representations. Multiple representations would complicate handling and therefore distract the user from the content.

As the information space is not bounded, mechanisms are needed to navigate in a possibly very large environment. According to the projection examined above, a navigation action is a change of the local viewport. Beside panning and zooming as common forms of navigation, the importance of a search feature increases with the size of the information space.

To enable multi-user interaction, a minimal set of object operations has to be realized by every application sharing an information space. These operations are: creation, manipulation, movement, and deletion. In addition

[1] This behavior is called *loose WYSIWIS* (What You See Is What I See).

to these, the linking of objects has to be provided as inter-object operation. Interaction between users takes place by interaction with information space objects. Therefore concurrency issues have to be considered here. If multiple users try to manipulate an object at the same time, manipulations are overwritten. A simple mechanism to avoid this behavior requires a small extension to the set of object operations: selection and deselection. A selected object can only be manipulated by the selector. This mechanism is easy to realize and sufficient to control the concurrency.

It is important to visualize a selection to make other users aware of it. This visualization is part of the so-called workspace awareness. 'Workspace Awareness' comprises all mechanisms that support the awareness among the users interacting in the (shared) information space. According to Section 3, we make use of passive awareness mechanisms only. In a loose collaboration, a user must be informed where the other users are acting. In close collaboration, a user must be informed about what the others are doing.

5 Synchronous Learning Environment

The so-called Synchronous Learning Environment (SLE) is a sample implementation of the concept introduced in Section 4. It was designed as sample environment and therefore the implemented information space model is a minimal model according to the findings.

The SLE is divided into four separable modules: Data Management, User Management, Communication, and Workspace. Modularization has several advantages: It allows to enhance or to replace single modules without affecting the others as long as the interface is maintained. Further it provides the possibility of exchanging single modules to adapt the environment to different requirements.

The data and user management modules are both Java servlets. The user management module provides user identification and user session handling. The data management module is informed about information space changes and maintains a persistent representation in a database. Both modules communicate with the server component of the workspace module. With a Red5 Open Flash Server and a Flex application running in the Adobe Flash VM, the workspace module is based on a classic client-server architecture. The communication module is built on the same architecture and offers real-time audio communication between all users.

The information space has the form of an information graph (two-dimensional), is unbounded and provides the functionality of placing information objects in an origin-centered, real-valued coordinate system. Implemented information object types are: Label, Image, Binary, URL, and Graphlink. Label is a simple single-line text object. A Binary object corresponds to the 'universal' object and handles arbitrary binary files. Object operations are limited to the minimal set: creation, manipulation, movement, deletion, selection, and deselection. External and internal links are realized with the URL and Graphlink object types. Links are realized as simple non-directed 1:1 correspondences.

A client projects the information graph to a map-based local workspace. Navigation elements are panning, zooming, and map-based navigation. Furthermore, SLE provides a search on all meta-information including substrings.

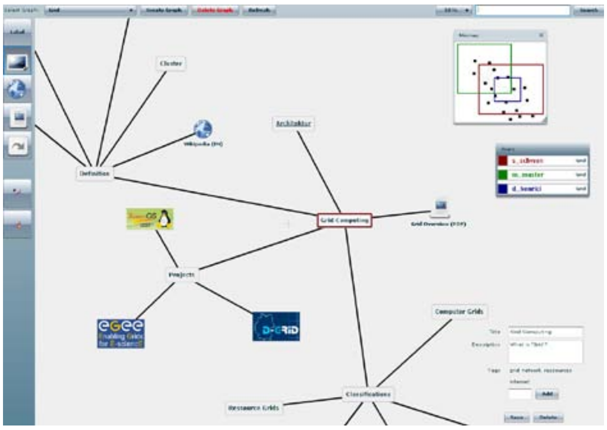

Figure 2: Synchronous Learning Environment

6 Evaluation

The flexible model created in Section 4 allows a range of concrete implementations. The evaluation aims to analyze the concrete model selection done in the sample environment SLE. It shows advantages and disadvantages in information structuring and collaboration based on the given implementation.
The SLE was tested in groups of two and three users. The test started with a blank information space. The topic was unknown to the participants to

create an equal level of knowledge. Information material was provided in digital form.

The basic information space structure was accepted by all users. The unbounded global workspace and the non-restricted level of detail where rated positively. It accommodated the demands of a dynamically growing information space much better than a bounded workspace does. Information objects of type Label were significantly more often placed than any other types. Beside this fact, the usage of other object types strongly depends on the form of the provided information material. The users felt that an object type which provides the possibility of holding complete text paragraphs was missing. The one-lined Label object type did not suit all applications. The available 1:1 correspondences given by the implemented link model fulfilled the users' needs. As an addition to the model, a means of adding information to links in form of a weighting was requested.

The given set of object interaction was used in its complete functional range. In many situations the possibility of changing the object type was required. This functionality is therefore an important feature for further implementations. The implemented concurrency handling worked well in learning groups of the given size. It has to be observed how collaboration efforts scale with the size of the learning group. The implemented awareness algorithms were accepted and used. An extension showing the current actions of the other users (like 'editing', 'navigating', and 'idle') was requested. The provided voice communication was rated as a very important element of collaboration. Video communication was not missed but also not excluded as a possible form of communication. Several aspects of coordination were transferred into the communication.

7 Conclusions and Future Work

The aim of this work was to virtualize a scenario in which knowledge is built up via information structuring. Such a virtualization provides the possibility of distributing the collaboration.
Based on the requirements of cooperation, communication and coordination (Section 3), a model for an appropriate shared material in form of an information space was created in Section 4. The model provides a high flexibility for concrete implementations without breaking the restrictions set by the requirements. Considerations about visualization and interactions were taken. The user coordination based on passive awareness mechanisms completed the collaboration concept. With the SLE, an implementation for

the created concept is available to demonstrate and evaluate the basic functionality. It can also clarify some technical issues which had not been considered before. In the last section, this implementation was evaluated under aspects of the concrete information space model to fine-tune the model requirements and collaboration mechanisms.
The given evaluation is limited to usage experiences within the sample environment SLE and therefore does not replace a comprehensive study about advantages and disadvantages of unmoderated learning in a two-dimensional information space as developed in Section 4. It also has to be investigated how cooperation and coordination scale with the size of the learning group. So next, such a study needs to be performed, ideally based on a full-featured and refined implementation of the concept.

References

[DB92] Dourish, P. and Bellotti, V. (1992). Awareness and coordination in shared workspaces. In CSCW '92: Proceedings of the 1992 ACM conference on Computersupported cooperative work, pages 107–114, New York, NY, USA. ACM.

[DS06] Dwyer, N. and Suthers, D. D. (2006). Consistent practices in artifact-mediated collaboration. International Journal of Computer-Supported Collaborative Learning, 1(4):481–511.

[EG91] Ellis, C. A., Gibbs, S. J., and Rein, G. (1991). Groupware: some issues and experiences. Commun. ACM, 34(1):39–58.

[Er94] Eriksson, H. (1994). Mbone: the multicast backbone. Commun. ACM, 37(8):54–60.

[FPL06] Fuks, H., Pimentel, M., and de Lucena, C. J. P. (2006). Ru-typing-2-me? evolving a chat tool to increase understanding in learning activities. International Journal of Computer-Supported Collaborative Learning, 1(1):117–142.

[Kr05] Kritzenberger, H. (2005). Multimediale & Interaktive Lernräume. Oldenbourg Wissenschaftsverlag.

[MM08] MindManager 8, MindJet GmbH, http://www.mindjet.com/

[OOM95] Olson, J. S., Olson, G. M., and Meader, D. K. (1995). What mix of video and audio is useful for small groups doing remote real-time design work? In CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems, pages 362–368, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

[Tu98] Tung, T.-L. (1998). Mediaboard: A shared whiteboard application for the mbone. Technical report, UCB CS Masters Thesis.

[ZSEP01] Zafer, A. A., Shaffer, C. A., Ehrich, R. W., and Perez, M. (2001). Netedit: a collaborative editor. Technical report.

[ZCG96] Zheng, S., Caron, S., and Georganas, N. D. (1996). Mmwb: A multimedia whiteboard for distance learning applications.