

---

# THE ROLE OF PERSONAL AND SHARED DISPLAYS IN SCRIPTED COLLABORATIVE LEARNING

---

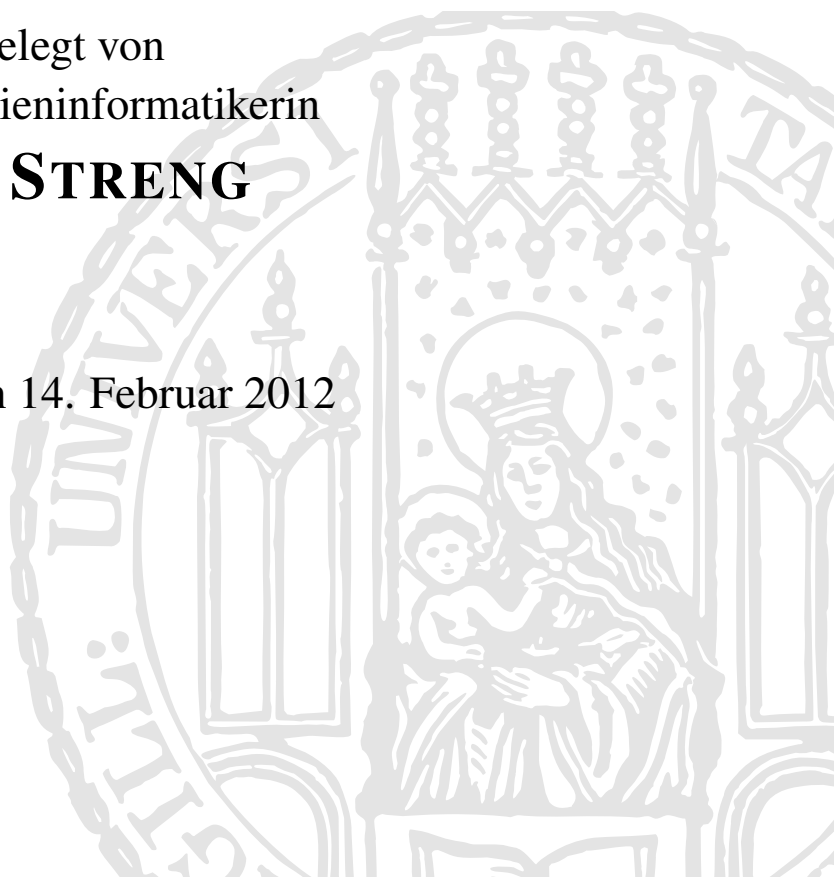
## DISSERTATION

an der Fakultät für Mathematik, Informatik und Statistik  
der Ludwig-Maximilians-Universität München

vorgelegt von  
Diplom-Medieninformatikerin

**SARA STRENG**

München, den 14. Februar 2012



Erstgutachter: Prof. Dr. Heinrich Hußmann  
Zweitgutachter: Prof. Dr. Pierre Dillenbourg

Tag der mündlichen Prüfung: 24ter April 2012



## ABSTRACT

Over the last decades collaborative learning has gained immensely in importance and popularity due to its high potential. Unfortunately, learners rarely engage in effective learning activities unless they are provided with instructional support. In order to maximize learning outcomes it is therefore advisable to structure collaborative learning sessions. One way of doing this is using collaboration scripts, which define a sequence of activities to be carried out by the learners. The field of computer-supported collaborative learning (CSCL) produced a variety of collaboration scripts that proved to have positive effects on learning outcomes. These scripts provide detailed descriptions of successful learning scenarios and are therefore used as foundation for this thesis.

In many cases computers are used to support collaborative learning. Traditional personal computers are often chosen for this purpose. However, during the last decades new technologies have emerged, which seem to be better suited for co-located collaboration than personal computers. Large interactive displays, for example, allow a number of people to work simultaneously on the same surface while being highly aware of the co-learners' actions. There are also multi-display environments that provide several workspaces, some of which may be shared, others may be personal. However, there is a lack of knowledge regarding the influence of different display types on group processes. For instance, it remains unclear in which cases shareable user interfaces should replace traditional single-user devices and when both personal and shared workspaces should be provided. This dissertation therefore explores the role of personal and shared workspaces in various situations in the area of collaborative learning. The research questions include the choice of technological devices, the seating arrangement as well as how user interfaces can be designed to guide learners.

To investigate these questions a two-fold approach was chosen. First, a framework was developed, which supports the implementation of scripted collaborative learning applications. Second, different prototypes were implemented to explore the research questions. Each prototype is based on at least one collaboration script. The result is a set of studies, which contribute to answering the above-mentioned research questions. With regard to the choice of display environment the studies showed several reasons for integrating personal devices such as laptops. Pure tabletop applications with around-the-table seating arrangements whose benefits for collaboration are widely discussed in the relevant literature revealed severe drawbacks for text-based learning activities. The combination of laptops and an interactive wall display, on the other hand, turned out to be a suitable display environment for collaborative learning in several cases. In addition, the thesis presents several ways of designing the user interface in a way that guides learners through collaboration scripts.

## ZUSAMMENFASSUNG

In den letzten Jahrzehnten hat kollaboratives Lernen auf Grund seines großen Potentials in hohem Maße an Bedeutung und Popularität gewonnen. Leider führt kollaboratives Lernen nicht von Haus aus zu optimalen Ergebnissen, da Lernende selten von sich aus effektive Lernaktivitäten durchführen. Um Lerneffekte zu erhöhen ist es daher ratsam kollaborative Lernsituationen zu strukturieren. Eine Möglichkeit der Strukturierung sind sogenannte “Kollaborationsskripte”, die eine Sequenz von Lernaktivitäten definieren, um ein bestimmtes Lernziel zu erreichen. Im Forschungsbereich “Computergestütztes kollaboratives Lernen” sind eine Reihe von Kollaborationsskripten entstanden, die Lernen nachweislich fördern. Diese Skripte stellen detaillierte Beschreibungen von Lernszenarien dar, die kollaboratives Lernen effektiver gestalten. Aus diesem Grund wurden Kollaborationsskripte als pädagogische Grundlage für diese Arbeit gewählt.

In vielen dieser Fälle werden Computer eingesetzt, um kollaboratives Lernen zu unterstützen. Traditionell fällt die Wahl häufig auf herkömmliche persönliche Computer. In den letzten Jahren entstanden jedoch neue Technologien, die für Kollaboration besser geeignet scheinen. Große interaktive Displays ermöglichen beispielsweise mehreren Personen gleichzeitig auf derselben Oberfläche durch Berührung zu interagieren wodurch das Bewusstsein für die Aktivitäten der anderen Gruppenmitglieder gefördert wird. Es gibt auch sogenannte “Multi-Display Umgebungen”, die mehrere Arbeitsbereiche bieten können, sowohl persönliche als auch Arbeitsbereiche für die gesamte Gruppe.

Über die Auswirkung dieser verschiedenen Displayarten auf kollaborative Prozesse gibt es jedoch viele offene Fragen. Es ist beispielsweise unklar, in welchen Fällen persönliche Computer durch Gruppendisplays ersetzt werden bzw. wann beides zur Verfügung gestellt werden sollte. In dieser Dissertation wird deshalb die Rolle persönlicher und gemeinsamer Arbeitsbereiche in verschiedenen kollaborativen Lernsituationen untersucht. Die Fragestellung umfasst die Wahl der Displays, die Sitzanordnung der Lernenden und das Design der Benutzerschnittstelle als Mittel um Lernende zu führen.

Um diese Fragen zu klären wurde ein zweiteiliger Ansatz gewählt. Zum einen wurde ein Framework entwickelt, das die Implementierung von Skript-basierten kollaborativen Anwendungen unterstützt. Im zweiten Schritt wurden verschiedene Prototypen entwickelt, mit deren Hilfe die Forschungsfragen untersucht wurden. Jeder Prototyp basiert auf einem oder mehreren Kollaborationsskripten. Das Ergebnis sind zahlreiche Studien, die zur Beantwortung der oben genannten Fragestellung beitragen. Bezüglich der Wahl geeigneter Displays zeigen die Studien verschiedene Gründe auf, die *für* die Einbeziehung persönlicher Geräte wie Laptops sprechen. Anwendungen für interaktive Tische, um die Lernende herumsitzen können und deren positive Auswirkung auf Gruppenprozesse in der Literatur stark betont werden, zeigten bei textlastigen Lernaufgaben jedoch erhebliche Nachteile auf. Die Kombination von Laptops und einem interaktiven Wanddisplay erwies sich dagegen in mehreren Fällen als geeignete Displayumgebung für kollaboratives Lernen. Neben der Verwendung unterschiedlicher Displaytypen zeigt die vorliegende Arbeit darüberhinaus verschiedene Möglichkeiten die grafische Benutzerschnittstelle so zu gestalten, dass Lernende gemäß Kollaborationsskripten geführt werden.

## ACKNOWLEDGMENTS

This work would not have been possible without the support and encouragement of many people. First and foremost, I owe my deepest gratitude to my supervisors. Heinrich Hussmann gave me the opportunity to undertake this interdisciplinary endeavor and I am thankful for his counsel throughout the years. Special thanks also goes to Pierre Dillenbourg for joining the committee and for investing so much of his valuable time to provide highly appreciated feedback.

I also want to thank Karsten Stegmann and Frank Fischer for their advice on educational aspects. Frank Fischer introduced me to collaboration scripts and supported our interdisciplinary cooperation on various levels. Karsten co-advised almost all of the projects and for this reason spent several hours in our lab every two weeks. Coming from an HCI background this cooperation was not only vital but also very enjoyable.

The groundwork of this thesis consists of several small projects. In each project at least students participated as part of their project, diploma or bachelor thesis. Without their valuable contribution it would not have been possible to get this work done in a time frame of about four years. The students who contributed to this thesis are (in chronological order): Younes Alj (2009), Sonja Böhm (2009), Max Halbinger (2009), Jonas Koch (2009), Simon Lutzenberger (2009), Sonja Böhm (2010), Wolfgang Matzke (2010), Mona Metzner (2010), Marc Mühlbauer (2011), Günther Obexer (2009), Christine Wagner (2010), Wenqi Zhang (2010), Tim Kugler (2011), Maraike Stuffer (2011), Sarah Tausch (2010), Sarah Torma (2011), Christine Wagner (2011), Fabian Winter (2011), Susanne Zirker (2011). The theses of all involved students will be cited again later in the according chapters.

I would also like to mention my colleagues who created an amazing working environment, which I enjoyed every office day over the past years. In particular I want to thank the PhDs who were senior to me and at times mentored me as well: Alexander De Luca (who I could always discuss study designs and evaluations with and who was a very entertaining office mate throughout my PhD), Sebastian Boring (who played devil's advocate in many fruitful discussions) and Richard Atterer (who not only picked the name 'Wanna Be PhD' for our PhD meeting but also helped bringing it into being).

A very warm thank you also goes to Helga and Allan Crockatt, Hans Streng and Rainer Liebhart for taking an interest in my research and for proofreading parts of the thesis.

Finally, I want to thank Albrecht Schmidt for introducing me to the field of human-computer interaction. Although I never directly worked with Albrecht, he made it possible for me to work with amazing people like Anind Dey and Hans-Werner Gellersen. It was also Albrecht who convinced me that there really is no other option for me than doing a PhD.



# TABLE OF CONTENTS

<b>I</b>	<b>INTRODUCTION AND RELATED WORK</b>	<b>1</b>
<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Problem Statement . . . . .	5
1.2	Contributions . . . . .	5
1.3	Structure . . . . .	7
<b>2</b>	<b>Related Work</b>	<b>11</b>
2.1	Collaboration Scripts . . . . .	11
2.1.1	Definition . . . . .	11
2.1.2	Example Scripts . . . . .	12
2.1.3	Taxonomies . . . . .	14
2.1.4	IMS-LD . . . . .	16
2.1.5	Collaboration Scripts as Foundation for our Methodology . . . . .	16
2.2	Collaborative Learning Applications . . . . .	17
2.2.1	Tabletops in Education . . . . .	18
2.2.2	Tangibles . . . . .	20
2.2.3	Mobile and Handheld Applications . . . . .	21
2.2.4	Classroom Technologies . . . . .	22
2.3	The Influence of the Display (Environment) . . . . .	24
2.3.1	Display Types and Display Environments . . . . .	24
2.3.2	Effects of Displays on Collaborative Processes . . . . .	28
<b>II</b>	<b>DISTRIBUTED CLASSROOM APPLICATIONS</b>	<b>33</b>
<b>3</b>	<b>From Graphical Learning Designs to Scripted Learning Applications</b>	<b>35</b>
3.1	Related Work . . . . .	36
3.2	An Example Application for Scripted Text Comprehension Training . . . . .	37
3.2.1	M.U.R.D.E.R. Script . . . . .	38
3.2.2	M.U.R.D.E.R. Application . . . . .	38
3.2.3	Usability Assessment . . . . .	40

3.3	eXtremely Simple Scripting (XSS) Framework . . . . .	41
3.3.1	How Macro Scripts Define the Application Structure . . . . .	43
3.3.2	Automatic Generation of the Application Structure . . . . .	44
3.3.3	An Interdisciplinary Development Process . . . . .	46
3.3.4	Tools and Application Controls . . . . .	48
3.4	Case Study . . . . .	50
3.4.1	Method . . . . .	52
3.4.2	Results . . . . .	53
3.4.3	Summary . . . . .	53
3.5	Towards an Integration of Shareable User Interfaces . . . . .	54
3.5.1	Scope of the XSS Framework . . . . .	54
3.5.2	Vision of an Extension for Shareable User Interfaces . . . . .	55
3.6	Summary and Discussion . . . . .	58

### III EFFECTS OF DISPLAY CONFIGURATIONS

61

#### 4 Choosing a Display (Environment)

63

4.1	Iterative Design of the ArgueTable . . . . .	64
4.1.1	Argumentative Knowledge Construction . . . . .	64
4.1.2	Iterative Design Process . . . . .	66
4.1.3	Paper Prototype Study . . . . .	67
4.1.4	Case Studies with Functional Prototypes . . . . .	71
4.2	ArgueTable vs. ArgueWall . . . . .	72
4.2.1	Method . . . . .	73
4.2.2	Results . . . . .	75
4.2.3	Summary and Discussion . . . . .	76
4.3	Follow-Up Studies . . . . .	77
4.3.1	Paper Prototype Follow-Up Study . . . . .	79
4.3.2	Mixed-Fidelity Prototype Study (Side-by-Side) . . . . .	80
4.3.3	Mixed-Fidelity Prototype Study (Face-to-Face) . . . . .	82
4.3.4	Mixed-Fidelity Prototype Study (Side-by-Side vs. Face-to-Face) . . . . .	83
4.3.5	Summary and Discussion . . . . .	84
4.4	Examining the Importance of Display Environment Factors . . . . .	87
4.4.1	Structured Academic Controversies . . . . .	88
4.4.2	Display Environments . . . . .	88
4.4.3	Method . . . . .	89
4.4.4	Results . . . . .	90



4.4.5	Summary and Discussion . . . . .	92
4.5	Lessons Learned . . . . .	93
<b>5</b>	<b>Implicit Guidance in Multi-Display Environments</b>	<b>95</b>
5.1	Note&Share: Facilitating Knowledge Sharing . . . . .	96
5.1.1	Concept . . . . .	96
5.1.2	Implementation . . . . .	97
5.2	Hidden Profile Experiment . . . . .	98
5.2.1	Method . . . . .	98
5.2.2	Results . . . . .	100
5.2.3	Summary and Consequences . . . . .	102
5.3	Experiment with an Open Problem Solving Task . . . . .	102
5.3.1	Method . . . . .	103
5.3.2	Results . . . . .	105
5.3.3	Summary and Discussion . . . . .	109
5.4	Related Work . . . . .	110
5.5	Lessons Learned . . . . .	111
<b>IV</b>	<b>EXPLICIT GUIDANCE MECHANISMS</b>	<b>113</b>
<b>6</b>	<b>Guiding Widgets and GUI Concepts for Shareable User Interfaces</b>	<b>115</b>
6.1	Task Assignment . . . . .	116
6.1.1	Task Tokens . . . . .	117
6.1.2	Method . . . . .	118
6.1.3	Results . . . . .	120
6.1.4	Summary and Discussion . . . . .	122
6.2	Role Assignment . . . . .	122
6.2.1	First Concepts: ‘Spin the Bottle’ and ‘Draw a Card’ . . . . .	122
6.2.2	Paper Prototype Study . . . . .	124
6.2.3	Conceptual Redesign: Ludo . . . . .	125
6.2.4	Method . . . . .	128
6.2.5	Results . . . . .	131
6.2.6	Summary and Discussion . . . . .	135
6.3	Towards Automatic Code Generation . . . . .	136
6.4	Related Work . . . . .	137
6.5	Lessons Learned . . . . .	138

<b>7</b>	<b>Using Ambient Feedback for Mirroring and Guidance</b>	<b>139</b>
7.1	Mirroring and Guidance Systems . . . . .	140
7.2	An Ambient Group Mirror for Qualitative Feedback . . . . .	142
7.2.1	Concept . . . . .	143
7.2.2	Design and Implementation . . . . .	144
7.2.3	Method . . . . .	146
7.2.4	Results . . . . .	147
7.2.5	Summary and Discussion . . . . .	149
7.3	Group Mirrors in the Classroom . . . . .	149
7.3.1	Project Seminar . . . . .	150
7.3.2	Concept, Design and Implementation . . . . .	150
7.3.3	Method . . . . .	154
7.3.4	Results . . . . .	156
7.3.5	Summary and Discussion . . . . .	158
7.4	Lessons Learned . . . . .	159
<b>V</b>	<b>CONCLUSION AND FUTURE WORK</b>	<b>161</b>
<b>8</b>	<b>Conclusion and Future Work</b>	<b>163</b>
8.1	Summary of Findings . . . . .	163
8.2	Discussion and Future Studies . . . . .	166
8.3	Future Work on the XSS Framework . . . . .	168
8.4	Final Remarks . . . . .	170
<b>Appendices</b>		<b>171</b>
A	Example of an IMS-LD Document . . . . .	173
B	XSS Framework Implementation . . . . .	175
C	Parsing IMS-LD Documents with the NDigester . . . . .	177
D	Finalization of the Client . . . . .	178
E	Transcription of Note&Share Discussions . . . . .	179
<b>Bibliography</b>		<b>181</b>

# I

## INTRODUCTION AND RELATED WORK



# Chapter 1

---

## Introduction

Collaborative learning has rightly attracted a lot of attention in recent years. Research has shown that collaborative learning can have positive effects on motivation, achievement and social relationships (cf. Slavin, 1995 and Qin et al., 1995). However, it has frequently been found that learners rarely engage in effective collaborative learning activities such as explaining or asking each other questions spontaneously (Kobbe et al., 2007). Consequently, instructional support is needed to improve learning outcomes (Kollar et al., 2006). One solution is the use of collaboration scripts, which structure collaborative learning processes by defining a sequence of learning activities. For instance, learners could be asked to first read a text, then summarize its content, ask each other questions and so forth. In addition, roles are assigned to different learners such as the role of a teacher who questions other learners. This approach induces social and cognitive learning activities which rarely occur in unstructured collaborative learning.

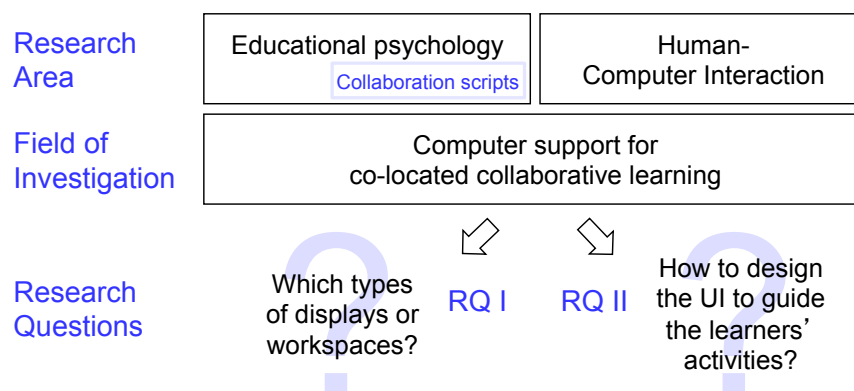
Collaboration scripts have become fairly popular, particularly in computer-supported collaborative learning (Kobbe et al., 2007). The use of technology in education has several advantages. First and foremost there is the digital advantage: Computers enable distance communication, efficient search and history functions as well as fast processing and visualization of large information chunks. Furthermore, documents can easily be copied or edited by several people simultaneously. This opens the door for new forms of collaborative learning. For instance, distance communication allows learners from different continents to learn together in multicultural manners (Berger et al., 2001). But technology creates new possibilities in face-to-face settings, too. The ArgueGraph (Jermann and Dillenbourg, 2003), for example, is a collaboration script that relies on computers. Students state their opinions by filling out questionnaires on a computer. The computer then evaluates the answers and builds pairs of learners with as opposing opinions as possible. Afterwards, the pairs are asked to fill out the same questionnaire again. As the pair needs to agree on a question which they answered differently the first time, the script furthers argumentation. Without technology this script would cause a disproportionate amount of work for the teacher.

Of course, the use of technology can have negative implications, too. One major issue is usability. Pen and paper is usually easier to use than computer applications. Often, difficulties in handling the application divert the attention from the educational task. If the usability of the application is poor, it becomes a barrier to learning rather than a support.

Assuming that the trend towards an increased usage of computer applications in the field of collaborative learning will continue, this dissertation examines how the *user interface* of scripted collaborative learning applications should be designed in order to avoid negative effects such those described above, and moreover, to guide learners according to collaboration scripts. Therefore, such computer applications are investigated from a human-computer interaction (HCI) perspective. In addition, the focus is set on *co-located* scenarios where learners are in the same place at the same time. In this context interesting research questions like the choice of display and particularly the role of private and shared workspaces arise.

Figure 1.1 illustrates the cornerstones of the thesis. The thesis is based on previous research in the areas of educational psychology and human-computer interaction. The field of investigation is computer support for scripted collaborative learning in co-located settings. The two overall research questions are:

- RQ I: Which types of displays or workspaces are suitable in which scripted collaborative learning scenarios?
- RQ II: How should user interfaces be designed to guide learners through collaboration scripts?



**Figure 1.1:** Research area and overall research questions addressed in this thesis.

The importance of the investigation of these questions goes beyond the scope of collaboration scripts. Nowadays, interactive whiteboards are bought by thousands of schools across Europe, with little empirical evidence of their effectiveness or basic theoretical recommendation of how teachers should use them. The same is happening with interactive tabletop displays (interactive table surfaces such as the Microsoft Surface<sup>1</sup>). This leads to a mismatch between the dynamism

<sup>1</sup> [www.microsoft.com/surface/](http://www.microsoft.com/surface/)

of the technology market and the available research results which could promote their use as well as inspire teachers and decision makers. The goal of this thesis is to improve this situation by systematically exploring the design space of technology-supported collaborative learning activities and empirically measuring the educational effects of these design choices.

## 1.1 Problem Statement

More and more often computers are used to support collaborative learning. This also applies to the field of collaboration scripts. However, there is a lack of research on the role of the user interface in collaborative learning. This even concerns basic questions such as the choice of display (environment), which is fundamental for any application that supports co-located collaborative learning. Several studies in the field of computer-supported collaborative work (CSCW) have investigated effects of different display characteristics such as their size, the number of displays (Inkpen et al., 2005) and their orientation (vertical wall vs. horizontal tabletop displays; Rogers and Lindley, 2004). However, there is little research on the effects of display environments on collaborative *learning*. In addition, there is no wider approach which provides an overview of various display characteristics (such as number, size, orientation, input capabilities, ...) or even allows the comparison of their relative significance.

Furthermore, when computers are used for collaborative learning, another fundamental question is how to convey the instructions to the learners. In the field of scripted collaborative learning this is usually done by tutors and teachers (in particular in co-located collaborative learning) or in the form of textual instructions (in particular in web-based distance learning applications such as S-COL, Wecker et al., 2010). As a consequence many collaborative learning applications do not take advantage of the design of the user interface as a means to guide learners. And even if they do, the means of guidance is often not the focus of the conducted research. The primary interest is usually on the *effect* of guidance on learning outcomes. As a consequence guidance is simply applied without investigating different aspects such as the appropriate strictness or the amount of guidance that can be conveyed by adept user interface design. In addition, guidance mechanisms are often not discussed in the publications or not addressed as such. Accordingly, there is not only no holistic approach to answering the question of how to design the user interface to guide learners - it is also hard to get an overview of the various possibilities from the relevant literature.

## 1.2 Contributions

This thesis contributes to the field of (scripted) computer-supported collaborative learning in three ways. First, it provides insights into the role of the display environment in scripted collaborative learning applications. Second, we explored different ways in which guidance can be facilitated through the user interface. Third, we developed a framework which supports the implementation of scripted collaborative learning sessions as well as their administration during runtime.

### *The Role of the Display Environment*

As it is almost impossible to choose a display environment based on the results reported in the existing body of literature, the thesis explores the significance of different display characteristics in scripted collaborative learning, with particular emphasis on the role of private and shared workspaces. For instance, a fundamental question is whether one shared display is sufficient for both individual and collaborative phases, or whether a combination of private and shared displays is beneficial. To examine such questions, a variety of different display characteristics needs to be considered. Examples of such characteristics are text input capabilities, whether or not the display facilitates eye contact and natural communication, and the awareness of the partner's activities during individual phases. Although previous studies have already verified various advantages and disadvantages of different types of displays, it is hardly possible to judge their significance. For example, is it more important to provide learners with private workspaces that allow them to work undisturbed and without fear of criticism (which would speak in favor of private workspaces during individual phases), or is it more important to be aware of co-learners' activities (which would speak in favor of shared workspaces)? To be able to compare the significance of multiple display characteristics, an exploratory approach is chosen.

### *How the User Interface Can be Designed to Guide Learners*

In addition, the thesis demonstrates different ways of designing the user interface in order to guide learners through collaboration scripts. The different prototypes presented throughout the thesis provide several examples. In addition, three guidance mechanisms are investigated in more detail. First, the approach of using different displays to clarify the tasks in subsequent learning phases is explored. Second, graphical user interface (GUI) elements ('*Guiding Widgets*') are presented, which assign tasks and roles to different group members sitting around a tabletop display. In this regard one main goal is to determine how strict the guidance by a computer application should be in order to structure the learning process without being too restrictive. Another question is whether multiple guidance aspects (for instance roles *and* tasks) can be assigned without overwhelming the learners or without making the user interface too complex. The third guidance mechanism that is explored in more detail is '*group mirrors*'. Group mirrors are separate systems, which are dedicated solely to the purpose of guidance. While Guiding Widgets are part of a learning application with which the learners interact, group mirrors merely monitor the learners and provide feedback. Again, an exploratory approach is chosen to give an overview of various guidance mechanisms and their effects.

### *A Framework for Scripted Collaborative Learning*

The last contribution is a framework, which supports the rapid development process of scripted collaborative learning applications. This *eXtremely Simple Scripting (XSS) Framework* automatically generates a greater part of the learning application from a given collaboration script. This automatic generation covers the complete server application as well as the basic client functionality. The client user interface needs to be finalized manually by programmers or user interface designers. To support interdisciplinary developer teams in planning how the final user interface



should look, the XSS Framework provides a simulation tool which allows developer teams to navigate multiple preliminary clients through the different learning phases. The main objective of building the framework was to reduce the amount of repetitive tasks during the development of the different scripted learning applications that were used to explore the research questions.

## 1.3 Structure

The thesis is structured in five parts. Part I introduces the topic and provides an overview of the relevant literature. Part II presents the framework which supports the rapid development of scripted collaborative learning applications. Part III and IV address the two overall research questions respectively. In part V the findings are summarized and discussed.

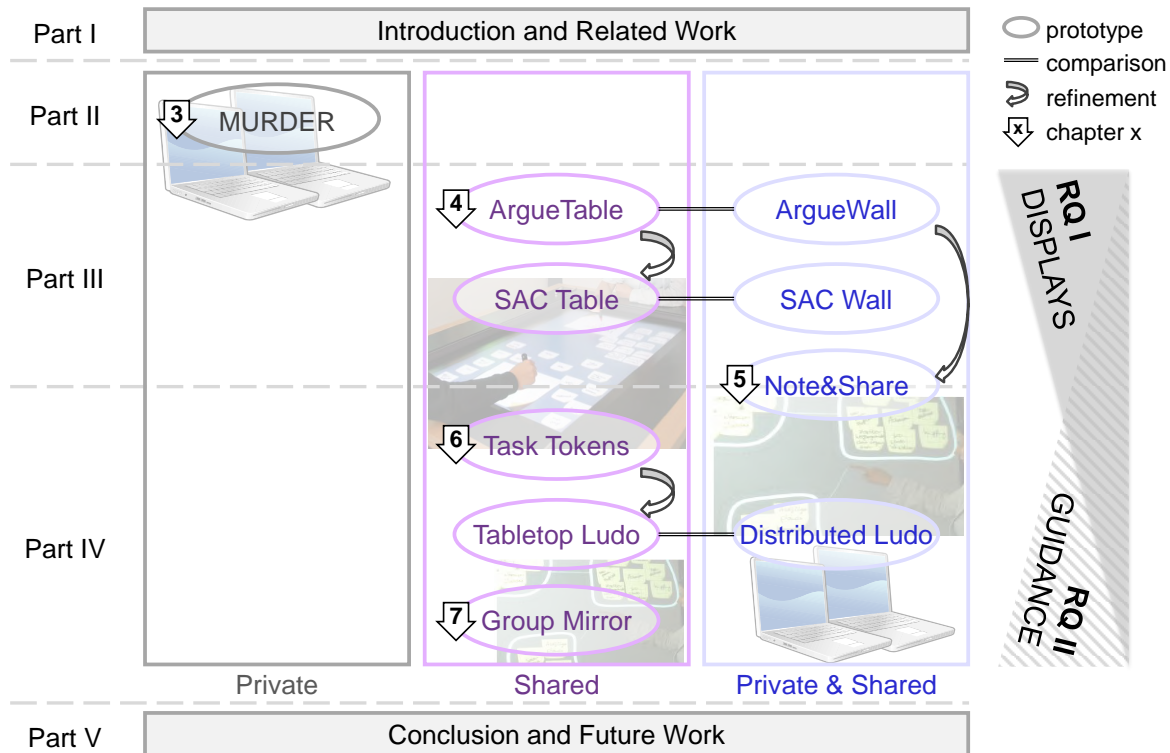
Each part is divided further into chapters. A closer look at these chapters reveals that the structure of the thesis is quite unconventional due to its exploratory nature. Figure 1.2 shows a roadmap of the chapters. Each oval corresponds to a prototype that was implemented to explore one or both of the overall research questions. The horizontal axis represents the type of display environment: private workspaces (e.g. laptops), shared workspaces (e.g. tabletop displays) or combinations of private and shared workspaces (e.g. laptops in combination with wall displays). The vertical axis illustrates the explored research question (RQ). Part II presents the XSS Framework, which is intended to facilitate the development of prototypes. It provides groundwork but does not explore research questions yet. Part III investigates the role of the display environment (RQ I). However, guidance mechanisms are used and evaluated in these projects as well. Part IV mainly explores guidance mechanisms (RQ II) but the display environment is inevitably a matter of investigation as well. These oppositional trends are illustrated by the research question triangles on the right of Figure 1.2.

The following paragraphs summarize the content of the remaining chapters:

*Chapter 2 – Related Work* provides an overview of collaboration scripts, which are the educational foundation for this thesis. In addition, a survey of collaborative learning applications is presented with a focus on the user interface. All of these applications are collaborative but not necessarily scripted. Finally, previous research on the effects of different displays and display environments on collaborative processes is discussed. Many of these studies were conducted in the field of computer-supported collaborative work (CSCW) rather than learning (CSCL).

### **PART II**

*Chapter 3 – From Graphical Learning Designs to Scripted Learning Applications* presents the XSS Framework which supports the rapid development of scripted collaborative learning applications. At first, an example application called M.U.R.D.E.R. is presented. This application is used to illustrate the motivation and functionality of the framework. Afterwards, a case study is presented, which was conducted to evaluate the



**Figure 1.2:** Thesis roadmap.

usability of the framework and verify that it supports the development of any collaboration script. At the end of the chapter, a concept for a framework extension towards an integration of shareable user interfaces is described.

### PART III

*Chapter 4 – Choosing a Display (Environment)* explores the first research question by comparing effects of different display environments. This is done in two different learning scenarios. For each scenario different prototypes were built. The first prototypes (ArgueTable and ArgueWall) are based on collaboration scripts, which support argumentative knowledge construction. To further investigate the findings and to verify whether they are applicable to other learning scenarios two more prototypes were built, which are based on another script called Structured Academic Controversies or short ‘SAC’ (SACTable and SACWall).

*Chapter 5 – Implicit Guidance in Multi-Display Environments* explores the use of different displays to implicitly guide learners through subsequent phases. The idea emerged during the design of the ArgueWall: To emphasize which task is expected of the learners in each phase, they are provided with different displays, each of which is dedicated to the specific task. This form of implicit guidance is used by a prototype called

Note&Share, which furthers knowledge sharing process. This chapter is related to both research questions.

#### **PART IV**

*Chapter 6 – Guiding Widgets and GUI Concepts for Shareable User Interfaces* presents user interface concepts that aim at guiding learners through collaboration scripts. Therefore, it investigates the second research question. The first prototype is Task Tokens, which guides learners through a sequence of tasks. Due to findings from a user study, an extended concept ‘Ludo’ was developed, which distributes roles in addition to tasks. Both concepts were developed for tabletop displays with around-the-table seating arrangements. To examine the role of the display environment in this scenario, Tabletop Ludo was compared to Distributed Ludo, which consists of laptops and a shared wall display.

*Chapter 7 – Script-Based Mirroring and Guidance Systems* presents another form of guidance, which is not conveyed through user interface components or concepts but through dedicated feedback systems. First, the role of the visualization of such mirroring and guidance systems for scripted collaborative learning scenarios is investigated in a laboratory study. To verify the findings outside the laboratory, a second study was conducted in a project seminar of a German high school.

#### **PART V**

*Chapter 8 – Conclusion and Future Work* summarizes and discusses the main findings of the previous chapters and gives an outlook on future work in the field of co-located scripted collaborative learning.



# Chapter 2

---

## Related Work

This chapter presents fundamental related work that will play a role throughout the thesis. At first, we introduce collaboration scripts, which provide the foundation of this work from an educational perspective. Second, an overview of collaborative learning applications is provided. Finally, studies on effects of different display environments are presented. Further literature, which is only relevant for specific projects, is addressed later in the according chapters.

### 2.1 Collaboration Scripts

A common problem with collaborative learning is that learners rarely engage in effective learning activities spontaneously (Kobbe et al., 2007). Therefore, it is advisable to structure the collaborative process in order to foster effective learning. One way of providing such scaffolds is collaboration scripts.

The following section defines the term ‘collaboration script’ and presents several examples. Afterwards, different taxonomies of scripts are presented (section 2.1.3). Furthermore, the markup language IMS-LD is presented, which can be used to specify learning designs (section 2.1.4). Finally, our approach of using collaboration scripts as foundation for this thesis is introduced.

#### 2.1.1 Definition

The term ‘script’ originates from cognitive psychology where it refers to internal memory structures of *‘sequence[s] of actions that define a well-known situation’* (Schank and Abelson (1977), p. 41). For instance, in restaurants we expect a specific sequence of actions: being seated, getting the menu, ordering, and so on. In collaborative learning scripts are usually externally imposed by educators to structure the learning process (King, 2007). A collaboration script defines a

sequence of phases, in each of which a specific task is done in a specific group formation. Dillenbourg (2002) put it this way: ‘*A script is a story or scenario that the students and tutors have to play as actors play a movie script*’.

According to Kobbe et al. (2007) collaboration scripts consist of five components: (1) participants, (2) activities, (3) roles, (4) resources and (5) groups. The first component is the *participants* or the learners for which the collaboration script is designed for. For instance, scripts usually target a certain number of learners (e.g. pairs, groups of four or entire classes). In addition, there can be constraints regarding different characteristics of learners such as age, sex, nationality or prior knowledge. Second, a script defines several learning *activities* such as reading, summarizing or questioning. Third, the participants can be assigned different *roles*, e.g. the role of a teacher or a student. The fourth component is the *resources* or learning material that is handed out to the students or exchanged between the students. Finally, the participants can be divided into *groups*. In addition to the five script components, there are three script mechanisms, which specify how activities, roles and resources are distributed in the group (*task distribution*), how learners are assigned to groups (*group formation*) and how components are distributed over time (*sequencing*). This theoretical framework provides the foundation for any specification of collaboration scripts. It was built by the European research team ‘CoSSICLE’ and is therefore referred to as the *CoSSICLE framework* (Weinberger et al., 2007a).

Before presenting examples of collaboration scripts, the meaning of the terms ‘collaborative’ and ‘cooperative’ should be clarified. For both terms different definitions have been used, which are discussed in detail by McInnerney (2004). According to Dillenbourg et al. (1996) ‘*cooperation*’ means splitting the task among the learners, while ‘*collaborative*’ means all learners work ‘*in a co-ordinated effort*’. Since the majority of literature in the field of collaboration script adopted this definition, the terms will be used accordingly in this thesis.

### 2.1.2 Example Scripts

Research on technology-based collaboration scripts has been very successful in terms of the development of a broad range of scripts that effectively foster processes and outcomes of computer-supported collaborative learning. In the following, several example scripts are discussed, in particular the ones that will play a role in this thesis. A more extensive overview of different collaboration script classes is provided by de Velde et al. (2004). Furthermore, Dillenbourg (2002), Kollar et al. (2006) and Kobbe et al. (2007) provide thorough analyses of several scripts.

#### *Peer Review-Based Collaboration Scripts*

Several collaboration scripts use peer reviewing or questioning to trigger effective learning activities and foster knowledge acquisition (Demetriadis et al., 2011). One example is the M.U.R.D.E.R. script (Dansereau et al., 1979), which facilitates text comprehension. The name is an acronym for Mood, Understanding, Recall, Detection, Elaboration, and Review, which describes the different phases that the learners go through. Students learn in pairs from a textbook.

One learner takes the role of the summarizer, the other one is the listener. After setting the *mood* for studying, both individually read a text passage and try to *understand* it. The Summarizer then *recalls* what has been read while the listener *detects* errors or omissions. After the listener has given feedback on the summary, both learners *elaborate* on the text passage. This process is repeated for the next text passage with switched roles. In the end, the learners *review* the text and reflect on what they have learned.

Another peer review-based collaboration script, which aims at improving text comprehension is Reciprocal Teaching (Palincsar and Brown, 1984). The script instructs learners to engage in four activities: (1) *summarize* a text section, (2) compose *questions* on the content, (3) *clarify* and (4) *predict* how the text continues. These activities are done in small groups of variable size. One group member has the role of a teacher who is responsible for asking the questions and summarizing the text. The script has been compared to traditional teaching methods in two experiments. Results show that Reciprocal Teaching led to greater gains and maintenance over time.

Next to Reciprocal Teaching and the M.U.R.D.E.R. script, there are several other scripts, which are based on peer review, such as ASK to THINK - TEL WHY (King, 1997), a peer assessment script by Trahasch (2004), the social script presented by Weinberger et al. (2005a) or Scripted Cooperation (O'Donnell et al., 1987).

### *Argumentative Knowledge Construction*

Knowledge construction can also be fostered through argumentations. Several collaboration scripts induce socio-cognitive conflict and provide cognitive strategies to solve the conflict. Often the conflicts are simulated by assigning roles, which are in favor/against a certain statement. Similar as in debate clubs the learners are asked to advance a view, regardless of their actual opinions. An example is *Structured Academic Controversy* (Johnson and Johnson, 1994). This script is intended for four learners who collaborate in different group formations. In the first phase, all four learners individually read handouts to prepare their position for the subsequent discussion. Two learners take the pro and the other two take the counter position. In the second phase, two pairs are built, one for each position. In the final phase, the positions are presented to the other dyad, discussed and merged into one final statement.

Another approach of inducing conflict was chosen in the *ArgueGraph* script (Jermann and Dillenbourg, 2003). Instead of assigning roles randomly, learners fill out online-questionnaires, in which they state their opinion. The computer builds pairs of learners who have as different opinions as possible. Each pair is then asked to answer the same questionnaire again but this time the pair has to agree on one answer. That way true conflicts are induced.

In addition, Stegmann et al. (2007b) have proposed a collaboration script that combines two important dimensions of argumentation: (1) the construction of single arguments and (2) the construction of argument sequences. With regard to the construction of single arguments they propose a simplified version of Toulmin (1958)'s argumentation model. As learners rarely provide grounds for their claims, the script prompts learners to provide three components for each

argument. First, the *claim* expresses the position of the argument, for instance ‘I think that smoking is unhealthy’. Second, *grounds* should be added to warrant the claim. Third, a qualification describes the relationship between claim and warrant, for instance to reveal possible limitations or uncertainties. The second script dimension is the construction of argument sequences. Based on a pattern proposed by Leitão (2000) argumentative dialogs should consist of arguments, counterarguments and integrations. More specifically, an argument should be followed by a counterargument, which questions or attacks the previous argument. In the end, learners can synthesize their previous positions in an integration. In a study the script facilitated a higher formal quality of both single arguments and argument sequences.

### *Jigsaw*

Yet another important family of collaboration scripts is Jigsaw. Aronson et al. (1978) proposed to split the class in small ‘Jigsaw groups’ of five to six students. Each group member of a Jigsaw group is given a different subset of information on the school subject and is responsible for sharing it in the group. Prior to this phase all students who received the same information meet in an ‘expert group’ to rephrase the content, discuss possible ways of presenting the information and think about possible questions. This script encourages students to collaborate and help each other as they rely on each others’ information.

There are various modifications of the original Jigsaw script. For instance, the UniverSanté script (Berger et al., 2001) was used to further collaboration between medical students from four different countries. Different clinical cases (e.g. cancer or infectious diseases) were provided to different group members. Again, groups collaborated in different formations. On the one hand, there was a forum that allowed students to collaborate over distance. On the other hand, there were face-to-face meetings in each country. Consequently, UniverSanté integrates online as well as face-to-face activities. Yet another variation is Concept Grid (Dillenbourg and Hong, 2008). First, complementary knowledge is distributed in the group. Then students have to explain the relationship between concepts that are positioned next to each other in a grid. This task requires that all information is shared.

Jigsaw does not play a role in this thesis because the key aspect is the different group formations in a class (except for Concept Grid). Therefore, the above-mentioned scripts are more suitable to investigate our research questions. The Jigsaw family was still presented to provide a more comprehensive overview of different scripts.

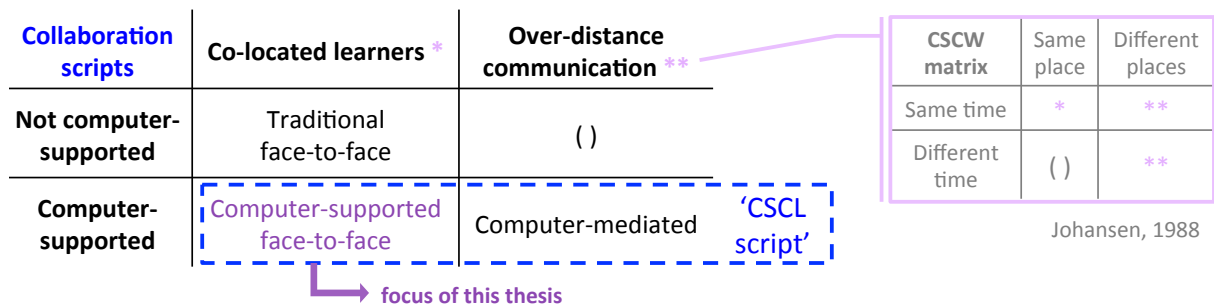
### 2.1.3 Taxonomies

Collaboration scripts can be classified according to several criteria. From an HCI perspective the most important distinctive feature is whether computers are used and how (face-to-face, computer-supported vs. computer-mediated). From an educational perspective important characteristics are the granularity of the script (macro vs. micro) as well as the placement (internal vs. external) and origin of the script (imposed vs. induced).



### *Face-to-Face, Computer-Supported and Computer-Mediated Scripts*

Traditional collaboration scripts, such as the M.U.R.D.E.R. script, have been developed for face-to-face settings in which the tutor or teacher gives instructions based on a collaboration script. Today, most scripts use some kind of computer support and are therefore often called ‘computer-supported collaboration scripts’ or short ‘CSCL scripts’ (Dillenbourg, 2002). As shown in Figure 2.1, CSCL scripts can be further distinguished according to the purpose for which computers are used. Computer-mediated scripts allow learners who are not in the same physical environment to collaborate (asynchronously or synchronously). Examples for computer-mediated scripts have been presented by Baker and Lund (1997), Pfister and Mühlfordt (2002), Trahasch (2004) and Guzdial and Turns (2000). However, computers can also be used to support co-located groups, for instance to convey the script instructions and/or provide the learners with computers while working on the tasks. Most of the example scripts presented above are (traditional or computer-supported) face-to-face scripts because this work focuses on co-located settings.



**Figure 2.1:** Collaboration script classified according to the utilization of computers.

The distinction between co-located and computer-mediated distance communication relates to the time-and-place-matrix, which Johansen (1988) introduced to classify CSCW applications. CSCW applications support groups that are either at the same place or at different places, at the same time (synchronous) or at different times (asynchronous). Transferred to collaborative *learning* applications the matrix can be boiled down to co-located face-to-face learning and (synchronous or asynchronous) distance learning. The category ‘same place / different time’ is not common in collaborative learning.

### *Macro vs. Micro Scripts*

Collaboration scripts can also be classified by their level of granularity (Dillenbourg and Jermann, 2007). Instructions regarding the structure of the collaborative process are regarded as *macro* scripts. An example is Reciprocal Teaching, which instructs learners to engage in a specific sequence of activities (summarizing, questioning, clarifying, predicting). The script does not specify *how* these tasks should be accomplished. This is the purpose of *micro* scripts, which operate on the level of dialogs. For instance, question openers can be used to specify *how* the activity of questioning should be done. Another example is the argumentation micro script, which

specifies how arguments should be built. That way micro scripts guide communication processes. Related to this are the three levels of learning protocols defined by Wessner et al. (1999). On the basic level, learning protocols guide elementary communication processes, on the next level, they support group activities that need some coordination control, and on the third level they support collaboration.

### *Origin and Placement of Scripts*

Yet another way of classifying collaboration scripts is their origin, i.e. where the script comes from and how it is brought to the learners. The above-mentioned example of the restaurant script showed that there are internal scripts, which are used to simplify recurring (collaborative) processes in daily life. Scripts used in collaborative learning are usually external, i.e. brought to the learners by educators (Fischer et al., 2007). This can be done in different ways. Ayala (2007) distinguishes between *imposed* scripts, which are explicitly shown to the learner and *induced* scripts, which are embedded in the learning environment. Furthermore, Carell et al. (2005) compare scripts to maps. While scripts are created by a teacher before the learning session, maps are developed by the students at the beginning of the learning process.

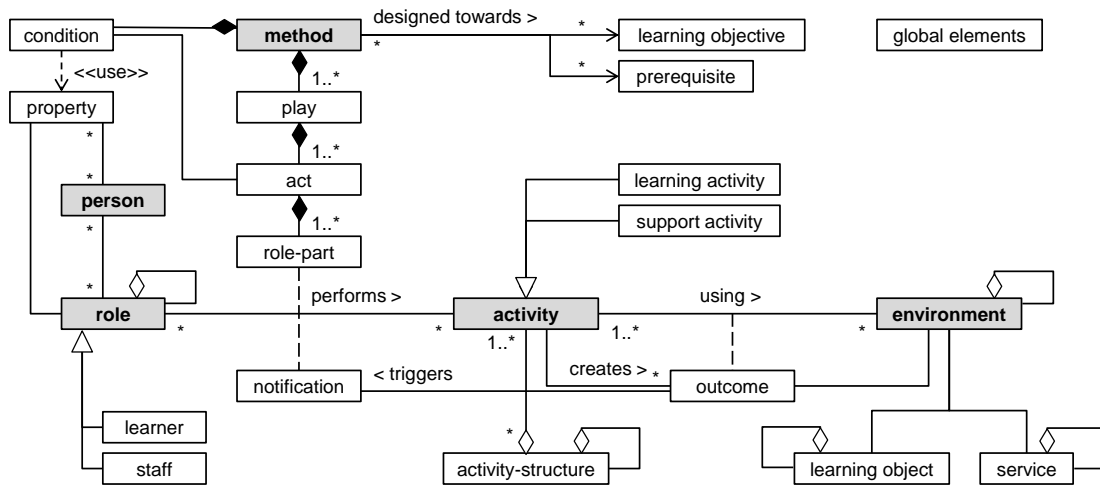
#### 2.1.4 IMS-LD

The IMS Global Learning Consortium (2003) has developed a markup language called *IMS Learning Design (IMS-LD)*, which can be used to specify learning processes. A learning process modelled with IMS-LD typically looks like this (cf. Figure 2.2): A *person* is assigned a *role*. In this role, the learner performs *activities* within an *environment*. The activities could be learning activities or support activities. An example for the latter is a teacher providing feedback or help to the learners. By ‘environment’ IMS refers to the learning objects and services that are used while performing the activity (Koper, 2001). Furthermore, *methods* determine which roles are assigned which activities at what time. In analogy to a theater script methods are structured as plays with acts and role-parts (Hernández-Leo et al., 2005).

Some of these concepts (e.g. roles, activities) are very similar to the components of collaboration scripts. However, IMS-LD is lacking some concepts that the modelling of collaboration scripts would require, for example groups, dynamic features and complicated control flows (Miao et al., 2005). While IMS-LD is still the de facto standard, tools and frameworks are forced to use workarounds to compensate for the shortcomings of IMS-LD when it comes to specifying collaboration scripts (cf. Malzahn et al., 2008).

#### 2.1.5 Collaboration Scripts as Foundation for our Methodology

Collaboration scripts were used to get an educational perspective on collaborative learning for several reasons. First, as the research field is several decades old, there is a collection of scripts



**Figure 2.2:** Conceptual model of overall learning design (IMS Global Learning Consortium, 2003).

that have proven to be effective. Second, collaboration scripts provide a detailed description of collaborative learning scenarios, which have proven to be beneficial for learning. The specification includes important educational aspects that need to be kept in mind when thinking about suitable user interfaces. Furthermore, the theoretic framework proposed by Kobbe et al. (2007) allows for general approaches across collaboration scripts.

For these reasons, collaboration scripts were chosen as foundation for this work in two ways. First, a framework for collaboration scripts was developed, which facilitates a rapid development process of scripted learning applications. The framework provides user interfaces for all involved stakeholders: educators, programmers, teachers and learners. Second, collaboration scripts were used as instrument for the investigation of our research questions. Starting with a scripted learning scenario that could benefit from technology, prototypes were designed and implemented in interdisciplinary projects. The teams consisted of researchers from the learning sciences as well as from human-computer interaction. The prototypes were then used to conduct experiments to investigate research questions from both fields.

## 2.2 Collaborative Learning Applications

The following section presents collaborative learning applications. The focus is on applications for co-located settings. The applications are viewed from an HCI perspective and consequently grouped by their user interface technologies. At first, tabletop applications are presented. Second, tangible user interfaces are discussed as they are often used to augment tabletop applications. The third subsection is on mobile applications - a field, which has received increasing attention over the last years due to the prevalence of mobile and smart phones. All these user interface

technologies are limited in the number of users that can interact simultaneously. The last subsection therefore discusses different ways of supporting entire classrooms, usually by combining different types of technologies.

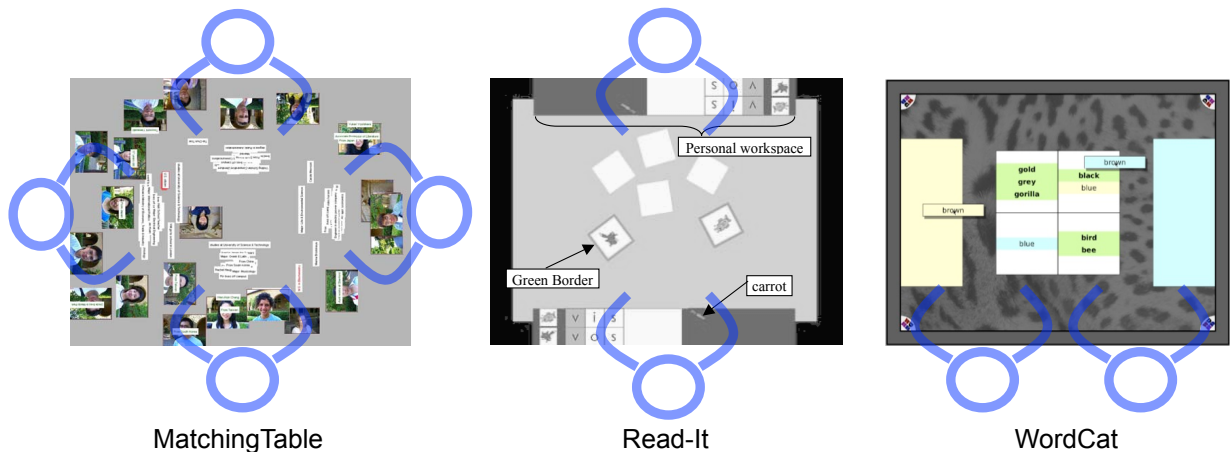
### 2.2.1 Tabletops in Education

Interactive tabletop displays are often used to support collaborative learning in small groups because they offer shared workspaces in which several people can directly interact. Researchers have implemented a variety of tabletop applications targeting a broad range of learning goals. At first, text-based tabletop applications are presented. These applications have common problems regarding text input and orientation, which play an important role in this work. The second part gives an overview of a wider range of applications.

#### *Text-Based Tabletop Applications*

Morris et al. (2006a) have implemented three applications to support language learning. The (1) *MatchingTable* lets learners match text with images (see Figure 2.2.1). On the (2) *ClassificationTable* students sort vocabulary to the four corners of the table, each of which represents a specific property. The (3) *PoetryTable* allows learners to create sentences from word tiles. The three applications have a very similar setting. Learners sit around the table facing each other. A common problem in this situation is that some learners have to read the text upside down, which impairs readability. This is particularly problematic in language learning applications because they heavily rely on text. On the *MatchingTable*, *ClassificationTable* and *PoetryTable* phrases and images automatically rotate to face the nearest edge of the table. The assumption is that learners will mainly interact with items in their direct proximity. This functionality is supported by the *DiamondSpin* tabletop interface toolkit (Shen et al., 2004). Another common ground is that all three applications are implemented on the *Diamond Touch* table (Dietz and Leigh, 2001), which allows for user identification. Its surface is equipped with antennas. Once a user touches the surface (and thus the antennas) a signal is transmitted through the user's body to a receiver that is installed in the work environment (e.g. the user's chair). This user identification system can be used to install multi-user coordination policies.

Another text-based learning application with a similar learning goal is *WordCat* (Rick, 2009). Two learners have to sort twelve words in a 2x2 matrix in a way that both columns and rows build categories. Here, the learners sit side-by-side to avoid the text orientation problem. Yet another solution was used in the *Read-It* application (Sluis et al., 2004). *Read-It* is an augmented tabletop application for children from five to seven years, which supports the development of reading skills. It teaches elementary reading processes and offers playful training games, e.g. the 'memory' game (see Figure 2.2.1). To deal with the text orientation problem, the space is divided into a shared workspace, which only contains pictures for which the orientation problem is less severe. To avoid confusion of letters, text is only shown in the personal workspace, which is directly in front of the child. Another design decision was to use brick elements instead of virtual memory cards. That way the interaction with the application is very similar to playing



**Figure 2.3:** Different solutions to the readability problem: On the *MatchingTable* (Morris et al., 2005) text and images are automatically oriented to the closest edge [left], *Read-It* shows text in personal workspaces (Sluis et al., 2004) [center], *WordCat* (Rick, 2009) uses side-by-side seating [right].

with real cards and no new interaction methods need to be learned. To allow for object detection, the tangibles are tagged with infrared reflecting tape.

### *Other Tabletop Applications*

Many other learning applications abstain from text on the tabletop display. For example, the *StoryTable* (Cappelletti et al., 2004) is another playful application, which supports cooperative story telling. In contrast to the text-based applications above, the *StoryTable* allows students to record audio snippets. Next to language and reading training there are many tabletop applications that teach math or science. For instance, using the *DigiTile* (Rick and Rogers, 2008) learners can collaboratively create quilts. During this process the application challenges learners to solve mathematical problems. Another example is *Quadratic* (Rick, 2010), which allows learners to explore algebraic expressions on a tabletop. Moreover, Falcão and Price (2009) built a tangible tabletop application that supports the exploration of the physics of light. A similar application is *Illuminating Lights* (Underkoffler and Ishii, 1998). In addition, tabletops have been used in apprenticeship. For instance, the *TinkerTable* (Zufferey et al., 2009) allows apprentices to construct warehouses with tangible shelves. The efficiency of the warehouse can be examined in simulations.

Furthermore, social skills is an important learning goal that needs to be mentioned. While collaborative learning application fosters social skills to some extent, there are some applications, which specifically target social aspects. For instance, *SIDES* (Piper et al., 2006) was designed for adolescents with Asperger's Syndrome who have difficulty in understanding social norms. *SIDES* is a non-competitive but highly collaborative game that aims at teaching social skills such as active listening and turn-taking.

In summary, the diversity of the presented examples shows that tabletop displays have been used to pursue various learning goals in the field of co-located collaborative learning.

### 2.2.2 Tangibles

Several tabletop applications presented in the last section use tangible user interfaces (TUI), which means that users are offered tangible objects to control the digital system (Ullmer and Ishii, 2000). For instance, Sluis et al. (2004) chose brick elements instead of virtual memory cards for the Read-It application. Their rationale was to make the interaction with the application more similar to playing with real cards. That way no new interaction methods need to be learned. For the same reason Falcão and Price (2009) used real pocket lamps and tangible objects as representations for the source of light (see Figure 2.4).



**Figure 2.4:** Exploring the physics of light (Falcão and Price, 2009)



**Figure 2.5:** TICLE (Scarlatos, 2002)

Another reason why tangibles are often used in education is that direct manipulation of physical objects is believed to be beneficial for learning as such (Marshall, 2007). For instance, understanding three-dimensional forms or haptic perception is much easier with real objects than with virtual representations. Furthermore, people are used to interact with physical objects, which is why TUIs are often considered to be more intuitive to use than other forms of user interfaces. Consequently, tangibles have a low barrier to participation, which makes them particularly suitable when it comes to engaging children in playful learning or inviting novice users to participate. This has for example been exploited by Horn et al. (2009), who permanently installed a tangible user interface to control a robot in the Boston Museum of Science. Wooden jigsaw puzzle pieces carrying commands like ‘shake’, ‘rotate’, or ‘growl’ are used by visitors to control the robot. That way visitors playfully take their first steps in learning how to program.

Similarly, Suzuki and Kato (1994) created a tangible programming language called ‘*AlgoBlock*’, which provides tangible cubes to a group of learners. In this case, tangibles were specifically chosen to support *collaborative* learning as they can be shared by groups and facilitate a high awareness of the groups’ activities. Another tangible interface for collaborative learning environments is *TICLE* (Scarlatos, 2002). A QuickCam tracks tangible Tangram puzzle pieces that are

tagged with reflective markings. A nearby display shows hints and/or rules. Marshall (2007) provides an extensive literature on tangibles for (collaborative) learning and discusses the possible benefits. However, so far there is little empirical evidence for the assumed positive effects.

### 2.2.3 Mobile and Handheld Applications

Despite limitations such as the small display size, mobile phones and handheld devices have been used for collaborative learning in various forms. Naismith et al. (2004) even identified collaborative activities as one main form of learning activities for mobile learning technologies. For instance, *ImageMap* (Roschelle and Pea, 2002) runs on handheld devices that are distributed among students in a class. The students receive images and answer questions about the images. The information is aggregated and made available on a shared display for discussions. Other systems use virtual shared workspaces instead of physical shared displays. For instance, *MediaBoard* (Colley and Stead, 2004) allows students to solve assignments collaboratively. The mobile phones are used to contribute content and knowledge to an online project via SMS, MMS and web access. Another purpose of using mobile phones in the context of collaborative learning was presented by Nakahara et al. (2005) who suggested to show the learners' participation levels on the wallpaper of their mobile phones. Again, the actual collaboration happens in a virtual shared workspace, in this case an online *bulletin board system*. Bulletin board systems are frequently used by learners for discussions, information exchange and collaboration in the context of projects or traditional classes.

A more extensive system is *handheld learning resource (HandLeR)* (Sharples, 2000), which offers virtual laboratories, classrooms or simulations on a tablet computer. HandLeR aims at supporting life-long learning and is therefore not bound to traditional classrooms. Furthermore, there exist a number of mobile applications that facilitate collaborative museum visits (cf. Brown et al., 2003 or Kukulska-Hulme, 2009). Transferring the idea from museum to whole cities, Calori and Divitini (2009) build a mobile collaborative learning application for what they call 'city-wide learning'.

Due to their availability mobile applications have also been used to facilitate education in developing countries (Motlik, 2008). In this context Jain et al. (2011) raised the question whether single or multiple displays should be used to support collaborative learning in development regions. They built two applications for the collaborative word game *Boggle<sup>TM</sup>* using two different setups. In the single-display setup each student is given a joystick to interact with a shared display. In the multi-display setup each learner is given a handheld device. A comparison showed that in the single-display setup (Figure 2.2.3, left) the teacher could more actively facilitate the collaborative game play compared to the multi-display setup (Figure 2.2.3, right). Another advantage over the multi-display setup was that children scored better and were more aware of their co-learners when using a shared display.

These examples show that mobile and handheld devices have been used for collaborative learning in various ways. However, because they are single-user devices, they are usually combined with



**Figure 2.6:** Single (left) and multi (right) display setup for collaborative word gaming (Jain et al., 2011).

virtual shared workspaces (such as bulletin boards) or physical shared workspaces (i.e. shared displays).

## 2.2.4 Classroom Technologies

The learning applications discussed so far are suitable for small groups of learners. This section discusses how technologies can support entire classrooms. Interactions and technologies in the classroom can be split to the following levels:

1. *Classroom level:* New forms of blackboards such as interactive whiteboards are used to address the entire class.
2. *Group level:* Shareable user interfaces are used by small groups, for example in breakout sessions. Here the above-mentioned technologies such as tabletop displays or tangible user interfaces come into play. It is also possible to use interactive whiteboards at this level.
3. *Individual level:* Personal devices such as remote controls, mobile phones or handheld devices can be used to enhance interactivity in the (physical) classroom. An example is group decision support systems that are applied in education. Instead of choosing one student to answer the teacher's question such systems allow all students to individually answer the teacher's question using the personal device. The answers of all students are then aggregated and discussed in the class. One of the first applications in that direction is Classtalk (Dufresne et al., 1996).

In addition, *computer-mediated interactions* play a role in many technology-supported classrooms. Computer networks enable students to interact with each other besides the face-to-face interaction that happens in the classroom. An example is the above-mentioned bulletin board system, which can be used by learners to collaborate, discuss and exchange information online. Bulletin board systems are one form of virtual classrooms. More comprehensive solutions are





**Figure 2.7:** Embedded Phenomena: HelioRoom [left] (Moher, 2006), RoomBugs [center] and RoomQuake [right] (Barron et al., 2006)

course management systems such as Moodle<sup>1</sup>. On the one hand such computer-mediated interaction can be used on top of face-to-face learning, for example to support collaboration on homework assignments. On the other hand, there are also various applications for pure distance learning (Simonson et al., 2000).

Many technology-supported classroom combine several of the above-mentioned categories. For instance, in the *Classroom 2000* (Abowd, 1999) the teacher shows slides on a pen-based digital whiteboard. In addition, students can see the slides on tablet computers. The ‘ClassPad’ application running on the tablets allows students to write down notes on the slides. In addition, the students as well as the teacher’s slides are combined to a series of web pages after the lecture. A similar approach is the *Highly Interactive Classroom* (Liang et al., 2001). It allows students to participate in learning activities using an infra-red remote controller. There is also a shared display in the form of an electronic whiteboard. Liu et al. (2002) replaced this controller with a WLAN-based handheld, which supports several more types of interaction.

The Highly Interactive Classroom and the Classroom 2000 are rather general approaches that can be applied to basically any kind of (slide-based) learning material. Other approaches instrument the classroom in a way that addresses a specific learning goal. An example is *Embedded Phenomena* (Moher, 2006) where scientific phenomena are mapped onto the physical classroom space. Tablet computers positioned around the classroom are used to show scientific information over the period of several weeks. For example, *HelioRoom* simulates the movement of the planets. The tablets are located at fixed positions at the walls and serve as viewports to observe the planetary motion. A second example is *RoomBugs*. Here, the tablets represent sandboxes, which insects live in. Students can introduce pesticides, control the humidity, leave marks etc. Finally, *RoomQuake* is an earthquake simulation system. The classroom acts as an active seismic field. Seismograms reflect the current vibration (see Figure 2.2.4).

These few examples already demonstrate that there are various ways in which technology can facilitate collaborative learning in classrooms. Most of them use combinations of different displays

<sup>1</sup> moodle.com

(personal devices such as tablet computers in addition to shared group and classroom devices). That way different group formations are supported: individual work, small-group as well as classroom interaction. In addition, many applications also consider interactions outside the classroom, for instance bulletin boards.

## 2.3 The Influence of the Display (Environment)

As demonstrated in Section 2.2, there is a great variety of user interface technologies that have been used to support collaborative learning: mobile devices, interactive tabletop and wall displays, personal computers, tangible user interfaces and more. What all these applications have in common is the presence of at least one display. The types of displays vary in several factors, for instance size (from very small displays in mobile phones to large interactive surfaces), orientation (horizontal tabletop displays vs. vertical wall displays) or input technology (direct touch, indirect pointing devices or tangibles). Furthermore, some applications use one shared display for the whole group, while others use one personal display for each learner. Yet others combine different displays to ‘display environments’.

All these factors have a potential influence on the collaborative process. This section first classifies different display environments. Then several studies are presented, which investigated effects of the choice of display (environment) on collaborative processes.

### 2.3.1 Display Types and Display Environments

Displays and display environments can be divided into three major categories. First, there have been various attempts of using *personal computers* to support collaboration. In addition, there are computer systems for groups, which are built of one shareable user interface (*‘Single Display Groupware’*) and computer systems that consist of several displays (*‘Multi-Display Environments’*). In the following the characteristics of these categories are analyzed. In contrast to the last section, which focused on CSCL applications, the following section will widen the scope to include research on human-computer interaction as well as computer-supported collaborative work (CSCW).

To describe the characteristics of display environments, the terms ‘personal’, ‘private’ and ‘shared’ workspaces will be used, following Tandler et al. (2001). These terms play an important role throughout this thesis. A *personal* workspace belongs to a single learner. Its content may be visible to other people, but there is one owner who (mainly) edits it. For example, parts of a tabletop display might be implicitly or explicitly dedicated to one group member. A *private* workspace is a personal workspace, which can only be seen by its owner. A typical example is a laptop as its content is usually only visible to its owner (depending on the seating arrangement). *Shared* workspaces can be viewed and accessed by the whole group, e.g. interactive surfaces.

### *Personal Computers*

Single-user devices such as desktop computers, laptops or tablet computers have been repeatedly used to support collaborative learning. This particularly applies to computer-mediated learning (e.g. Edelson et al., 1996) but single-user devices can also be found in co-located settings (e.g. Haué and Dillenbourg, 2009). While in computer-mediated learning each learner needs to have an own device, learners can share devices when they are co-located. However, sharing single-user devices can be problematic. Druin et al. (1997) found that collaborating children can get frustrated if only one mouse is available because they have to decide who is in control. A potential problem for adults is that they might be forced to sit closer together than they feel comfortable with (Hall, 1966). Tablets usually use direct touch, which means that users manipulate objects directly on the surface using fingers or pens. Sometimes even multi-touch is supported, which allows multiple people to interact simultaneously. However, in comparison to single display groupware the display size is very limited. Therefore, tablets have been combined with additional devices such as projectors for classroom instruction (Mock, 2004).

Due to these limitations of personal computers, Kaplan et al. (2009) argues that they ‘[...] are not efficient tools to learn in group.’ However, integrating private workspaces into learning environments can be beneficial. For instance, users might be reluctant to interact in shared workspaces due to fear of criticism (cf. Warr and O’Neill, 2006 and Looi et al., 2008).

### *Single Display Groupware*

To support co-located collaboration, technologies that allow for ‘*collective co-located manipulation*’ and ‘*group awareness*’ are needed (Kaplan et al., 2009). This is supported by single display groupware, which allow multiple users to simultaneously interact on a shared display. That way production blocking can be reduced (Prante et al., 2002).

The shared display can either be vertically oriented (e.g. interactive whiteboards or wall displays) or horizontally oriented (interactive tabletop displays). Vertical displays provide the same perspective to all group members. There are different input technologies. A very common one is direct touch. Such interactive whiteboards are, for example, widely spread in classrooms in Great Britain (Brown, 2003). An alternative approach is manipulation through mice and keyboards. However, if only a single mouse or keyboard is provided, this can negatively affect group interactions because only one user can interact with the display at a time. To deal with this problem, *Dynamo* (Izadi et al., 2003) allows users to simultaneously interact with an interactive surface using multiple mice and keyboards. Each user has an own colored mouse cursor and a personal ‘palette’, which is shown on the bottom of the interactive surface. The palettes provide access to USB disks or email and therefore act as media sources and media sinks. This allows to display different media types (audio, video, websites, 3D graphics, maps, etc.) simultaneously.

As for horizontal displays, direct touch is suitable for most forms of interaction because the users are usually in direct proximity. An exception is text input, which can be quite challenging on tabletop displays. There are various solutions such as physical keyboards, soft keyboards, handwriting recognition or gestural alphabets, but all of them have drawbacks. While physical

keyboards have the best performance, they can be a barrier between users and interaction spaces (Hinrichs et al., 2007). Another issue is the placement of control widgets. Morris et al. (2006b) compared the use of centralized vs. replicated controls on a tabletop groupware application. Although 80% of the participants preferred the replicated controls, the centralized controls lead to a significantly higher number of users that participated in the task.

Furthermore, a distinctive feature of tabletop displays is the existence of different types of workspaces, which automatically form in around-the-table seating arrangements. Scott et al. (2004) conducted two observational studies, through which they identified three workspaces on tabletops: personal, group and storage. The personal workspaces are located along the edges in direct proximity of the users, whereas the group space was located in the center. The location, size and shape of the storage spaces depended on the content and activities. Another approach to the definition of personal workspaces was taken by the Mitsubishi Electric Research Laboratories (MERL) by developing the *DiamondSpin* toolkit for ‘around-the-table’ interaction (Shen et al., 2004). It allows arbitrary document positioning and orientation and supports multiple (personal) work areas on a tabletop display. By default each document is oriented towards the closest outside edge of the tabletop (cf. MatchingTable, Classification and PoetryTable in section 2.2.1). The idea is that every participant can read the documents closest to him but they can be rearranged by simple drag-and-drop gestures. There are different orientation modes, e.g. Table for Two/Four/Eight, which create a clear assignment of documents to participants through a discrete orientation (instead of a continuous one). That way multiple personal work areas are created. There is no shared work area. Instead the tabletop surface can be dragged, which allows rotating the document altogether, and of course single documents can simply be dragged into another person’s work area.

Finally, several researchers have suggested that horizontal displays require supplemental vertical displays. For instance, Shen et al. (2009) found that ‘[...] *the most compelling moments occur when the tables interoperated with vertical displays and other devices*’. There are various reasons why tabletops should be enhanced with other types of displays. For instance, in around-the-table seating arrangement different user have different perspectives on the content. This can impair comprehension, in particular if users need to read text that is not correctly oriented (Kruger et al., 2003). Furthermore, graphical elements might be distorted on the tabletop (Wigdor et al., 2007).

### *Multi-Display Environments*

The previous subsections have already pointed out possible benefits of combining both personal computers as well as single display groupware with other displays. Hutchings and Pierce (2006) identified several reasons for dividing an application’s interface across devices. The most important ones are to obtain privacy, to share information, to establish different views of the same information and to improve usability by choosing a device based on the quality of input and output capabilities. This is supported by multi-display environments (MDE), which are ‘[...] *comprised of co-located personal (e.g. laptops) and shared devices (e.g. large displays) that are networked to form an integrated virtual workspace*’ (Biehl et al., 2008). One of the earliest systems to combine private and shared workspaces is *DOLPHIN* (Streitz et al., 1994). In this case

the shared workspace is an interactive electronic whiteboard. Personal computers serve as private workspaces. All participants can access the shared workspace using a duplicated version that is located on their personal computers, which can be edited using a WYSIWIS (What You See Is What I See) editor. Another early approach was done by Greenberg et al. (1999), who combined personal (PDA) and public displays in a note taking application and later created a 'Notification collage', which randomly places incoming media elements onto a shared collaborative surface (Greenberg and Rounding, 2001).

While the first approaches used vertical (wall) displays as shared workspaces, the *UbiTable* (Shen et al., 2003) combines laptops with a horizontal tabletop display. It is designed for spontaneous, unplanned collaboration (e.g. at the airport). The tabletop display has a shared workspace in the center, which allows users to share and copy files. In addition, there is a digital ink function for simple sketches and notes. On the edges there are personal workspaces, which enable file transfers from and to the laptops.

In addition, there are MDEs that combine interactive wall and tabletop displays and thus provide more than one shared workspace. An early approach in this direction is the *iRoom* (Johanson et al., 2002). Later the interaction framework *IMPROMPTU* (Biehl et al., 2008) was developed to allow users to share any off-the-shelf applications across multiple displays. The *IMPROMPTU* user interface consists of (1) a collaboration control, which reveals three sharing states: Show, Share, Neither nor, (2) a collaborator bar, which shows each collaborating user and his shared application windows and (3) a shared screen dock, which allows users to place application windows on a shared display. Yet another approach is *WeSpace* (Wigdor et al., 2009). It also consists of a large wall display and a direct-touch tabletop display. While *IMPROMPTU* shares application windows, in the *WeSpace* the entire desktops of users are displayed in the shared workspace as live images. Furthermore, *IMPROMPTU* uses the laptops' pointers for interaction, whereas the *WeSpace* offers several interaction methods: The wall displays can be either controlled through the laptop (using mouse and keyboard), or through the multi-touch tabletop display. The *WeSpace* Layout Manager enables users to control screen images of laptops, which are shown on the shared surfaces. Laptops can be classified as important (large screen image), public (small screen image) or private (no representation on the shared displays). This allows the participants to control their privacy and visibility levels.

### *Multi-Display Reaching Techniques*

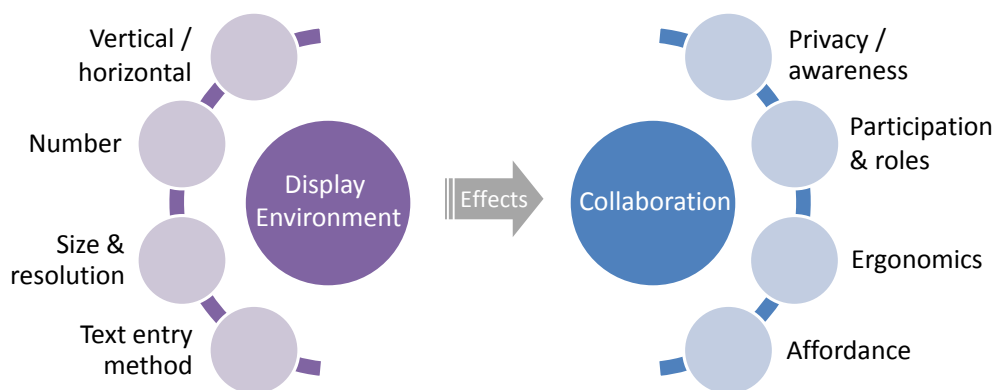
In MDEs multi-display reaching techniques are required to establish interaction across the different displays (Nacenta et al., 2006). An early approach was Hyperdragging (Rekimoto and Saitoh, 1999), which uses a camera-based object recognition system to allow users to integrate their laptops in an environment with pre-installed projected walls and tables. When the user moves his mouse cursor to the edge of a screen, the cursor will not stop there, but will move on to the physical surrounding, for example a table. *Pick-and-Drop* (Rekimoto, 1998) is similar but uses a stylus instead of a mouse. Digital objects can be picked up by tapping on it, moved to the target position and dropped there with another tap. Other multi-display reaching techniques use gestures, e.g. Slingshot (Hascoët, 2003). Yet another approach is RadarView (Swaminathan and

Sato, 1997), which shows miniature representation of devices in the environment once a digital object is tapped. The *Gateway Interaction Technique* (Guinard et al., 2007) follows a similar idea. So-called *gateways* are positioned at the edge of the screen to represent nearby devices. The position of the gateway on the screen points to the direction in which this device is located (similar to a compass). That way, devices can be searched for, which is very useful for identifying pervasive services that are unknown to the user. The Gateway Interaction Technique relies on a positioning system to determine the relative position and orientation of co-located devices.

Nacenta et al. (2007) compared different interaction techniques in terms of conflict rates, performance as well as transfer and reaching patterns. All techniques had strengths and weaknesses and their utility largely depends on the task at hand.

### 2.3.2 Effects of Displays on Collaborative Processes

In the field of human-computer interaction (HCI) a number of studies have shown that the type of display (environment) can affect collaborative processes. Some of the display characteristics that play a role are the number, orientation, size and privacy of a display, the seating arrangement, the users' proximity to the display, as well as whether the input occurs in the same physical space (Mandryk et al., 2002). Several of these aspects also apply to co-located collaborative learning scenarios. First, depending on the number, size and orientation of a display, it can offer private workspaces or be visible to the whole group, which increases the awareness of the learning partner's activities. Second, the balance of participation and role distribution within the team can be affected by the proximity of different learners to the display or input device. Third, there are ergonomic aspects such as arm fatigue or limited elbow-room. The result of all these factors combined is that different types of displays are used for different purposes and thus have different affordances. An overview of the display factors and their influence on collaboration is shown in Figure 2.8. In the following these effects will be discussed in detail.



**Figure 2.8:** Overview of display factors and their influence on collaboration

### *Privacy vs. Awareness*

The privacy of a display affects the level of awareness in the collaborative environment (Mandryk et al., 2002). Shared displays allow group members to monitor each other's activities and can therefore ease coordinative processes. However, there are a number of reasons for integrating private workspaces into collaborative environments. For instance, an increased awareness of other people's activities could also distract learners from their primary task. This can impair the outcome of certain collaborative processes. For instance, it has frequently been found that groups of individuals create more ideas than brainstorming groups (Diehl and Stroebe, 1987). To avoid the underlying effects such as free riding and production blocking, electronic brainstorming systems have successfully been applied (Gallupe et al., 1991). Similarly, private workspaces could enhance productivity in other cases as well. Furthermore, learners often appreciate private workspaces, which is why many collaborative environments provide a mix of private and shared spaces (e.g. Baghaei et al., 2007). These examples show that there is a trade-off between privacy and awareness when choosing between private and shared displays.

Next to the awareness of the co-learners' interactions with the application, eye contact is an important factor in face-to-face collaboration (cf. Shell et al. (2004) and Nardi and Whittaker (2002)). On tabletop displays the amount of eye contact is affected by the seating arrangement. In a study conducted by Inkpen et al. (2005) groups sitting in a right-angled condition had significantly more eye contact compared to sitting side-by-side or face-to-face. The spatial arrangement can also improve the performance and collaboration (Buisine et al., 2012).

### *Balance of Participation and Role Distribution*

The integration of personal devices in multi-display environments can have valuable benefits (Rogers and Rodden, 2003). For instance, they provide private workspaces, which allow group members to create content privately before sharing it with others. That way potential social embarrassment can be avoided, which is a common reason for being reluctant to interact. Empirical studies on computer-supported learning confirm that combining shared and private workspaces can facilitate more equal participation in face-to-face collaborations. In a study students were provided with tablet computers on which a tool called Group Scribbles (Looi et al., 2008) was implemented. As expected, verbal discussions were dominated by more outspoken students. However, introverted students shared their ideas by posting notes to the group space. The graphical user interface of Group Scribbles is divided into a group space in the upper pane as well as a private space in the lower pane. This allows students to first create virtual notes (so-called 'scribble sheets') in their private workspace without risking embarrassment. When they are satisfied with the result the scribble sheet can be moved to the upper pane, which is synchronized across all tablets (Looi et al., 2008).

Another aspect that can be affected by the display configuration is the role distribution within the team. For instance, Rogers and Lindley (2004) found collaboration to be more balanced on horizontal displays compared to vertical ones. While the role of the writer was mostly fixed in groups that collaborated around wall displays, the person who wrote on the tabletop display usually switched. When asked about their behavior, participants mentioned that it is 'socially

awkward’ to ‘explicitly offer the pen to the others’. On the contrary, the pen can easily be put down on the tabletop displays, where any other group member can pick it up. Similar observations have been made by Everitt et al. (2006), who also found collaboration to be democratic around tabletop displays.

Furthermore, the seating arrangement can influence the role distribution within groups. Side-by-side is perceived as the most intimate seating arrangement and preferred by collaborating teams whereas face-to-face seating is often chosen by competing teams (Sommer, 1967). Some recent systems designed their spatial arrangement according to these findings (Kauko and Häkkinen, 2010).

Finally, there is the question of how many people are enabled to control the application. Traditional single-user computers are restricted to one mouse and keyboard. If such devices are used for collaboration, asymmetrical participation is likely because only one person can control the application at a time. In other words, the limitation of ‘*entry points*’ can cause a barrier to balanced participation. Shareable user interfaces on the other hand allow multiple users to interact at the same time. Rogers et al. (2009) examined whether shareable interfaces promote more group participation as compared to single-user displays. They did an experiment in which they varied the constraint of entry points. The most constrained condition was a laptop with a single mouse. The second condition was a multi-touch table. The condition with the most entry points offered a multi-touch table and tangibles to the group. A qualitative analysis showed more balanced participation in terms of verbal utterance in the least constrained condition (tabletop plus tangibles).

### *Ergonomic Aspects and Human Factors*

Inkpen et al. (2005) conducted an exploratory study to examine the effects of the seating arrangement as well as orientation, size and number of displays on collaborative processes. Regarding the orientation of the display they found ergonomic difficulties in the interaction with vertical displays, such as difficulty in writing and back stiffness, while horizontal displays were considered comfortable and natural. On the other hand, the perspective was found to be a strong advantage of vertical displays.

Furthermore, when designing tabletop applications, the size of a tabletop display as well as the seating arrangement should be taken into account. Ryall et al. (2004) found that users appreciate elbow-room, which can be limited on small tabletop displays. On the other hand, it is difficult to reach certain parts of very large tabletop displays (Inkpen et al., 2005). Therefore, the range of reasonable tabletop sizes is rather limited. Consequently, Ryall et al. (2004) recommend to provide additional vertical displays for larger groups. Further problems on tabletops can be occlusion (Shen et al., 2009) and orientation (cf. Section 2.2.1). Orientation is not only critical to comprehension, it also affects how people coordinate their actions as well as how they communicate (Kruger et al., 2003).

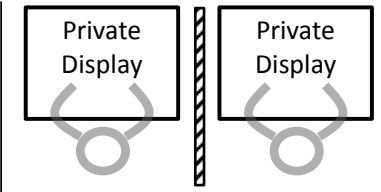
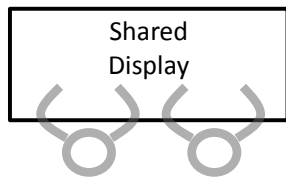


### Display Affordances

The term ‘*affordance*’ was coined by Gibson (1977) and was later applied in the field of human-computer interaction (Norman, 2002). Affordance means that the appearance of the object suggests how it should be used (i.e. ‘affords’ a certain behavior). With regard to displays, the different display characteristics cause different affordances, which means that some displays are more suitable for specific tasks than others. For instance, nobody would doubt that large wall displays or projections are ideal tools for presenting content to a larger audience. Although the affordances of displays for collaborative tasks in small groups is not always that obvious, previous research has shown some tendencies. For example, tabletop displays “[...] *afford[s] writing, face to face communication, territoriality (Scott et al., 2004), and group awareness, while a wall display allows and supports shared visualization, overview and context awareness*” Hilliges et al. (2007). Observations in multi-display environments such as the MultiSpace (Everitt et al., 2006) confirm that different display types are used for different purposes. For instance, organizational tasks, such as arranging and sorting items, were frequently done on tabletop displays whereas wall displays were rather used for comparisons. This suggests that the different characteristics of displays create different affordances, which is not surprising considering the above-mentioned display characteristics and their effects.

### Effects of Private and Shared Displays

As the role of private and shared displays is particularly important for this thesis, Figure 2.9 recapitulates those effects that are related to the presence of private or shared displays. The effects have been addressed above in more detail.

		
<b>+</b>	<ul style="list-style-type: none"> <li>• Privacy appreciated</li> <li>• More balanced participation</li> <li>• Efficient parallel work</li> </ul>	<ul style="list-style-type: none"> <li>• Direct collaboration in joint workspace</li> <li>• Awareness of other people’s activities</li> </ul>
<b>-</b>	<ul style="list-style-type: none"> <li>• No direct (,on-display’) collaboration</li> </ul>	<ul style="list-style-type: none"> <li>• Social embarrassment</li> <li>• Evaluation apprehension</li> <li>• Blocking effects</li> <li>• Freerider effects</li> </ul>

**Figure 2.9:** Strengths and weaknesses of private and shared displays.

An important benefit of private displays is that privacy is often appreciated (Baghaei et al., 2007). Furthermore, it can lead to more balanced participation as the private displays provide less outspoken learners with an alternative way to participate (Looi et al., 2008). In addition, private

displays are efficient for parallel (i.e. cooperative) work in which group members pursue different tasks. A strong disadvantage of private displays is the lack of direct ('on-display') collaboration, which is at the same time the strongest advantage of shared displays. Another advantage of shared displays is an increased awareness of the partners' activities. Drawbacks of shared displays can be social embarrassment (Rogers and Rodden, 2003), evaluation apprehension (Looi et al., 2008), blocking effects (Hymes and Olson, 1992) and freerider effects (Diehl and Stroebe, 1987).

# III

## DISTRIBUTED CLASSROOM APPLICATIONS



# Chapter 3

---

## From Graphical Learning Designs to Scripted Learning Applications

As demonstrated in section 2.1, research on collaborative learning led to a broad range of collaboration scripts, which effectively foster learning processes and outcomes. In order to describe such scripts Kobbe et al. (2007) proposed a generic framework according to which collaboration scripts can be described as a specific configuration of five components (participants, activities, roles, resources, and groups) and three mechanisms (task distribution, group formation, and sequencing). Based on this groundwork, the following chapter<sup>1</sup> presents a functional framework that supports rapid development processes of computer-supported collaboration scripts: the ‘eXtremely Simple Scripting’ (XSS) Framework<sup>2</sup>. Its goal is to optimize the development process by

1. providing part of the functionality to reduce the programmers’ work load
2. establishing a continuous development process that supports all steps from creating a script to developing the application to running the application in the classroom, and
3. providing all domain experts who are involved in the development process as well as tutors with suitable user interfaces.

---

<sup>1</sup> Parts of this chapter have been published. The M.U.R.D.E.R. application has been presented in Streng et al. (2009b), the XSS Framework in Streng et al. (2011a).

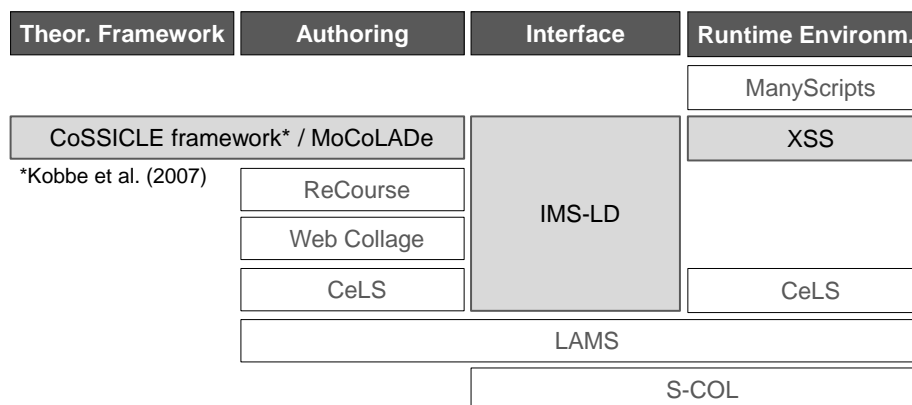
<sup>2</sup> Parts of the implementation of the XSS Framework were conducted by Jonas Koch (2009), Simon Lutzenberger and Wenqi Zhang (2010).

In the following, related work on the development of scripted collaborative learning applications is summarized (section 3.1). Afterwards a scripted collaborative learning application called M.U.R.D.E.R is presented. It is an extension of the original M.U.R.D.E.R. script developed by Dansereau et al. (1979). The M.U.R.D.E.R. application was developed in parallel to the framework and serves as a practical example throughout the chapter. For instance, using the example of the M.U.R.D.E.R. application it is demonstrated how collaboration scripts define the structure of scripted learning applications. This is exploited by the eXtremely Simple Scripting (XSS) Framework which automatically configures parts of the learning application based on the information specified in the underlying collaboration script. Section 3.3 explains how the automatic configuration works, illustrates the resulting development process and introduces a number of tools that support interdisciplinary developer teams. Afterwards, a case study is presented, which was conducted to examine the framework's functionality as well as the usability of the tools. Finally, the scope of the framework is discussed. Currently, the XSS Framework is restricted to distributed applications where each learner is equipped with a personal device (e.g. a laptop). The last section therefore presents a concept for a framework extension which seamlessly integrates heterogeneous displays (e.g. tabletop displays).

## 3.1 Related Work

There are several approaches which aim at the development of scripted collaborative learning applications. On the basis of the CoSSICLE framework from Kobbe et al. (2007) the graphical modelling tool MoCoLaDe (Harrer and Malzahn, 2006) has been developed, which allows educators to designing collaboration scripts. These script designs can be exported as IMS-LD documents, which are machine-readable specifications of learning designs (cf. section 2.1.4). ReCourse (Griffiths et al., 2009) is another IMS-LD editor, however, it is not based on the CoSSICLE framework. Another tool that creates a computer readable script specifications in IMS-LD is Web Collage (Villasclaras-Fernández et al., 2009a). Instead of an implementation based on a script formalization (such as the CoSSICLE framework), specific script patterns (e.g. a jig saw pattern) and assessment capabilities are offered to the user. Furthermore, CeLS provides capabilities for designing and enacting collaboration scripts on the basis of IMS-LD (Ronen et al., 2006).

Wecker et al. (2010) propose a different approach. Instead of creating an abstract machine-readable description of a collaboration script, S-COL (Scripting for Collaborative Online Learning) provides ready-made instances of collaboration scripts for any kind of web-based environment. It is implemented as a browser plug-in and therefore can easily be used on different learning platforms. Another web-based approach is *ManyScripts*. It offers teachers a set of predefined scripts that can be adapted in terms of content, group size, timing of the phases and more. However, it does not support the authoring of new scripts (Dillenbourg et al., 2009). Yet another tool which is restricted to a set of pre-defined learning activities is LAMS (Dalziel, 2003). It is also neither conform to IMS-LD nor to the CoSSICLE framework. Stegmann et al. (2011) provide an overview of these approaches. Figure 3.1 is adopted from the paper. The approach proposed in



**Figure 3.1:** Overview of approaches to the development of scripted learning applications according to Stegmann et al. (2011). Our approach is highlighted in blue.

this thesis is highlighted in blue. As the figure shows quite clearly, it is the only approach which allows a continuous development process *and* follows the CoSSICLE framework. The latter is particularly important since it allows for an unlimited variety of supported scripts.

CeLS and similar approaches enable educators to design their collaboration scripts and enact them in IMS-LD compatible learning environments. The great benefit of this approach is the low barrier for educators. Collaboration scripts can be implemented without extensive programming skills. The drawback is that the control over the user interface implementation is lost. This can be a strong disadvantage in terms of human computer interaction (HCI). While the S-COL approach provides full control over the GUI, programmers are needed to implement the entire ‘logic’ of the scripted learning application as well as the graphical user interface. Another limitation of most existing approaches is that they are restricted to web-based learning environments. Applications for novel devices like smartphones, tabletop displays or tablet computers are not considered. In addition, IMS-LD based learning environments are usually full-featured platforms rather than simple-to-use ‘out of the box’ solutions that are needed in classroom settings. For instance, while user management is needed in university-wide learning environments, such features are unnecessary for teachers who realize a peer-review script in class. The solution, which we propose in the following, provides full control over the GUI and at the same time incorporates the capability to read and enact IMS-LD descriptions of collaboration scripts.

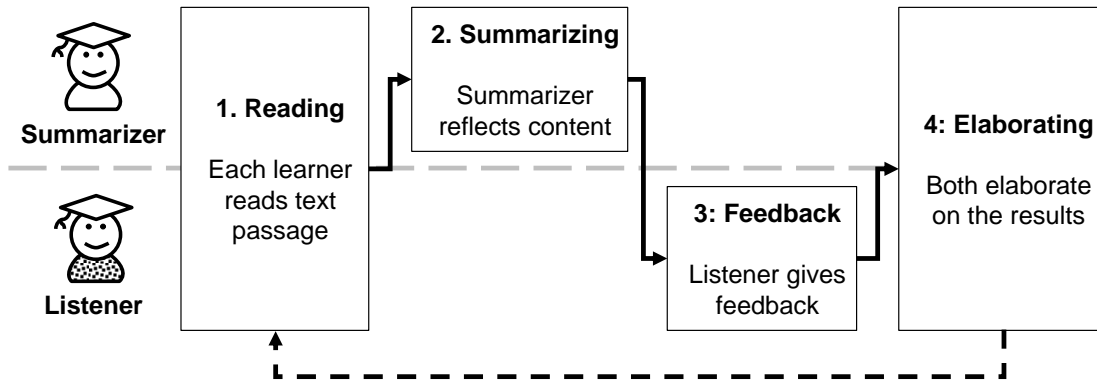
## 3.2 An Example Application for Scripted Text Comprehension Training

This section presents the development of a collaborative learning application for text comprehension. The underlying collaboration script was derived from a collaboration script called M.U.R.D.E.R. (Dansereau et al., 1979), which was originally developed for pen-and-paper sce-

narios. The M.U.R.D.E.R. application is a typical example of an application that can be developed with the XSS Framework.

### 3.2.1 M.U.R.D.E.R. Script

M.U.R.D.E.R. is a face-to-face script which aims at supporting reading comprehension (Dansereau et al., 1979). The abbreviation stands for Mood, Understanding, Recall, Detect, Elaborate and Review, which represent the different stages that the learners iterate through. It is designed for two learners who rotationally adopt the roles of a summarizer and a listener (see Figure 3.2). In the first phase the learners read a text passage on their own. Afterwards one person summarizes the text while the other one listens and tries to find errors or omissions. In phase three, the listener gives feedback and subsequently the results are elaborated on together. For the next paragraph the roles are switched.



**Figure 3.2:** Roles and Learning Activities in the M.U.R.D.E.R. script.

The M.U.R.D.E.R. script has proven to be an effective learning strategy (Dansereau et al., 1979). After an assessment in a learning skills course, the pre-test-post-test gains were significantly higher than in the control group.

### 3.2.2 M.U.R.D.E.R. Application

Based on the M.U.R.D.E.R. script a computer application was built. The application adopts the phases, roles and activities from the original pen-and-paper script. In addition, learners are instructed to create keywords that are intended as support the learning tasks throughout all phases. In phase one both learners are asked to identify the five most important keywords. Identifying the main ideas of a text is a fundamental comprehension skill which has been successfully used before, for instance by Dishner and Readence (1977). Therefore, we combined two successful scaffolds: the M.U.R.D.E.R. script and the identification of keywords. In the second phase the



keyword list serves as story thread for the summarizer. The listener can easily compare his own keyword list with the renarration and use deviations as basis for his feedback. In the elaboration phase both lists are compared and the two learners have to agree on a final revised list.

In particular for the last step computers are very useful as they visualize which keywords were marked by both learners and where differences occurred. In addition, the computer application guides learners through the sequence of tasks step by step.

### *Choice of display environment*

The application is designed to run on two connected co-located notebooks. In this case providing private workspaces was preferred over one shareable user interface because the learners are not supposed to see their partner's keywords. That way both learners are forced to individually think about the facts they consider most important. Moreover, the task of agreeing on a final keyword list can only trigger fruitful discussions if the two keyword lists are not identical.

### *Design of the graphical user interface*

Computer applications can have many advantages. However, if the usability of the application is poor, it becomes a barrier to learning rather than a support. Furthermore, when applications are designed in a fun and playful manner, they can distract learners from the learning task (cf. Hinrichs et al., 2008 or Sugimoto, 2009). To avoid such negative implications, a 'minimal design' was chosen. The idea is to carefully select few graphical user interface elements in order to guide the learners' activities. At a particular time only those widgets are supplied that are needed to complete the current task. This is contrary to many other learning applications, which provide various ways of interaction to put the learner in control of the learning process and/or use multimedia elements to motivate the learner.

Figure 3.3 shows a screenshot of the GUI in the first phase. It consists of few standard elements (text fields and buttons). Only two standard interaction techniques are needed to fulfill the task: (1) By marking words in the text keywords are automatically extracted and placed in the keyword list. (2) To change the order of the keywords or delete one of them, drag-and-drop is used. The screens in phases two and three look very similar. The only difference is that the text is hidden and whoever listens to the other person's narration can add notes in a text field. In addition the listener can checkmark keywords that were mentioned by the summarizer. In phase four all information is put together for the elaboration: The original text with markings as well as keywords lists and notes of summarizer and listener. As shown in Figure 3.4, the screenshot in phase four does not look 'minimal' anymore. However, it is still easily understandable because all elements and their handling are already known from the previous phases. In other words, the GUI is gradually expanded and the learner's attention is implicitly guided to the task at hand (cf. Figure 3.5).

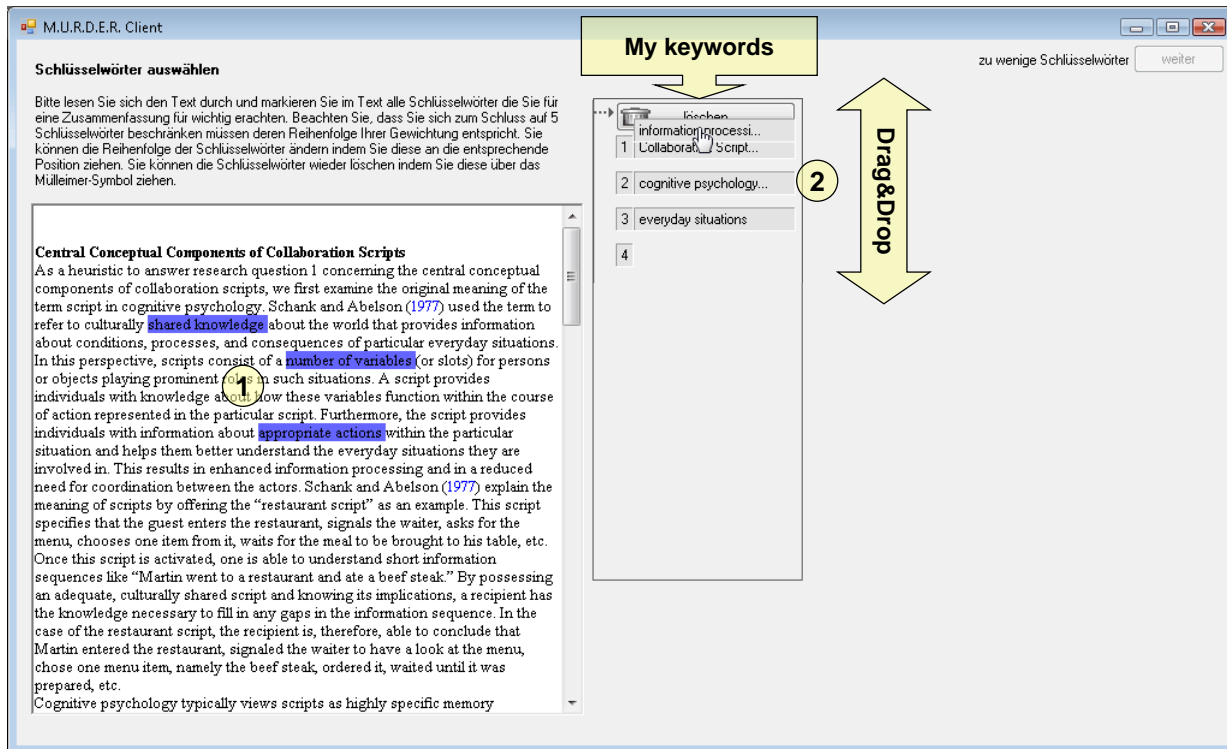


Figure 3.3: Screenshot of the M.U.R.D.E.R. application in Phase 1.

### 3.2.3 Usability Assessment

To assess the usefulness of the M.U.R.D.E.R. application two user studies have been conducted in regular seminars at the University of Munich. In both studies the application was compared to a pen-and-paper version in which the same instructions were given.

The first study ( $n=14$ ) examined variables connected to learning. It took place in two successive sessions of a seminar on motivation psychology. In the first session one half of the students was supported by the application while the other half got the pen-and-paper version of the script. In the following week the allocation was swapped. Both versions of the script contained exactly the same instructions. The post-questionnaire results show that the two groups can be regarded as equally motivated and showing similar effort ( $p>.20$ ; type-II error $<.15$ ).

The second study ( $n=23$ ) focused on the usability of the application. It took place in a media psychology course during a single session (see Figure 3.6). In this course the application was used by all participants. The participants' ratings regarding usability and functionality were very satisfying. Table 1 shows the investigated aspects, means and standard deviations (SD). In the two columns at the right the number of questions asked to evaluate the aspects (#items) and the reliability (Cronbach's  $\alpha$ ) are stated. Observations confirm that the sequence of action was overall as planned without the tutor having to intervene. This indicates that the application can

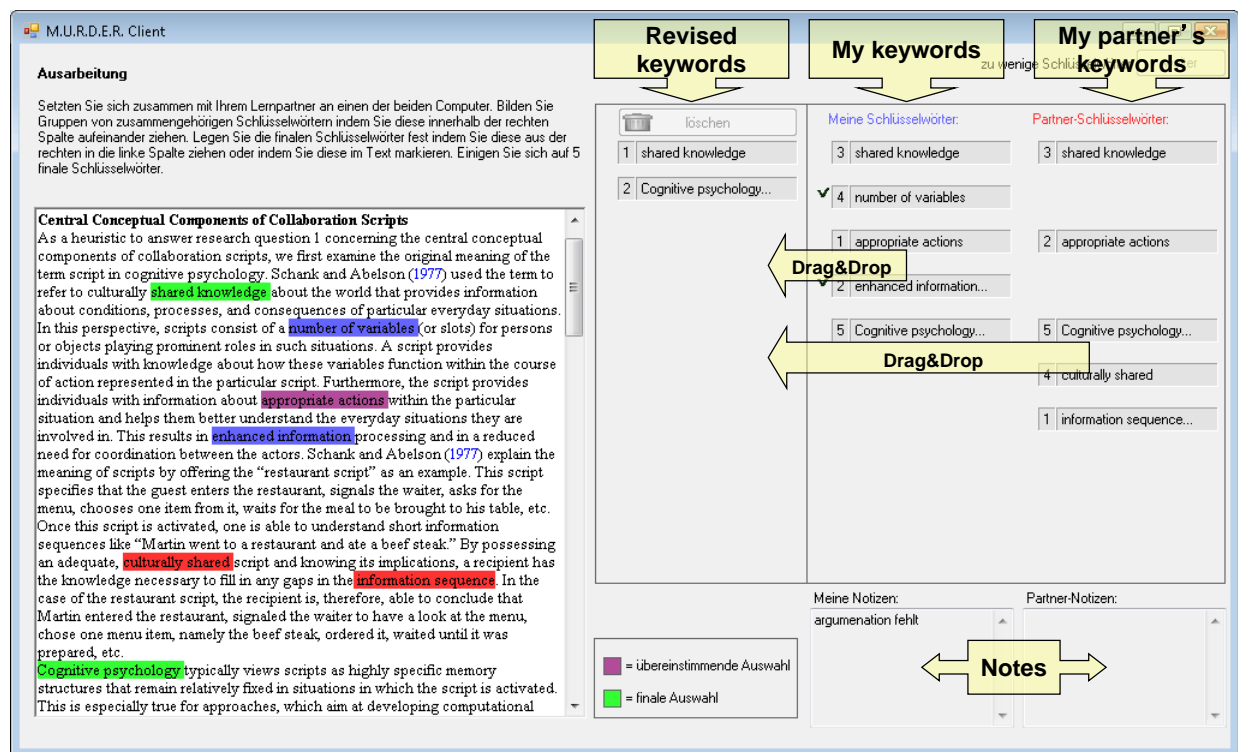


Figure 3.4: Screenshot of the M.U.R.D.E.R. application in Phase 4.

take over certain aspects of guidance and thus part of the tutor's job. Finally, about 75% of the participants would prefer the computer application to the pen-and-paper version.

The studies show that the M.U.R.D.E.R. application was designed in a way that it does not interfere with the learners' motivation and subjective effort. Furthermore, the application has certain features (e.g. an automatic group formation) that ease the use in classroom settings. Therefore, the computer application can be regarded superior to the pen-and-paper version because - while offering more features - the same effects are obtained.

The M.U.R.D.E.R. application is one example of a successful scripted learning application. In the following we will use this example application to illustrate the working principle of a framework, which supports the development process and maintenance of scripted learning applications.

### 3.3 eXtremely Simple Scripting (XSS) Framework

The XSS Framework automatically generates parts of scripted applications such as the M.U.R.D.E.R. application. In addition it offers a number of tools to support the development and maintenance of such applications. In the following it is explained which application parts can be automatically generated and why (sections 3.3.1 and 3.3.2). Afterwards, the development

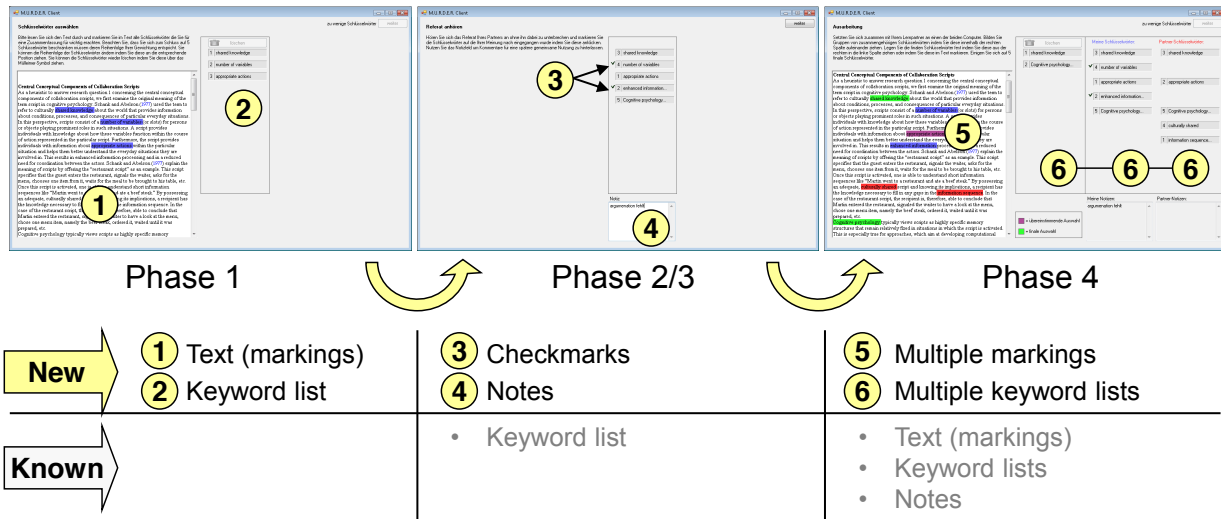


Figure 3.5: Step-by-step introduction of GUI elements.



Figure 3.6: Setup in the media psychology course.

process is presented starting with design of the collaboration script over the automatic generation of application parts and the finalization of the application to its usage in the classroom (section 3.3.3). Finally, the tools and application controls are presented, which support an interdisciplinary development process as well as tutors in administrating the application during runtime (section 3.3.4).

	Mean*	SD	# Items	Cronbach's $\alpha$	
Overall impression	3,55	0,70	8	0,87	
Usability	Simplicity	3,65	0,85	3	0,57
	Intuitiveness	4,23	0,79	3	0,49
	Comeliness	3,90	0,80	3	0,68
	Serviceability	4,10	0,77	3	0,46
Functions	Marking	3,73	0,83	3	0,63
	Sorting	4,17	0,85	3	0,75
	Deleting	4,45	0,73	3	0,73
	Grouping	3,43	0,77	3	0,57

\*scale:  
 5 very good  
 4 good  
 3 not sure  
 2 bad  
 1 very bad

Figure 3.7: Ratings regarding usability and functionality.

### 3.3.1 How Macro Scripts Define the Application Structure

According to Kobbe et al. (2007) collaboration scripts consist of specific components (participants, activities, roles, resources and groups) and mechanisms (task distribution, group formation and sequencing). This corresponds to the *macro* level of collaboration scripts (Villasclaras-Fernández et al., 2009b). In the following we will demonstrate that once these components and mechanisms of a collaboration script are defined, the overall structure of the learning application is determined as well.

The mapping between collaboration script and application structure is easiest explained using a practical example. Figure 3.8 shows the application structure as defined in the collaboration script that was used in the M.U.R.D.E.R. application. On the top there are the four consecutive phases, which are iterated through for each text paragraph. Below there are white boxes that summarize the learning activities per roles and phase. Each white box represents one screen of a GUI that offers the means to fulfill the corresponding tasks to one learner. For instance, the white box

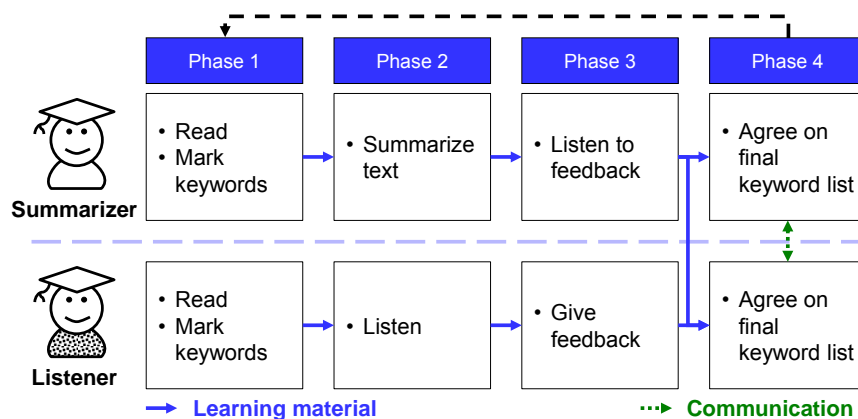
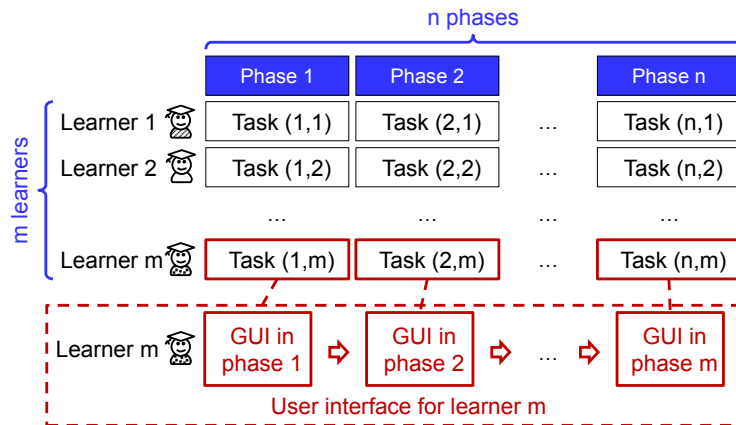


Figure 3.8: Application structure derived from the M.U.R.D.E.R. script.



**Figure 3.9:** Generalized structure of scripted collaborative learning application.

on the top left can be thought of as a screen containing a text (that the summarizer has to read) and some interaction technique to mark keywords. The box in the second column represents the screen that supports the learner in summarizing the text - in our case a list of keywords, which can be checked off.

The continuous blue arrows represent output of one learning phase which is used as input in a subsequent phase (i.e. the application's data flow). For instance, the keyword list, which is created by the summarizer in phase 1 (output), serves as a narration thread in phase 2 (input). In phase 4 the summarizer's keyword list is sent to the listener and vice versa. All of this structural information can be extracted from the collaboration script.

Figure 3.9 shows the mapping between collaboration script and application structure in general. The collaboration script defines the number of phases and learners. Each learner is provided with a GUI consisting of a series of screens. Each of the screens supports the learner in accomplishing the learning task of the current phase. The script also defines the distribution of resources (i.e. input and output for the learners in the different phases) as well as the synchronization between the learners (i.e. when learners have to wait until their partners are finished with the current phase). This corresponds to the client-server-communication of the distributed learning application.

Consequently, a greater part of scripted applications is predetermined by the underlying collaboration script. To generate these parts automatically and save developers unnecessary programming effort we implemented a functional framework called *eXtremely Simple Scripting (XSS) Framework*, which is presented in the following section.

### 3.3.2 Automatic Generation of the Application Structure

The XSS Framework uses a client/server-based approach. Each learner interacts with one client, which provides the graphical user interface. The server is the central component, which controls the entire application.

As we have demonstrated in section 3.3.1, the overall structure of scripted collaborative learning applications is defined by the underlying collaboration script. Due to this structural information, parts of scripted learning applications can be automatically generated: A fully functional server, the client-server-communication and parts of the clients (so-called ‘Dummy Clients’).

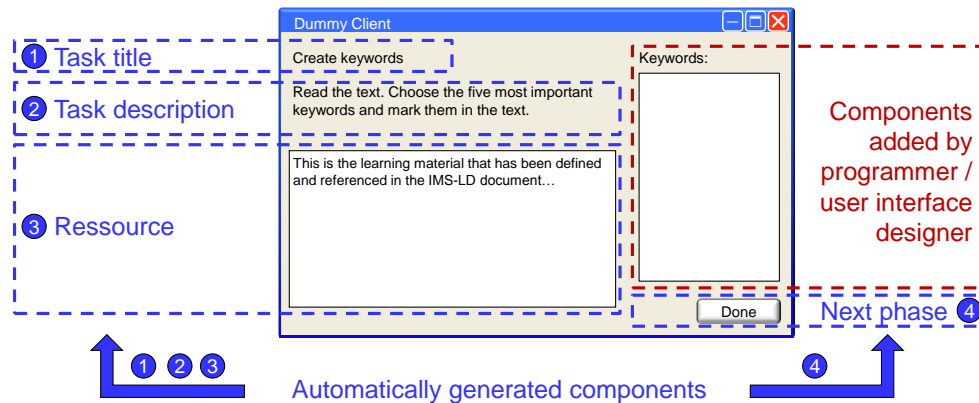
### *Fully Functional Server*

The server is heart and brain of the application. It is responsible for the *task distribution* and provides the resources to the clients. Resources are documents (e.g. rich text format files) that are accessed through relative file paths. This allows teachers to easily adapt the learning content at any time and without complicated installation procedures. In addition, the server controls the *sequencing* of learning phases. Therefore, the server is in charge of phase transitions and decides when learners have to wait for their partners before continuing with the next phase (synchronized phase transition) or when they can continue regardless of their partner’s progress (asynchronous phase transition). Furthermore, the *group formation* is done by the server. The decision which learners are put in which group can either be done automatically or manually entered by the tutor (see group administration tool in section 3.3.4). In the latter case, the groups can either be formed randomly or according to different criteria such as age, gender, nationality, knowledge or skills. For this purpose the server can store information of registered learners, which can either be provided by the teacher or directly entered by students through the client user interfaces.

After the collaboration script is imported the server configures itself accordingly. Then, the server is ready-to-use, which means there is no further development or adaptation required. The server can be started directly by double-clicking the CSCLServer.exe. However, it is recommended to use the *SimpleStarter*, which integrates tools for developers as well as tutors after the server is started (see section 3.3.4).

### *Dummy Clients*

The clients provide the graphical user interfaces to the learners. Each client is specifically designed for one learner: Its graphical user interface corresponds to the sequence of learning activities that the learner is asked to accomplish according to the collaboration script (see Figure 3.9). As the entire application logic is integrated in the server, the clients merely receive the learning instructions and resources, present them to the learner, provide the learner with the means to fulfill the task and send the learning outcome back to the server. The instructions and resources are defined in the collaboration script. Therefore, it is possible to automatically generate so-called ‘Dummy Clients’. Dummy Clients are click-through prototypes. Their graphical user interface shows a title and description of the current task, the learning material and a ‘Done’ button to move on to the next phase (see Figure 3.10). The client-server-communication, including exchange of material between clients as specified in the script, is already implemented. For instance, in a script for two learners who build a dyad, a common task could be reviewing the partner’s outcome of the previous phase. The functionality of exchanging learning materials (e.g. summaries and reviews) between the clients is already integrated in the Dummy Client. Therefore, by starting multiple clients the entire collaboration script can be run through.



**Figure 3.10:** DummyClient with automatically generated components.

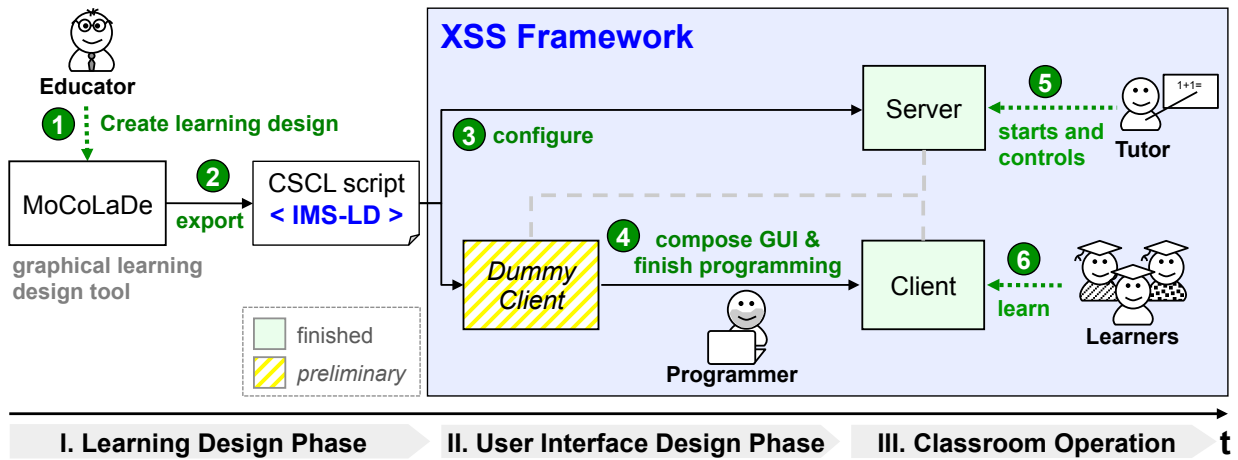
The remaining functionality is added manually by programmers. In the example of the M.U.R.D.E.R. application this would be the functionality of marking keywords in the text and organizing them (see Figure 3.10). While the user interface design process can be supported with GUI designers or user interface languages (e.g. XAML) it cannot be automatized. The reason is that IMS-LD documents exported by MoCoLADe only cover the macro level, which defines the structure of the learning application (e.g. the sequence of tasks, the role distribution and the group formations). The micro level that specifies *how* these tasks should be accomplished is not included. As the micro level provides the essence of the user interface definition, the user interface needs to be designed and programmed manually. This should ideally be done under consultation of educators who contribute their experience on micro scripting. Consequently, the framework saves programming effort but it still requires programming skills to create scripted learning applications with the XSS Framework. In the next section we will describe the resulting development process.

### 3.3.3 An Interdisciplinary Development Process

To successfully develop scripted collaborative learning applications experts from different domains need to collaborate: Firstly, *educators* create collaboration scripts to define how groups of learners should collaborate in order to maximize learning outcomes. Secondly, *programmers* implement the application. Thirdly, *user interface designers* should be integrated in the development process as the user interface can have a considerable impact on collaborative process (as will be pointed out later in chapter 4). In addition, *tutors or teachers*, who run the application in the classroom, need to be considered.

Therefore, next to the reduction of the programming effort a second major goal was to support interdisciplinary teams in designing scripted learning applications in a collaborative manner (similar as Pleuss et al. (2007) proposed for the interdisciplinary development of interactive applications). Furthermore, we aimed for a continuous process starting with the design of a collaboration





**Figure 3.11:** Framework Architecture.

script throughout the application development to administration in the classroom. To achieve this we propose the following development process, which is illustrated in Figure 3.11. The first step is to create a collaboration script. This is usually done by educators. For our purposes a machine-readable specification of the script is needed (e.g. IMS-LD). An exemplary IMS-LD document is presented in Appendix A. As it is difficult to directly create and edit such documents, a graphical editor can be used - in our case MoCoLADe. MoCoLADe allows non-programmers to graphically create collaboration scripts and to export them as IMS-LD document. This document is then imported by the XSS Framework and transformed into a fully functional server and Dummy Client. Therefore, the process follows a model-driven approach with IMS-LD as domain-specific language (cf. Zarraonandia et al., 2009).

As *Dummy Clients* are already capable of the full client-server-communication, it is possible to start the required number of Dummy Clients (which is defined in the collaboration script) and run through the application in a simulation mode. Such simulations support interdisciplinary teams in discussing how the finalized client user interface should look like. This is particularly important as educators should be involved in the user interface design in order to contribute the micro scripts. As the Dummy Clients are minimally functional but nonetheless interactive representations of the application these simulations are easily understandable for programmers and non-programmers alike. Therefore, simulations with Dummy Clients support interdisciplinary development processes of scripted collaborative learning applications. Once programmers have finalized the client, the application is ready for classroom operation. A tutor or administrator can start and control the server and provide the learners with clients.

The framework is implemented in C#. At its core there are two projects. *CACL* contains basic classes for scripted collaborative learning applications (Role, Sequence, Activity, etc.). A simplified UML diagram is attached in Appendix B, Figure B.3. The second important project is

*NDigester*<sup>3</sup>, an open-source tool for parsing XML documents. It deserializes the IMS-LD documents and creates objects from pre-defined CSCL classes (see code listing in Appendix C). At the beginning of the development process the functionality of the client objects corresponds to the Dummy Clients (see section 3.3.2). To finalize the client user interface programmers adapt the method *NextPhase*, which defines the content of the client's viewer (see code excerpt in Figure 3.12). A more comprehensive code listing of the client is attached in Appendix D. As the definition of the client user interface is separate from the XML parsing, it is possible to import IMS-LD documents after the client code has been changed. Therefore, modifications of the collaboration script do not require programmers to start from scratch (although the client code needs to be adapted accordingly). More detailed information on the implementation is presented in Appendix B.

```
// enter new phase and adapt the viewer
private void NextPhase(Activity activity) {
    // set title and task description
    viewer.taskTitle.Text = activity.name;
    viewer.taskDescription.Text = activity.description;

    // show input resource
    viewer.editor.Rtf = activity.attachedResource.rtfText;

    ...

    // enable ready button
    viewer.readyButton.Enabled = true;
}
```

**Figure 3.12:** Code listing: In the Dummy Client the Viewer only shows title, task description and resource of the current phase. This part is completed by programmers during the finalization of the client user interface functionality.

Client and server applications are executable (.exe) files. No prior installation or configuration is required on the target devices except for IP address and port numbers, which can be entered on the welcome screen. In addition, the XSS Framework offers several tools to support the application development as well as maintenance during runtime. These tools are presented in the following section.

### 3.3.4 Tools and Application Controls

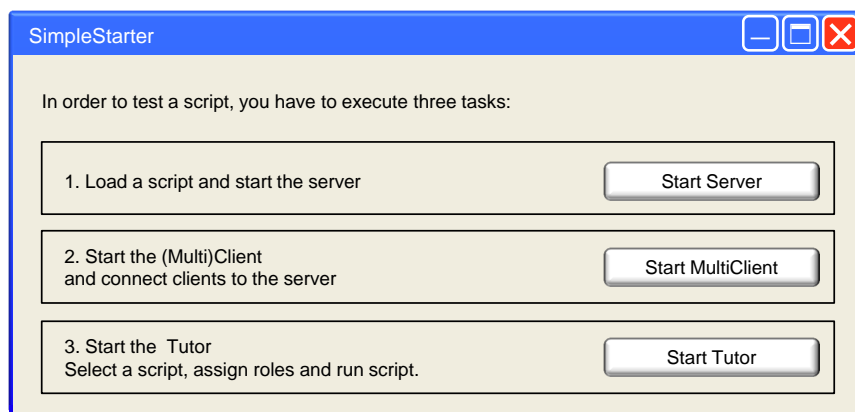
Across all development phases as well as during runtime the involved parties need to be provided with adequate user interfaces. The XSS Framework therefore provides a set of auxiliary tools. The *Simple Starter* guides users of the framework through the necessary steps. The first step is starting the server. Second, the *Multi Client* provides developers with an environment to run

<sup>3</sup> <http://sourceforge.net/projects/ndigester/>

simulations. Finally, the *Group Administration Tool* allows tutors to administrate the learning application in the classroom.

### *Simple Starter*

The entry point to using the XSS Framework is the *Simple Starter* (Figure 3.13). It is a user interface wizard that guides the user through a sequence of steps. Depending on the development stage the Simple Starter is either used by the programmer and/or educator responsible for designing the learning application or the tutor once the application is fully developed.

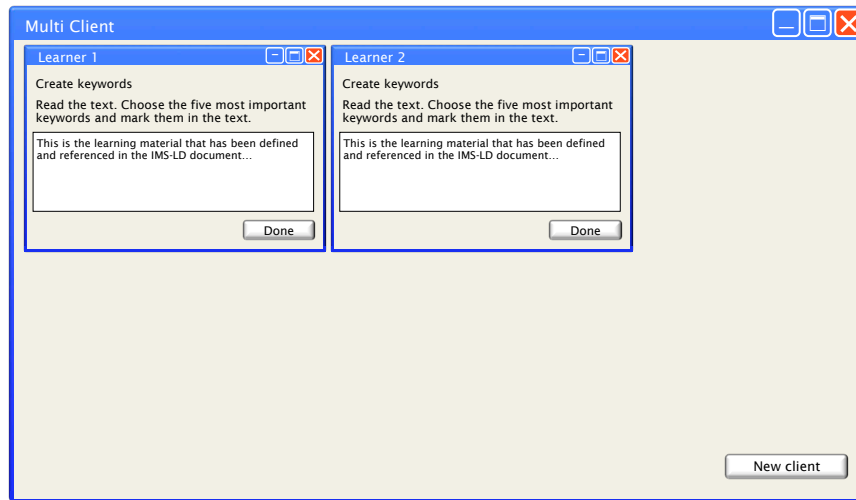


**Figure 3.13:** SimpleStarter.

The first step is starting the server. The user can select a collaboration script (i.e. an IMS-LD document) to be imported. The script is interpreted and the information is then used to automatically configure the executable server. Once the server is running, it is ready to accept requests from clients or the Group Administration Tool.

### *Multi Client*

The second step is starting the clients. In the classroom clients run on different devices (as all learners have their own devices). During the development Dummy Clients can be used to run simulations. For simulations it is easier to start multiple clients on the same machine instead of setting up a network. Using the 'Done' button application developers can click all clients through the phases. However, each client has its own window and disappears into the background when another client is activated. With an increasing number of clients, running simulations on one machine becomes more and more difficult and the main task can easily shift to searching for a specific client among several windows that all look very similar. To avoid this the *Multi Client* was developed, which clearly arranges multiple clients in one window. Running simulations allows creators of collaboration scripts to verify that their model was correctly interpreted. In addition, the simulation mode supports interdisciplinary teams of educators, designers and programmers to discuss the functionality and design of the user interface in each phase.



**Figure 3.14:** The *Multi Client* arranges several clients in one window in simulations.

### *Group Administration Tool*

The third and final step is starting the *Group Administration Tool* which supports the group building process. It is split into two sections: the (unassigned) user section at the top and the group section at the bottom (see Figure 3.15). In the user section the tutor can import user data from a csv-file (using the button 'Load User Data') or enter data manually (a click on the 'i' button in the 'More...' column opens the corresponding form). The types of information (e.g. sex, age, nationality, etc.) are those specified in the collaboration script (see Figure 3.16).

As an alternative to manual group formations done by the tutor, groups can also be automatically formed by the framework. This can either be done randomly or according to criteria such as age, gender or nationality. For instance, it can be useful to build groups with group members of the same (homogeneous) age but mixed (heterogeneous) gender.

## 3.4 Case Study

A user study was conducted to assess the usability and functionality of the XSS Framework. Furthermore, the study should show whether the framework is flexible enough to support different types of collaboration scripts. During the development of the framework regular tests were conducted using the M.U.R.D.E.R. script. However, as a proof of concept we needed to see if the framework works with other collaboration scripts as well.

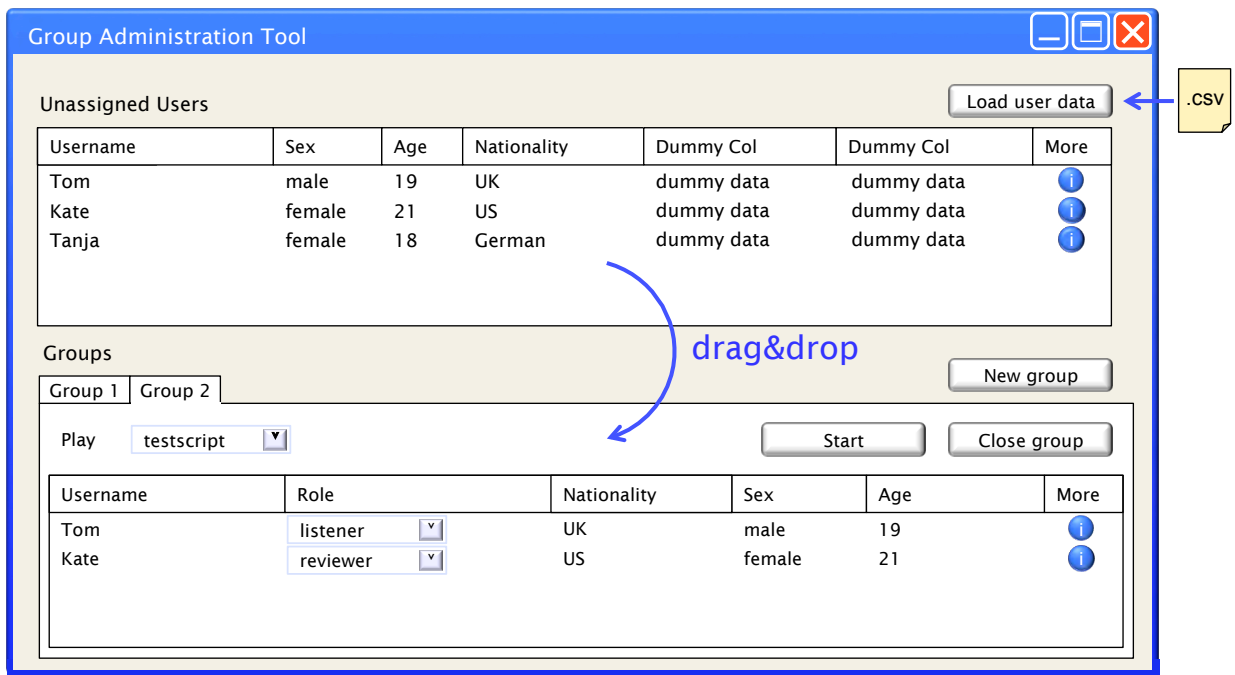


Figure 3.15: Group Administration Tool for tutors.

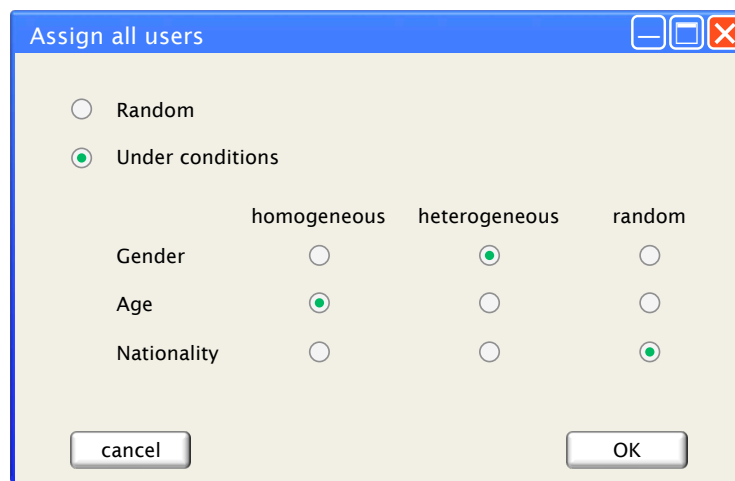


Figure 3.16: Automatic assignment of users to groups.

### 3.4.1 Method

#### *Participants and Designs*

We invited four doctoral students of the learning sciences who worked with different collaboration scripts. That way the participants could be asked to use their own collaboration scripts. Three participants were female. Their age ranged between 26 and 34.

#### *Learning Task and Procedure*

The subjects modelled their own collaboration scripts using MoCoLADe and exported an IMS-LD document. Using the XSS Framework the IMS-LD was then imported to create the according learning application (using the Simple Starter). Furthermore, a simulation was run with the Multi Client Tool and the Group Administration Tool was used for the group formation. However, the Dummy Clients were not finalized. Thus, the complete development process (see Figure 3.11) was run through except for steps four and six.

As the subjects were not familiar with MoCoLADe the study was divided in two parts. In the first session the experimenter gave a MoCoLaDe training. In the beginning of the training session there was a lecture on MoCoLaDe's functionality and components. Afterwards the participants were given laptops and modelled an exemplary collaboration script following a step-by-step tutorial given by the instructor. Finally, the participants modelled their own collaboration scripts and exported it as IMS-LD document.

The second part of the study was scheduled for the following day. At first there was a short introduction to the framework. Then the development process was demonstrated using the exemplary script from the day before. Afterwards each participant imported the IMS-LD file that they had created in the last session, ran the application and tested the Multi Client as well as the Group Administration Tool. In the end, there was a questionnaire followed by an open discussion about missing features as well as other suggestions for improvement.

#### *Measures*

At the end of the second day, the participants were handed out questionnaires to examine the usability of the XSS Framework. The questions were split into usability assessment of the server and of the Group Administration Tool (cf. Figure 3.17). In addition, the questionnaire contained open questions regarding features that should be added to the XSS Framework. Their ideas were then discussed in more detail in interviews which were conducted after the questionnaires had been handed in.

## 3.4.2 Results

### *Functionality*

The doctoral students modelled a peer-questioning (King, 1997), two peer-review (Weinberger et al., 2005b) and an argumentative collaboration (Stegmann et al., 2007b) script. All of these learning scripts were successfully modelled in MoCoLaDe, imported and run as simulation using the XSS framework. However, IMS-LD currently does not support iterative cycles of activity sequences or conditional behavior (if ... then do activityX else do activityY). Both occurred in the user study. While this shortcoming is not caused by the XSS Framework, it limits the framework's functionality. To solve the problem the participants' collaboration scripts were simplified. Multiple cycles were reduced to one cycle and instead of conditional behavior a default activity was defined.

The participants were overall satisfied with the functionality. However, they asked for additional features during classroom operation. Similar to the Multi Client, which provides an overview during simulations, an Observation Tool could provide an overview of all running client windows and update changes in real time. The Observation Tool should also visualize which clients are in the same group. Furthermore, in the final questionnaire three participants suggested adding the functionality to export data. In the open discussion this suggestion was further discussed. As the individual steps as well as the results of the learning session often need to be evaluated, it should be possible to export a protocol and the outcome of the learning session.

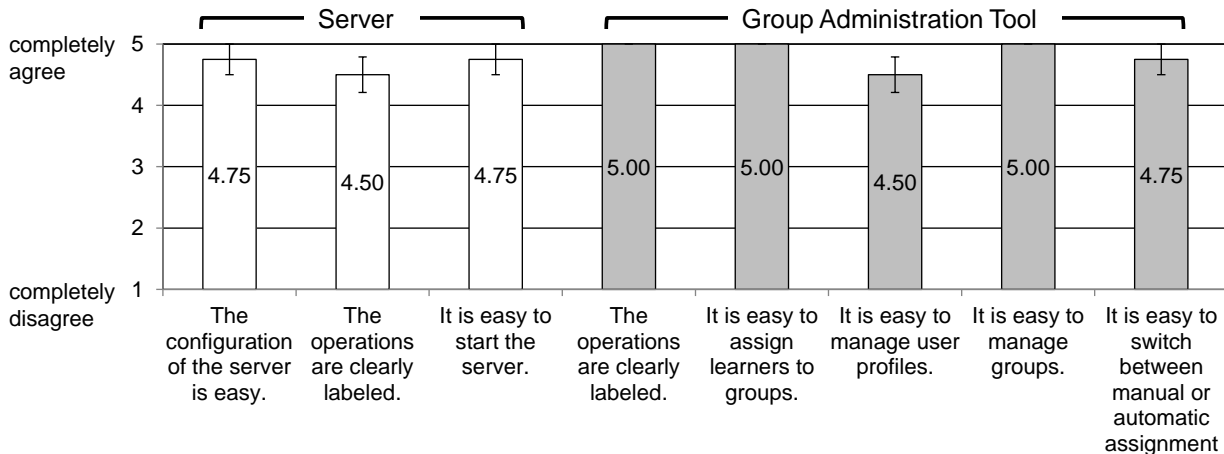
### *Usability Assessment*

To assess the usability of the user interfaces the final questionnaire contained several questions regarding the server and Group Administration Tool. The questions (for example 'The operations are clearly labelled') had to be answered on a Likert scale from 1 (completely disagree) to 5 (completely agree). All questions were answered very positively with averages of 4.5 or higher (see Figure 3.17).

In the discussion one participant proposed to combine the three windows (Simple Starter, Group Administration Tool and Multi Client) into one single window. Whether this is still feasible once additional features (e.g. the Observation Tool) are implemented, and how the windows could be merged to maintain a clear arrangement needs to be considered carefully.

## 3.4.3 Summary

Our case study provided evidence that the XSS Framework in combination with MoCoLaDe could be a helpful tool for educators. Doctoral students in the field of educational sciences were able to author their collaboration scripts with MoCoLaDe and test them using the framework's simulation tool. The XSS Framework not only provides tools for the development of collaboration scripts but also serves as runtime environment for collaboration scripts. Our results also



**Figure 3.17:** Usability assessment of server and Group Administration Tool.

showed some limitations of the current approach. So far IMS-LD does not allow cycles of activities or conditional sequences of activities. As cycles and conditions are fairly common in collaboration scripts the framework functionality should be further extended beyond the capabilities of IMS-LD. In addition, the participants wished for additional functionalities. During the learning phase they would like to monitor the activities of the learners. Finally, it should also be possible to archive process data and outcomes.

## 3.5 Towards an Integration of Shareable User Interfaces

Like all approaches in the related work, the XSS Framework is currently limited to macro scripts as well as to distributed applications where each learner is provided with a separate device. Consequently, the XSS Framework does not support shareable user interfaces, which can be valuable tools for collaboration. To overcome this limitation an extended XSS Framework is envisioned. The vision comprehends general ideas rather than an elaborated concept.

### 3.5.1 Scope of the XSS Framework

The XSS Framework analyzes collaboration scripts on a *macro level* and generates parts of *distributed* applications. Therefore, it currently neither supports *micro* scripts nor the development of applications running on *shareable* user interfaces.

The limitation to macro scripts was already addressed at the end of section 3.3.2. IMS-LD documents exported by MoCoLaDe define scripts on a macro level (task sequence, role distribution,



group formations and the like) but there is no specification of the *micro* level. Therefore, the automation of the XSS Framework is restricted to the general structure of scripted collaborative learning applications (as illustrated in Figure 3.9). The framework only processes predefined lists of phases in each of which learners are asked to engage in predefined activities. Scaffolds on a micro level, which specify *how* each task should be accomplished (cf. section 2.1.3) and therefore provide the fundamental requirements for the user interface design, are not processed automatically. In Figure 3.9 this would correspond to the design of the content within the white boxes. Therefore, the user interface needs to be (1) designed by educators and user interface designers in a collaborative manner and (2) manually programmed. This limitation can not be overcome without a metalanguage for micro-scripts (cf. Hernández-Leo et al., 2006).

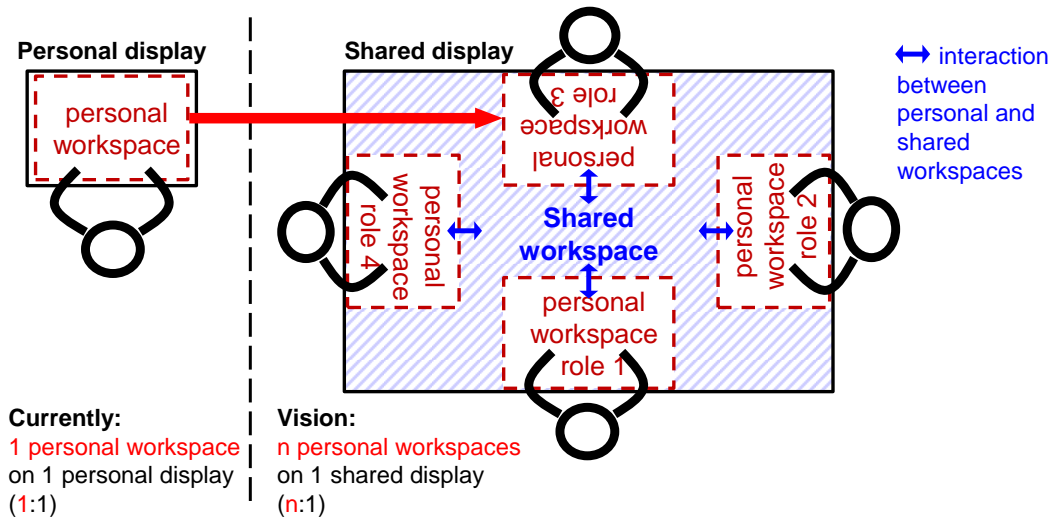
A second major limitation concerns the target devices. The components which are automatically generated by the XSS Framework consist of one client per participant, each of which runs on a personal device. Therefore, the framework supports the development of distributed applications such as the M.U.R.D.E.R. application where each learner has an own laptop. The personal device guides the learner through the phases in a wizard-like manner, which is a very strict form of guidance. However, in many situations shareable user interfaces such as interactive wall or tabletop displays can be beneficial - either in addition to or instead of the personal user interfaces. Several benefits of shareable user interfaces have been presented in section 2.3.2. Therefore, the following section presents a conceptual solution which allows to apply the framework's functionality to different types of displays.

### 3.5.2 Vision of an Extension for Shareable User Interfaces

This section presents the vision of a framework, which supports the development of scripted learning applications on heterogeneous displays. In other words, the envisioned framework not only supports personal devices but also shareable devices such as interactive wall or tabletop displays. At first, it is explained how the concept of workspaces can be used for that purpose. Second, some issues are addressed, which the use of different display types as well as the concept of workspaces entail. In the end, there is an outlook on possible framework solutions which could resolve some of these issues.

#### *The Concept of Workspaces*

The framework's functionality could be extended to shareable user interfaces using the concept of *workspaces*. Instead of providing each learner with a series of screens that are shown on a *personal display*, the series of screens could be mapped to a *personal workspace* that is part of a *shared display*. This is illustrated in Figure 3.18. The right part shows how multiple workspaces could be arranged on a shared interactive tabletop display. The personal workspaces (dashed rectangles) are located on the edges in direct proximity of their owners. In its simplest form the content of the personal workspace could be identical to the content that would be shown on a personal device (similar to Figure 3.10). The area in the center of the display can be used as *shared workspace*. This segmentation of territories occurs on any tabletop interaction regardless



**Figure 3.18:** From personal workspaces on personal displays to multiple personal workspaces on shared displays.

of the existence of visual demarcations as proposed here (Scott et al., 2004). When no lines of demarcations are shown the personal workspace is defined solely by the proximity to a user and naturally merges with the shared workspace. However, using marked-off workspaces allows to define the content of a workspace (e.g. using a markup language) and flexibly allocate it to different kinds of displays. For instance, a learner's graphical user interface (cf. Figure 3.10) can either be allocated to a laptop or to a marked-off area on a tabletop display. This choice of the display environment should be made by the user interface designer once the collaboration script is defined. It is the first step of the client user interface finalization.

### Issues

The above mentioned concept entails several issues. Some of these issues are caused by the heterogeneity of displays. When workspaces are designed for different types of devices, the display's characteristics need to be taken into account. First, there are basic aspects such as size and resolution of a display. Second, single- and multi-user devices need to be distinguished. While the display's orientation does not play a role for single-user displays, it makes a significant difference whether shareable displays are vertically oriented (wall display) or horizontally oriented (tabletop display). In the latter case, the users might not have the same perspective (e.g. Figure 3.10). A third factor is the input technology. For instance, whether input is done with direct touch or with indirect pointing devices affects how users interact with the application. For instance, on interactive tabletop displays not all parts of the display are reachable by all users. This problem does not occur with mice or other pointing devices. Finally, there are different techniques for text input, which also affect the layout of workspaces. When no hardware keyboard is available (e.g. on tabletop or tablet computers) text needs to be entered using on-screen keyboards or handwriting recognition, both of which can occupy large areas of a workspace.

In addition to the heterogeneity of displays, placing multiple workspaces on a shareable user interface allows for other interaction techniques than wizard-style applications. While in the M.U.R.D.E.R. application the exchange of keywords is automatically initiated by the server between phases, tabletop displays encourage users to move content between workspaces using simple gestures like drag-and-drop. Thanks to a number of different display-reaching techniques (Nacenta et al., 2005) such interactions between workspaces are also possible when the workspaces are located on different devices. For instance, *Pick-and-Drop* (Rekimoto, 1997) allows user to pick up digital objects with a pen, move the pen to another display and drop the object there. There are also techniques that work with pointing devices (e.g. *Hyperdragging* by Rekimoto and Saitoh, 1999) and thus allow to send objects to displays that are not within arm's reach. A framework for heterogeneous devices needs to consider such interactions between workspaces. Defining the interactions between workspaces is just as important as defining the workspaces' GUI components and the interactions within the workspace.

### *Solutions*

The basic version of the envisioned framework could leave the consideration of the above-mentioned issues to the programmer. The framework could simply support the definition of workspace content and interactions and provide the means to allocate them to heterogeneous displays. In this case, the programmer would implement the user interface for a specific target display and the framework could still handle the supported mechanisms. For instance, the framework could distribute tasks both across several devices or across several workspaces within the same device.

The more advanced the framework, the more of the above-mentioned aspects could be handled by the framework instead of the programmer. At the end of this theoretical scale is a framework, which allows the user interface designer to define workspace content and interactions in an abstract manner. The workspaces would be automatically configured and would also automatically adapt to the display it is attached to. In this hypothetical scenario there would be no need for a programmer. User interface designers could use graphical tools to define workspace content and interactions, similar as educators use graphical tools to define the collaborations scripts.

One step in this direction could be a unified gesture language as proposed by Echtler et al. (2010). In addition, there is research on seamless interaction between heterogeneous devices in the area of Ubicomp (e.g. by Greenberg et al., 2011 and Gellersen et al., 2009). While these approaches provide partial solutions, there are still many open questions. For instance, how do different kinds of workspaces and display environments affect collaborative learning? What are suitable guidance mechanisms for different workspace and display configurations? These questions need to be addressed in order to define the requirements of an extended XSS Framework more comprehensively.

## 3.6 Summary and Discussion

The XSS Framework supports the development of scripted collaborative learning applications. Collaboration scripts are designed by educators. Based on the information that is specified in the script, the framework generates part of the functionality automatically: a fully functional server and Dummy Clients. The clients need to be finalized before they can be used in the classroom. The framework provides tools which support interdisciplinary teams of educators, programmers and user interfaces during this process. In addition, there is the Group Administration Tool which allows tutors to administer the application during runtime. Therefore, the framework allows for a continuous process starting with the development of collaboration scripts to the development of the scripted application to the administration in the classroom.

A case study was conducted to examine whether the framework can process various kinds of collaboration scripts. To do so, we invited doctoral students from the field of learning sciences. The students modelled their own collaboration script and tested the functionality of the XSS Framework. All four scripts were modelled successfully. The usability of the framework tools was rated very positively. However, the study also showed some limitations of the current approach. So far IMS-LD does not allow cycles of activities or conditional sequences of activities. As cycles and conditions are fairly common in collaboration scripts, they should be integrated in IMS-LD and a future version of the framework. Furthermore, the participants suggested to add more functionalities to the runtime-environment. As a tutor, they would like to monitor the activities of the learners and archive process data and outcomes for further use.

Next to the room for improvement that was revealed in the case study, there is another direction in which the framework could be extended. Currently, the framework supports the development of distributed applications. Each learner is provided with a client on a personal device, for example a laptop. Over the last decades new forms of computers such as interactive surfaces emerged. Research showed that interactive tabletop displays, for example, can have positive effects on collaboration. Therefore, an extension for the XSS Framework was proposed, which supports the development of learning applications on heterogeneous displays including shareable user interfaces. However, there are several open questions with regard to the design of such mixed-display applications, in particular the two important questions, which were already raised in chapter 1: (RQ I) Which types of displays or workspaces are suitable in which scripted collaborative learning scenarios and (RQ II) How should user interfaces be designed to guide learners through collaboration scripts? The question of guidance is particularly challenging for heterogeneous displays because the wizard-style guidance, which is often used when each learner is provided with a personal device, may not be applicable to mixed-display applications. We argue that an extended framework should support developers with regard to these design decisions. As previous research has not yet sufficiently answered these questions they will be investigated in the following chapters. The design of the framework will not be further explored in this thesis since it depends heavily on the clarification of these elementary questions.

Chapter 4 through 7 each address a key question that is related to RQ I and RQ II. An overview is shown in Figure 3.19. While chapter 4 focuses on RQ I, the focus gradually shifts to RQ II in the

subsequent chapters. Chapter 4 explores RQ I by comparing benefits of single shared displays (in this case a tabletop display) and multi-display environments on collaborative learning processes. This is done in several studies in which different prototypes are used (e.g. ArgueTable, ArgueWall, SACTable, ...). These prototypes have already been shortly introduced in the thesis roadmap in the introduction (see Figure 1.2). Chapter 5 investigates how multi-display environments with private and shared workspaces can be used to implicitly guide activities of collaborating learners. This idea, which emerged in chapter 4, addresses both RQ I (the choice of display environment) and RQ II (guidance). The following chapters then explore explicit guidance mechanisms. Chapter 6 presents Guiding Widgets, which are GUI components that are integrated in the learning application. Chapter 7 investigates dedicated mirroring and guidance systems, which provide feedback on separate displays.

Figure 3.19 also illustrates the three levels of research questions that exist in this thesis. On the top level there are the two global research questions RQ I and RQ II. They are distinguishable by Roman numerals. Second, each chapter addresses a key question, which can be found in a textbox at the beginning of all following chapters. These chapter questions are not numbered. Third, in each study there are more specific research questions numbered in Arabic numerals (RQ1, RQ2, ...). The study research questions are renumbered, which means the numbers are only valid within the study they are reported in.

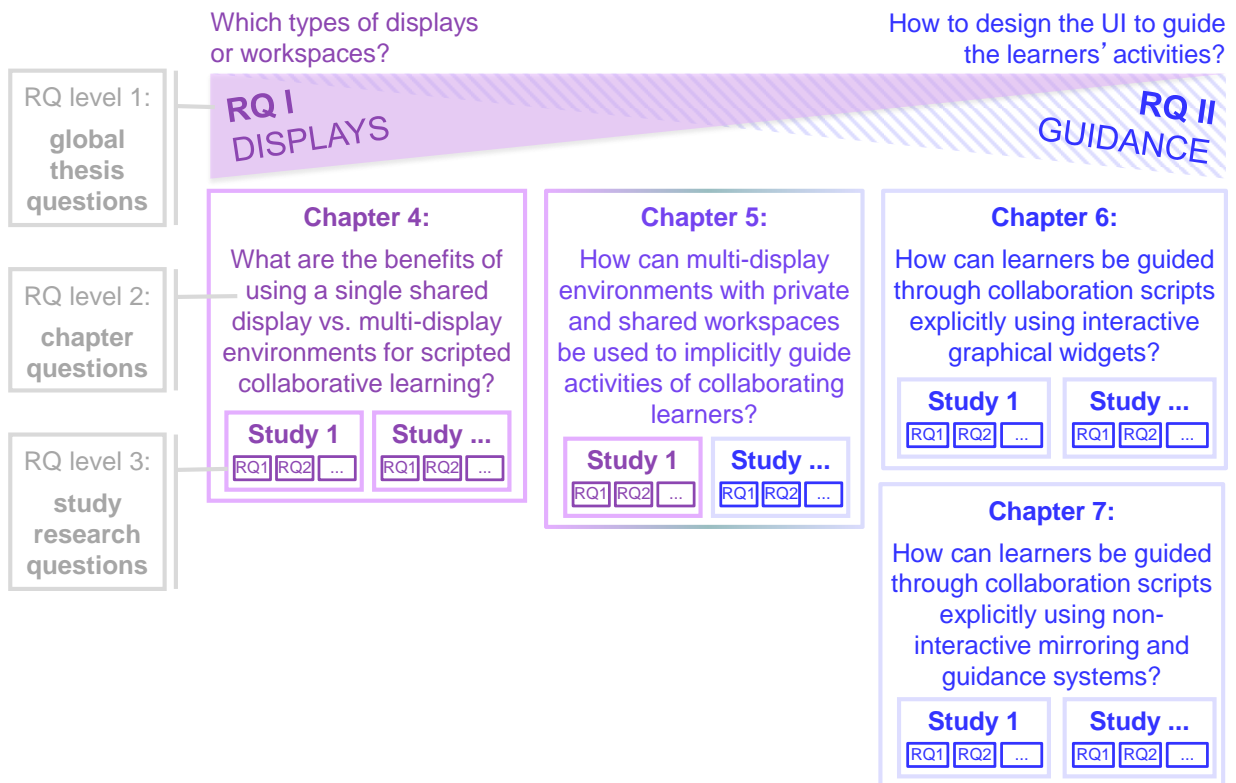


Figure 3.19: Key questions in the following chapters.



# III

EFFECTS OF DISPLAY  
CONFIGURATIONS





# Chapter 4

---

## Choosing a Display (Environment)

Traditional learning applications - collaborative or not - are designed for personal devices such as desktop computers. Over the past two decades new technologies such as interactive surfaces emerged (Wellner, 1993), which have shown to have advantages in particular for small groups (Müller-Tomfelde, 2010). Since then the diversity of display types that can be used for collaborative learning applications has increased. Consequently, when designing new collaborative learning applications, one of the first questions that needs to be answered is which display type(s) to use. Previous studies showed various assets and drawbacks of different types of displays. For instance, shared displays provide an awareness for the learning partner's activities whereas private displays allow group members to work undisturbed and without the feeling of being observed. These and other characteristics have been addressed in section 2.3.2 in more detail. However, most of the studies were conducted in the field of computer-supported collaborative work and therefore did not investigate learning-specific aspects. In addition, each display (environment) has both advantages and disadvantages, which are difficult to compare. The following chapter therefore investigates the benefits of using a single shared display vs. multi-display environments with shared and personal workspaces.

**Chapter Question:**

What are the benefits of using a single shared display vs. multi-display environments for scripted collaborative learning?

The first part of this chapter<sup>1</sup> presents a set of user studies on a tabletop application called 'ArgueTable', which was designed to facilitate argumentative knowledge construction. Argumentation was chosen because it is an important collaborative learning task (Andriessen and Coirier,

---

<sup>1</sup> Parts of this section have been published in (Streng et al., 2011b).

1999) and at the same time common in daily life. The iterative design process of the ArgueTable (section 4.1) led to an alternative display environment ('ArgueWall'), which uses laptops together with a shared wall display. The two display environments ArgueTable and ArgueWall were compared in a user study, which is presented in section 4.2. Results showed that participants created more argument sequences in the distributed setup (ArgueWall). To narrow down the cause of this effect several follow-up studies were conducted (section 4.3).

Finally, another study is presented, which investigated how important the specific benefits and drawbacks are to the learners (section 4.4). For instance, do learners prefer one single display across individual and collaborative phases or do they prefer to switch between displays, each of which is most suitable for the respective task at hand? The study indicates that learners are willing to switch displays and even physically relocate themselves if this results in more suitable display types for the respective tasks. An increased awareness of shareable user interfaces was rated significantly less important in our scenario.

## 4.1 Iterative Design of the ArgueTable

According to argumentative knowledge construction the quality of argumentation is a very important factor for successful learning. The following section presents a set of case studies on a learning application that aims at supporting argumentative knowledge construction<sup>2</sup>. The application provides scaffolds for the construction of single arguments and argument sequencing. During an iterative design process different user interface designs were evaluated regarding their effects on the knowledge construction process. Starting from a tabletop application ('ArgueTable') the design process led to an alternative display environment ('ArgueWall'), which uses laptops together with a shared wall display.

At first, the collaborative learning scenario of argumentative knowledge construction is introduced (section 4.1.1). Second, related work on computer-support for argumentation is presented (section 4.1.1). Both provide the foundation for the design of our application. Afterwards, the iterative design process of the ArgueTable is discussed (section 4.1.2). In the beginning of the iterative design process a paper prototype study was conducted (section 4.1.3), followed by case studies with functional prototypes (section 4.1.4).

### 4.1.1 Argumentative Knowledge Construction

#### *Educational Foundation*

The quality of argumentation has been broadly regarded as an important aspect of collaborative knowledge construction (Andriessen et al., 2003). Koschmann (2003) even equalizes learning

---

<sup>2</sup> The practical part of this work was conducted by Max Halbinger (2009) and Sonja Böhm (2010) under supervision of the thesis author.

and high quality argumentation during collaboration. High quality argumentation includes at least two central features: First, learners have to provide sound reasons for their claims (Baker, 2003). This (self-)explanation is considered to be a powerful learning activity that fosters learning (Renkl et al., 1998). Furthermore, Stegmann et al. (2007a) provided empirical evidence that the quality of single arguments is positively related with the depth of cognitive processing. Second, the construction of argument sequences has been regarded to provoke socio-cognitive conflicts and their positive resolutions (Kollar et al., 2007; cf. Doise and Mugny, 1984). Leitão (2000) proposes a sequence of argument, counter argument and integration for optimal learning.

While high quality of argumentation has shown to be beneficial for learning, it is rarely found in unsupported collaborations (Stegmann et al., 2007b). Learners usually do not provide grounds and qualifications for their claims. In addition, learners usually do not raise socio-cognitive conflicts through counter argumentations. Therefore, instructional means are required to enhance collaborative learning. Following this idea, computer-supported collaboration scripts have been implemented, which successfully facilitated collaborative knowledge construction by increasing the quality of argumentation during discussions (Weinberger et al., 2007c). Scripts can effectively structure different aspects of learners' interactions such as the content quality or the formal structure of argumentation. That way, scripted computer-supported collaborative learning induces specific patterns of learner interactions, which improves learning outcomes (Weinberger et al., 2010).

### *Computer-Support for Argumentation*

Scheuer et al. (2010) have done an extensive literature review on systems that support argumentation. A result of their review is a classification of different types of argument representations that have been used in such systems: linear, threaded, graph-based, container and matrix. Empirical studies show that the type of argument representation can affect collaboration. Suthers et al. (2008) conducted a study in which they compared the use of knowledge maps and threaded discussion. When using knowledge maps learners began to hypothesize earlier in the discussion, elaborated more and were more likely to converge on the same conclusion than when using threaded discussions. Furthermore, they scored higher in the post-test questions, which required an integration of information that was distributed across the group members. This suggests that knowledge maps facilitate collaboration better than threaded discussions. Other benefits of using knowledge cartography such as mind maps, concept maps or argument and evidence maps for learning are addressed in Okada et al. (2008).

In addition to different tools that support argumentation, the assigned task plays a major role. According to the 'Script Theory of Guidance' (Fischer et al., in press) learners use the same tool in different ways according to the task at hand. A study by Pfister and Oehl (2009) provides evidence for this assumption. Learners in a chat environment were either told that they would receive rewards based on their individual performance (individual goal focus) or based on their group's performance (group goal focus). The group goal focus led to better argumentation compared to the individual goal focus. This suggests that the quality of argumentation not only depends on the tool but also on the task. A similar finding was made by Lund et al. (2007) who

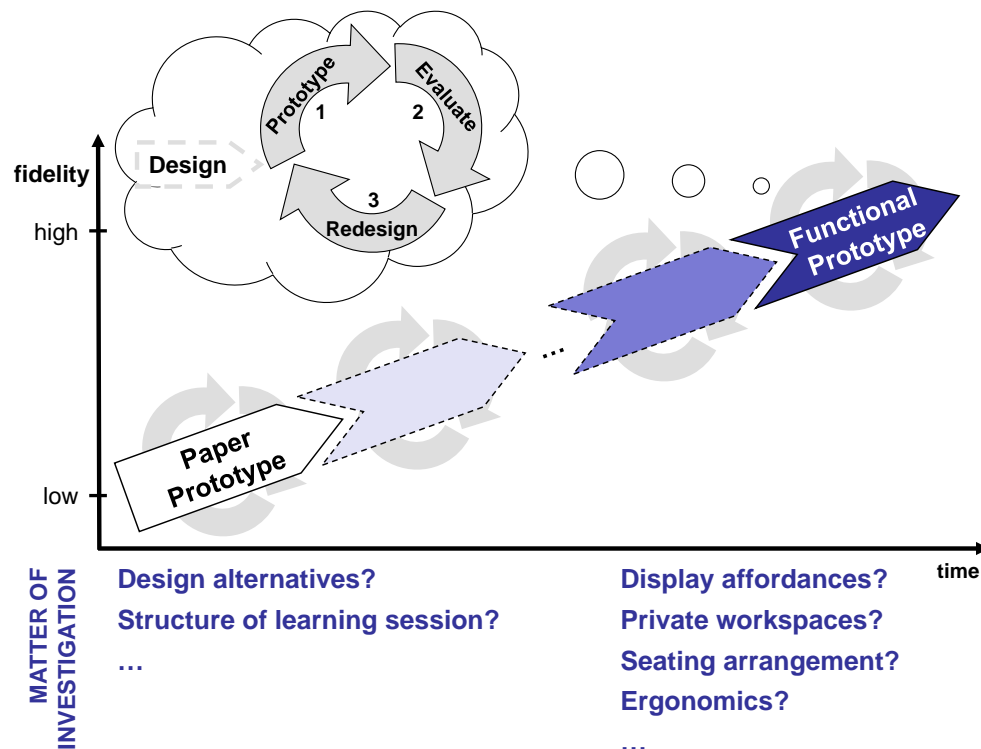
compared different ways of using ‘argumentation diagrams’. The diagrams are graphs consisting of argument/thesis nodes and argumentative links, which represent the debate. In a study they examined the effects of using argumentation diagrams as a means for debate to using the diagram as representation of the debate. Results show that learners who used the argumentation diagram as a representation expressed less personal opinions during the collaboration. In the post-test, in which learners created individual graphs, learners who used the diagram for representational purposes added more examples, consequences and causes. This suggests that the instructions and the specific usage of argumentation diagrams should be chosen according to the specific learning goal.

### 4.1.2 Iterative Design Process

Against this theoretical background our goal was to create an application, which guides learners during the creation of arguments. According to Toulmin (1958) these arguments should consist of claim, grounds and qualification. Furthermore, we aimed at facilitating argument sequencing (Leitão, 2000). To develop an optimized user interface, we followed an iterative design process as proposed by Nielsen (1993). During this design process the user interface is iteratively tested by users (in our case learners) and then refined according to their feedback. That way usability problems can be identified early in the design process and the user interface is gradually improved.

For the evaluation of our design ideas different types of prototypes were used. In the beginning a paper prototype was used. Paper prototypes provide a cheap method for comparing design alternatives (Rudd et al., 1996). Furthermore, paper prototype studies allow to examine the general structure of learning sessions and to observe which steps in the learning process need to be supported. Later in the design process functional (‘high-fidelity’) prototypes were used as they bear a closer resemblance of the final product and therefore are more suitable to test user-interface issues. Figure 4.1 illustrates the iterative design cycle (as proposed by Dix et al., 2004, p. 220) and the use of high- and low-fidelity prototypes at different stages of the development process.

The learning application we envisioned should support two learners during argumentation. The learners are assigned conflicting positions, for example pro and con a certain statement. During the discussion, arguments are noted down in terms of keywords and related to previous arguments. As a result the application shows an overview of the course of discussion. Due to findings in the field of human-computer interaction, we initially chose a tabletop display for this scenario. As opposed to personal devices (e.g. laptops), tabletop displays provide a shared workspace, which is inevitable for argument sequencing. Compared to wall-mounted interactive displays (which are another form of shared workspaces), tabletop displays have been shown to (1) better facilitate balanced participation rates (Everitt et al., 2006) and (2) enable natural face-to-face communication, e.g. in terms of eye contact (Inkpen et al., 2005). Balancing participation was an important design goal because both learners should equally contribute to the discussion in order to maximize learning effects. A natural face-to-face seating arrangement was chosen because it seems to be most suitable for competing teams (Sommer, 1967) and because eye contact is



**Figure 4.1:** Usage of high- and low-fidelity prototypes in the iterative design process.

an important part in interpersonal communication (Nardi and Whittaker, 2002). Therefore, we decided to seat learners on opposite sides of the tabletop display.

### 4.1.3 Paper Prototype Study

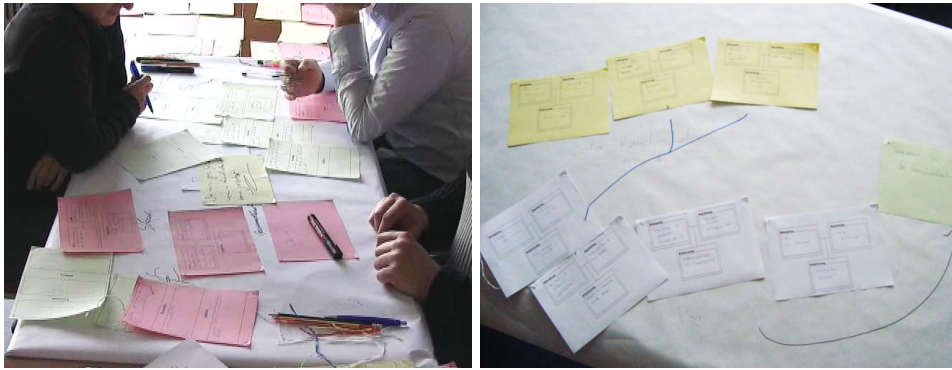
A paper prototype was developed to investigate the following research questions<sup>3</sup>.

*RQ1: How should argument representations look like in order to be comprehensible and at the same time support learners in creating arguments?*

*RQ2: Do learners sequence their arguments sufficiently when they are instructed accordingly?*

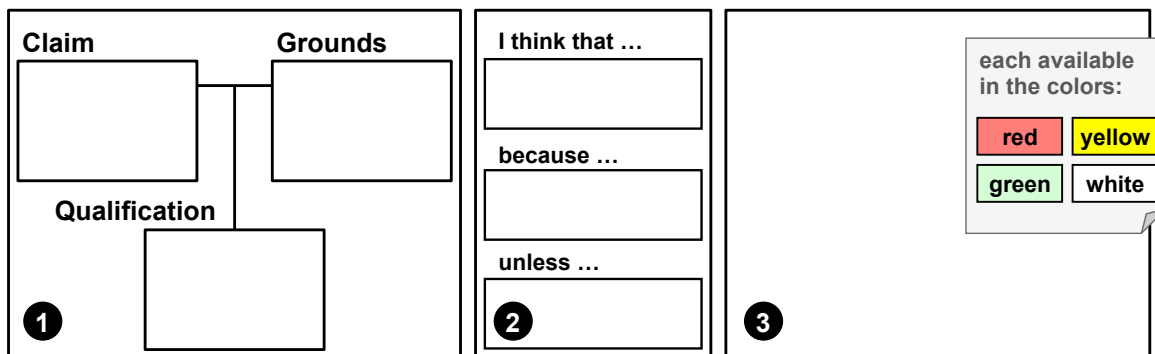
The paper prototype consisted of a table which was covered with a table-sized piece of paper (see Figure 4.2, left). On the table there were different stacks of cardboards, which could be used to note down keywords to summarize an argument. To examine RQ1, we designed three argument representations that were printed on the paper cards. (1) In the 'triangular design' three boxes named 'claim', 'grounds' and 'qualification' were arranged in a triangle (see Figure 4.3, left).

<sup>3</sup> The study research questions RQ1, RQ2, ... are renumerated in each study, which means the numbers are only valid within the study they are reported in. The study research questions are not to be confused with the global thesis questions RQI and RQII as explained in section 3.6.



**Figure 4.2:** Setup in the paper prototype study (left) and grouping of arguments (right).

This design is similar to the collaboration script implemented by Stegmann et al. (2007a), which successfully supported the construction of single arguments in asynchronous online discussions. (2) The 'linear design' showed three boxes that were connected by the clause openers 'because' and 'unless' (see Figure 4.3, center). To be consistent to the sequential structure of the sentence, the boxes and clause openers were vertically arranged. (3) Finally, the 'free design' was an empty paper card, which was provided to allow learners to use their own layouts or design their own argument representations (Figure 4.3, right). Each of the three designs was available in different colors (red, green, yellow and white) to allow distinctions between different kinds of arguments (e.g. pro, con and integration).



**Figure 4.3:** Three design alternatives used in the paper prototype, each available in red, yellow, green and white: (1) triangular, (2) linear and (3) free.

Our second goal was to facilitate argument sequencing. A traditional classroom approach would be instructing the learners to sequence their arguments and providing means to do so (e.g. pens for drawing links between arguments). This case study was intended to investigate whether this approach sufficiently supports argument sequencing (RQ2). Argument sequences could be built in two ways. First, connecting lines could be drawn on the large background paper. Second, flexible spiraled wires could be attached to two paper cards to establish a link between them. While drawing lines may be easier, the wires allow for rearranging the cardboards without breaking the links.

### *Method*

To examine the research questions, a paper prototype study was conducted. Overall N=12 learners in groups of two or three participated in the paper prototype study. 58% of the participants were female. Four of them were graduate students, the remaining eight were doctoral students. Altogether five groups were built, three dyads and two groups of three. The participants were asked to imagine they took a course called 'Learning how to argue'. In the beginning a short introduction was given, in which the structure of arguments was explained. Each learner was assigned a position (e.g. pro or con a given statement). Learners were instructed to create arguments consisting of claim, grounds and qualification, to provide pros and cons, to build argument sequences by relating their own arguments to their learning partner's arguments, and finally to draw a conclusion. The participants were told that in the end the content of the table should represent the course of discussion. There was no time limit to ensure that all groups would be able to reach a conclusion. After the groups had finished their argumentation, there was an open discussion on the paper prototype, which the experimenter started by asking why the according design alternatives were chosen. Later, more general feedback was collected and finally the group was asked to agree on their favorite design.

The discussions were videotaped. The videos were later analyzed to investigate which design alternatives were chosen as argument representations and whether argument sequences were built sufficiently.

### *Results*

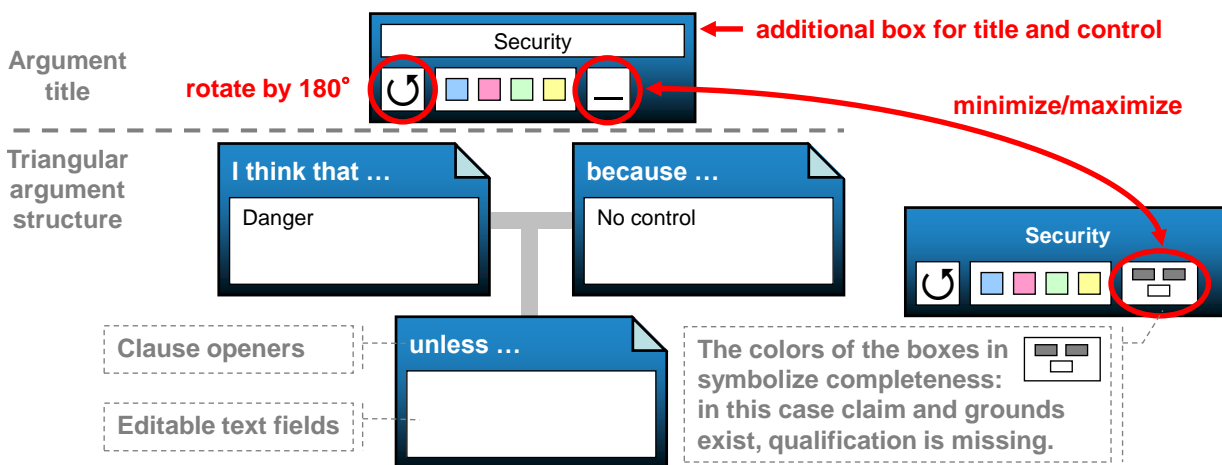
With regard to the argument representation, seven participants chose the triangular design, four chose the linear and one person the free design. All participants stuck to the design they initially chose. Only for the conclusion three of the five groups switched to the free design. When asked about the reasons for choosing the triangular design, the answer was always that the layout creates a good overview of the argument components. One participant mentioned that the linear design suggests a hierarchy that he did not like. In addition, some participants complained about the structure of the sentence, which is imposed on the learners. At the same time, the sentence structure was the main argument named by all participants who chose the linear design. A compromise that was agreed on in several groups was to keep the layout of the triangular design and replace the text by the three clause openers 'I think that ...', 'because ...' and 'unless ...'. Furthermore, it was suggested to add a field for a keyword that summarizes the argument.

Concerning the argument sequencing, two of the five groups used the wires to link arguments in the beginning of the discussion. Both of them decided to draw lines after a while. The remaining three groups only used pens. However, arguments were rarely directly related to previous arguments. Consequently, hardly any argument sequences were built. The common approach was to present arguments without relating them to one another, and in the end structuring the collected arguments. The result of the structuring process was mostly groups of related arguments (Figure 4.2, right) rather than sequences. During the final discussion the participants were asked why they did not/rarely sequence their arguments. The main reason was that finding a direct relationship between arguments was much more difficult than grouping them.

Apart from the two research questions there were some other interesting observations. We expected learners to (spontaneously) discuss the given topic and along the way note down their arguments on the cardboards. Although the instructions were given accordingly, none of the groups immediately started with the discussion. All participants started to silently brainstorm for arguments that supported their position. The arguments were noted down on cardboards but not presented until all group members were finished with the brainstorming. In other words, the learners automatically inserted an individual phase prior to the discussion. Moreover, we observed a seeming correlation between the positioning of argument representations and the number of sequences built. Some groups moved ‘active’ arguments (arguments that are currently discussed) to the center to show them to the discussion partners. In the other groups, arguments remained in the proximity of their creators. These groups hardly built any sequences.

### *Summary and consequences for the ArgueTable redesign*

The feedback of the participants led to a redesign of the interface for the construction of single arguments (see Figure 4.4), which was used in the next case study. Since the next study was done with a functional prototype, controls for minimizing and maximizing the argument representation were added. By clicking on the ‘minimize’ button the argument structure (claim, grounds and qualification) is hidden. Only the title (summarizing keyword) and controls are shown. In the minimized state (Figure 4.4, right) a maximize button shows a thumbnail of three hidden boxes. In the thumbnail empty boxes are white, filled boxes are gray. Furthermore, a rotate button allows users to rotate the argument representation by 180° in order to show the content to the opponent.



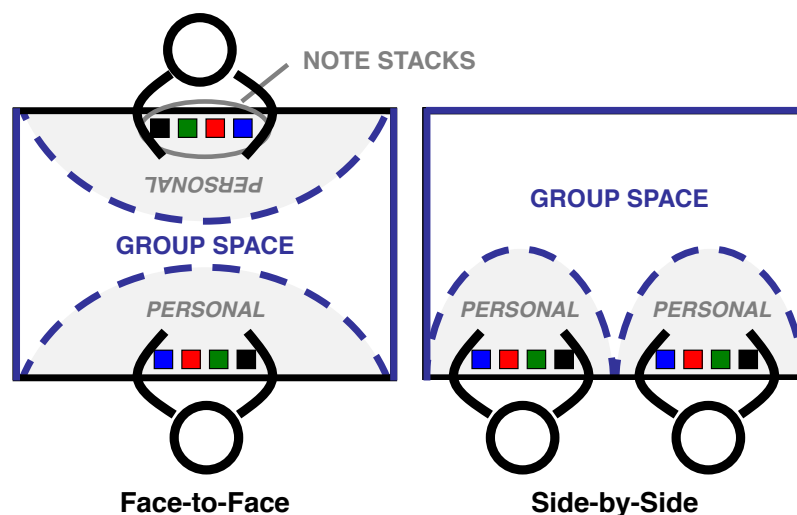
**Figure 4.4:** Redesign of argument representations.

Apart from the redesign of argument representations we identified argument sequencing as difficult task, which is rarely done by learners even when instructed to do so. Therefore, argument sequencing needs to be scaffolded accordingly.



### 4.1.4 Case Studies with Functional Prototypes

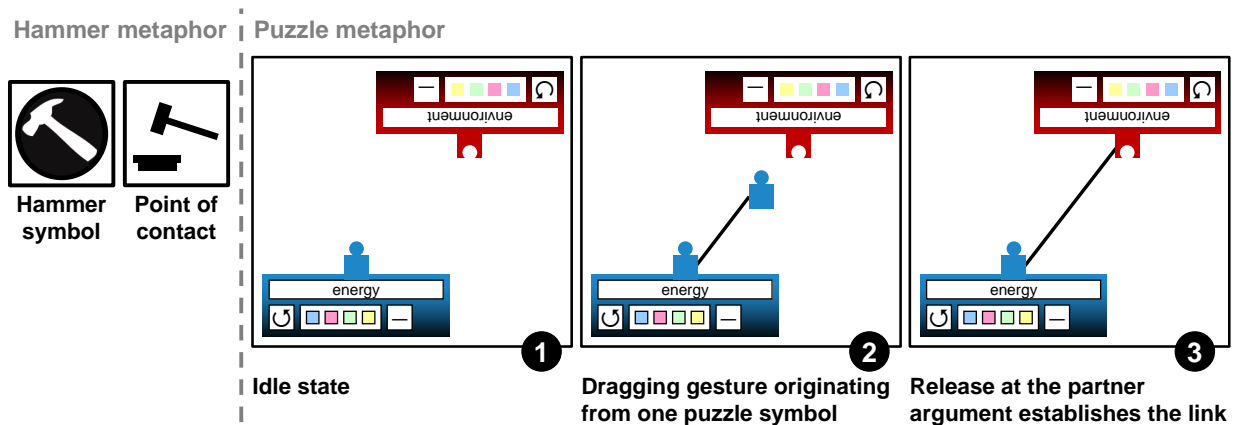
The feedback gained in the paper prototype study was considered during the development of a functional prototype. The functional prototype was used in further case studies to examine how argument sequencing could be better supported. As we had noticed a seeming correlation between the amount of argument sequencing and the position of arguments, the first case study examined the influence of seating arrangement and territories that occur in different seating arrangements (see Figure 4.5)<sup>4</sup>. A video analysis showed that a joined workspace existed in both seating arrangements. However, participants stated that they considered the workspace arrangement in the side-by-side condition to be more suitable for argument sequencing. Negative aspects of the side-by-side seating arrangement were the limited elbow-room and eye contact. The latter was considered to be important for a natural communication. In addition, participants felt observed by their learning partners, which they disliked. A downside of the face-to-face seating arrangement was the text orientation problem, which existed but was not considered to be disturbing because the learners read out the arguments anyway. Consequently, the face-to-face seating arrangement was maintained in the further studies.



**Figure 4.5:** Expected territories in face-to-face and side-by-side seating arrangements.

Furthermore, different metaphors were tried out in order to facilitate argument sequencing. First, the hammer metaphor was implemented to illustrate that arguments should attack previous arguments of the opponent. Pro and con arguments were equipped with hammer symbols that served as starting points for dragging gestures and points of contact, which served as arrival points. As the hammer metaphor was not well understood, the hammer symbols were replaced with puzzle pieces (see Figure 4.6). The puzzle metaphor illustrated that pro and con arguments should be linked. While the puzzle metaphor was well accepted, it did not seem to increase the amount of sequencing.

<sup>4</sup> Due to space limitation the case studies are briefly summarized. A detailed description of the studies is presented in Böhm (2010).



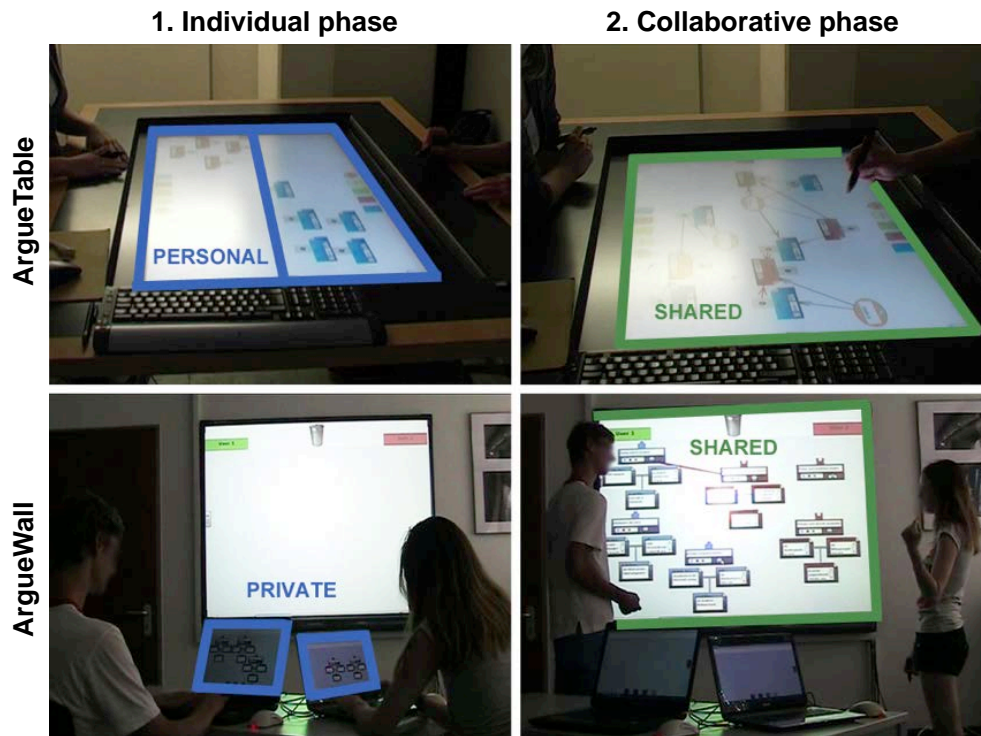
**Figure 4.6:** Hammer metaphor (left) and puzzle metaphor (right) to support argument sequencing.

The case studies helped us to improve the functional prototype but the examined factors did not seem to affect the amount of argument sequencing. However, observations led us to the hypothesis that, as the same physical space was used for both phases, it was not clear enough that there is still further interaction needed in the second phase. We therefore started to think about separate physical workspaces for the two learning phases. This made us question the reasons for choosing a tabletop scenario in the first place. If learners need an individual phase for preparing arguments, why not use laptops in this phase? Additional benefits of using laptops are that they provide a familiar working environment as well as easier text input. An obvious downside would be the necessity of switching to a shared display after the individual phase. At the same time, the display switch could have a positive effect on the argument sequencing: Using separate displays for both phases could emphasize that further interaction is required after the creation of argument notes (which was rarely done on the tabletop display).

## 4.2 ArgueTable vs. ArgueWall

Due to the experience gained in several case studies on the ArgueTable, we conducted a final study in which the tabletop application was compared to a distributed application consisting of laptops and interactive wall (see Figure 4.7)<sup>5</sup>. We decided to use a wall-mounted display instead of a tabletop display as shared workspace because the combination of laptops and a tabletop display did not seem feasible. For instance, as occlusion has to be avoided, it would not be clear where the laptop should be placed. Hinrichs et al. (2007) describe how even much smaller devices such as hardware keyboards can create a barrier between user and tabletop or make interaction clumsy if the keyboard is removed and retrieved when needed.

<sup>5</sup> The ArgueWall was implemented by Christine Wagner (2010). The ArgueTable vs. ArgueWall study was conducted by Sonja Böhm (2010) and Christine Wagner (2010) under supervision of the thesis author.



**Figure 4.7:** Private/personal and shared workspaces in the ArgueTable and ArgueWall conditions.

### 4.2.1 Method

In the study two research questions were examined:

*RQ1: To what extent does the display setting (ArgueTable vs. ArgueWall) affect processes during argument preparation?* We expected the awareness of the learning partner's activities to be higher on the ArgueTable compared to the ArgueWall condition (where the individual work takes place in private workspaces). For instance, on the ArgueTable it should be easier to detect when the learning partner is finished with the individual argument preparation phase than in the ArgueWall condition. At the same time, we expected the individual work to be less disturbed in the ArgueWall condition because there is less distraction by the learning partner's activities.

*RQ2: To what extent does the display (ArgueTable vs. ArgueWall) affect collaborative argumentation?* On the ArgueTable both phases take place on the same tabletop display. On the ArgueWall there is a display switch between the individual phase, which is done on laptops, and the collaborative phase, which takes place at an interactive wall display. Switching the display between the individual and collaborative phase could emphasize that the interaction with the argument nodes is not yet over and that (next to the verbal discussion of the arguments) they should be arranged and linked to each other. Therefore, we expected more argument sequences in the ArgueWall condition. In addition, we were aware of the possibility that only one of the learners stands up and interacts with the wall display. This could lead to unwanted permanent role distri-

butions, such as thinker and writer. Finally, we expected the feeling of confrontation to be higher in the ArgueTable condition due to the face-to-face seating arrangement.

### *Participants and Design*

Ten dyads (N=20) participated in our study. 45% of the participants were female. Their age ranged from 20 to 31 (25 on average). All of them were graduate students. The type of display setting (ArgueTable vs. ArgueWall) was varied using a within-subject design. Consequently, all participants used both types of display settings. The order of conditions was counterbalanced.

As illustrated in Figure 4.7, the ArgueTable condition was realized using a tabletop display for both phases. During the individual phase, the workspace was graphically divided into two workspaces. During the collaborative phase, the same tabletop display served as shared workspace. In the ArgueWall condition two laptops were used in the individual phase. Participants created the arguments next to each other, but without direct sight on the learning partner's display. During the collaborative phase a wall-mounted display was used. Arguments could be transferred to the shared display using a simple dragging gesture. To create argument sequences, the learners had to stand up and move and/or connect their arguments. For the reasons mentioned above, the combination of laptops and a tabletop display did not seem feasible. Thus, we opted for an exploratory study, in which each individual display environment is a reasonable solution instead of creating rather artificial environments that allow a direct comparison.

### *Learning Task and Procedure*

At the beginning of each run the functionalities of the applications were demonstrated by creating exemplary arguments on the topic 'ban on smoking in restaurants'. The functionalities of both applications during the individual and the collaborative phase were identical: By dragging notes from a stack new argument representations can be created, each consisting of the three components claim, grounds and qualification. Argument notes can be minimized and maximized to save space and provide a better overview of all arguments. They can also be freely arranged on the shared, interactive display using direct touch dragging gestures. Furthermore, argument notes can be linked in order to build argument sequences. It is also possible to minimize entire argument sequences.

After demonstrating the functionalities participants were asked to discuss two different topics ('pro/con tuition fees' and 'pro/con speed limits on highways'), each under a different condition. For each topic, learners first should prepare three important arguments that support their position. To ensure that the discussion would be controversial one participant was assigned the pro position, the other one the counter position. After each condition, a questionnaire with items regarding the previous discussion was handed out.

### *Measures*

The subjective assessment was measured with identical questionnaires that were handed out after each condition. For all items a 5-point Likert scale was used from 1 ('completely disagree') to 5

(‘completely agree’). The *awareness* was measured with the item ‘It was easy to detect when my partner was finished with preparing arguments’. To assess the subjective *distraction* the answers of the item ‘I could prepare my arguments undisturbed’ were reversed. Whether the learners felt *observed* by their partners was measured with the item ‘During the argument preparation I felt observed by my learning partner’. *Confrontation* was measured with the item ‘In the discussion I perceived my learning partner as opponent’. In addition the *number of argument sequences* was used as indicator for the quality of argument sequences (cf. Stegmann, Wecker and colleagues (2007). To operationalize this, the number of links between arguments at the end of the discussion was retrieved from log files. Furthermore, in order to examine the role distribution within the dyad, the participants were not given any instructions as to who should interact with the wall display. When directly asked, the instructor would answer ‘At least one person needs to stand up and interact [...]’. From the video recordings it was then retrieved which group members stood up and whether each group member significantly contributed to the verbal discussion. In addition, the videos were analyzed to identify possible *role distributions* in the groups.

## 4.2.2 Results

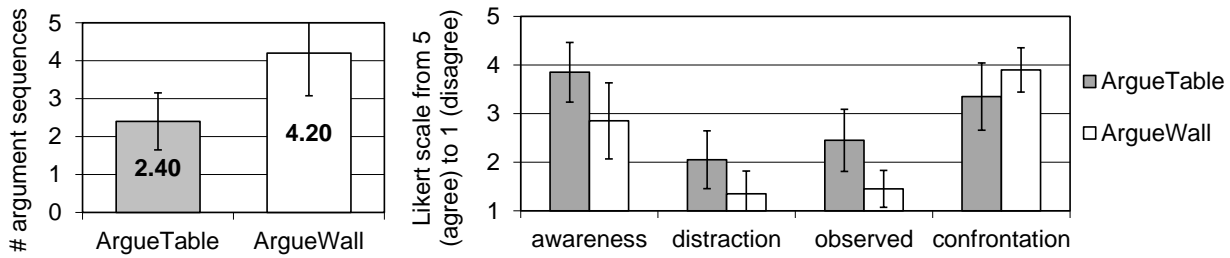
### *Effects on argument preparation*

Regarding the subjective *awareness* the mean answer was  $M=3.85$  ( $SD=1.23$ ) for the ArgueTable and  $M=2.85$  ( $SD=1.57$ ) for the ArgueWall condition (see Figure 4.8, right). A paired-sample t-test showed a significant effect ( $t(19)=2.30$ ,  $p=.03$ , two-tailed,  $r=.47$ ). Therefore, the awareness for the learning partner’s activities was higher in the ArgueTable condition. However, the evaluation of the questionnaire indicated that the awareness was not considered as important factor for the choice of display environment. As for the subjective *distraction*, participants felt more distracted when using the ArgueTable ( $M=2.05$ ,  $SD=1.19$ ) compared to the ArgueWall ( $M=1.35$ ,  $SD=0.93$ ). A paired-sample t-test showed a significant effect ( $t(19)=-2.33$ ,  $p=.03$ , two-tailed,  $r=.47$ ). In addition, participants felt more *observed* on the ArgueTable ( $M=2.45$ ,  $SD=1.28$ ) than in the ArgueWall condition ( $M=1.45$ ,  $SD=.76$ ,  $t(19)=3.45$ ,  $p=.003$ , two-tailed,  $r=.62$ ).

### *Effects on collaborative argumentation*

With regard to the argumentative *discourse quality*, the mean number of argument sequences built on the ArgueTable was  $M=2.40$  ( $SD=1.51$ ) compared to  $M=4.20$  ( $SD=2.25$ ) on the ArgueWall (see Figure 4.8, left). A paired-sample t-test showed a significant effect ( $t(9)=-2.59$ ,  $p=.029$ , two-tailed,  $r=.65$ ). Therefore, on the ArgueWall significantly more arguments were linked than on the ArgueTable. The subjective *confrontation* was  $M=3.35$  ( $SD=1.39$ ) for the ArgueTable  $M=3.90$  ( $SD=0.91$ ) for the ArgueWall condition. A paired-sample t-test showed a significant effect ( $t(19)=-2.34$ ,  $p=.030$ , two-tailed,  $r=.47$ ). Thus, the partner was perceived more as an opponent in the ArgueWall condition and less so in the ArgueTable condition.

Concerning the role distribution, in all groups both learners stood up at the beginning of the collaborative phase and engaged in the discussion. A role distribution (such as one person being



**Figure 4.8:** Number of links between arguments (left) and subjective results (right) in the ArgueTable vs. ArgueWall study.

responsible for the interaction with the application, the other one for contributions with regards to content) was not found.

### 4.2.3 Summary and Discussion

In order to investigate the effects of different display environments on collaborative knowledge construction we compared the two applications ArgueTable and ArgueWall. For this purpose we examined to what extent the display setting affects processes during argument preparation and collaborative argumentation. As expected, the awareness of the learning partner's activities was higher in the tabletop condition. Observations showed that participants only needed a brief look to detect when their partner was finished with preparing his arguments whereas working on a laptop made it necessary to use verbal communication and eye contact. The evaluation of the questionnaire confirmed this finding. However, according to informal interviews that were done at the end of the study, awareness was not considered to be an important factor. Furthermore, learners felt less disturbed while working on the laptops than on the tabletop display in the individual phase. In addition, the feeling of being observed was more present at the ArgueTable. A private workspace, which is not visible to the learning partner, was often mentioned as the most important advantage of the ArgueWall setting.

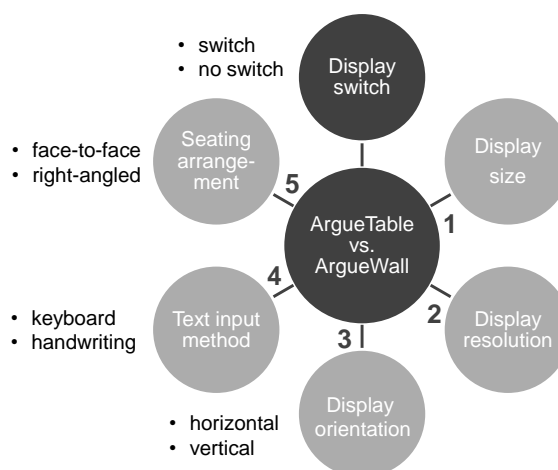
Concerning effects of the display setting in the collaborative argumentation phase and the quality of the argument sequences in particular, the results of the study also showed clear advantages of the ArgueWall: The mean number of sequences was significantly higher on the wall-mounted display. Participants built nearly twice as many links between their arguments than on the tabletop display. The participants' informal feedback confirmed that it was easier to create sequences on the ArgueWall due to a better overview and that an explicit separation of personal and shared workspaces was helpful at this. The need to switch displays between the individual and collaborative phase seemed to increase the concentration on the discussion phase and its associated tasks. Regarding the question of confrontation, the answers showed that the learning partner was less perceived as an opponent in the tabletop condition. According to informal interviews sitting at different tables and working on separate displays during the individual phase triggers the notion of self-dependency. This was surprising because in the ArgueTable condition the learners sat face-to-face, which has been found to be the most suitable seating arrangement for competing

teams (cf. Sommer, 1967). It seems whether one is working on a shared or separate displays has a larger effect on the feeling of confrontation than the seating arrangement. Finally, a role distribution within the dyad could not be detected. In all groups both learners went to the wall-mounted display and interacted with the application. Their part in the discussion was mostly balanced. These observations were confirmed by the informal interviews in which participants considered the distribution of work to be fairly equal.

In conclusion, the study revealed clear advantages of an explicit private workspace for individual phases in collaborative learning scenarios. This aspect was considered more important than the awareness of the learning partner's progress. Moreover, the number of argument sequences was much higher on the ArgueWall, which can be ascribed to both a more clearly arranged workspace and increased concentration due to the display switch between the individual and the collaborative phase.

### 4.3 Follow-Up Studies

The comparison of ArgueTable and ArgueWall showed that the display environment can have significant effects on collaborative processes. The original reason for developing the ArgueWall was the assumption that a display switch emphasizes the tasks in the different phases. However, next to the display switch the two display environments differ in at least five other factors (see Figure 4.9). First, the size of horizontal displays is naturally limited due to problems of reach (Inkpen et al., 2005). Thus, the horizontal display had a smaller size and a lower resolution. Furthermore, the orientation of the shared display varied (horizontal tabletop vs. vertical wall display). In addition, different text input methods were used: handwriting recognition on the tabletop display and hardware keyboards on the laptops. Finally, the seating arrangement varied. Due to the iterative design process (cf. section 4.1.2) a face-to-face seating arrangement was



**Figure 4.9:** Factors varying between ArgueTable and ArgueWall.

chosen for the ArgueTable. On the contrary, learners were sitting at a right angle when using laptops in the ArgueWall condition to ensure privacy and to allow both learners to view the shared display.

This section presents a set of follow-up studies that were conducted to isolate the factor(s) that was/were responsible for the effect on argumentation sequencing. To provide an overview of the series of follow-up studies, the main findings are anticipated at this point. As the display switch was the reason for developing the ArgueWall in the first place, we expected the display switch to be the cause of the effect. Consequently, we aimed for a design with the two conditions (C1) one display and (C2) display switch. All other factors were fixed across both conditions. The first follow-up study was conducted with a low-fidelity paper prototype (cf. ‘*Paper Prototype Follow-up Study*’ in Figure 4.10). That way the text input mechanism was ruled out as factor because the arguments were written on paper cards. In addition, the display size and resolution were left out of the equation. Due to ergonomic reasons a horizontal display was chosen. The seating arrangement was side-by-side.

Study	Prototype fidelity	Seating	Matter of investigation	Reason for follow-up study
ArgueTable vs. ArgueWall	High (functional)	Mixed	Display Switch	6 possible factors
Paper Prototype Follow-up Study	Low (paper)	Side-by-side		
Mixed-Fidelity Prototype Follow-up Study 1	Mixed (Paper arguments on tabletop)	Side-by-side	Display Switch	Higher prototype fidelity needed
Mixed-Fidelity Prototype Follow-up Study 2		Face-to-face		
Mixed-Fidelity Prototype Follow-up Study 3		Mixed	Seating Arrangement	Role of readability?

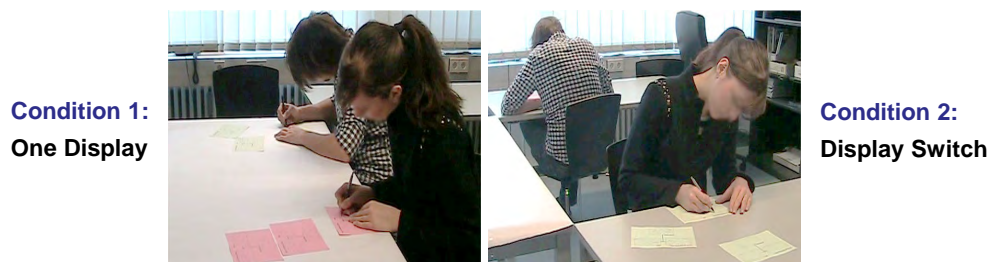
**Figure 4.10:** Overview of follow-up studies.

The paper prototype study could not reveal any effect of the display switch on argument sequencing. One problem was that sequencing paper arguments by attaching wires to them was clearly more cumbersome than dragging gestures on the interactive surface. In addition, we observed that the non-interactive wall and table surfaces did not create the same display affordance as the functional prototype. Consequently, we created a ‘mixed-fidelity prototype’ consisting of paper arguments on a tabletop display. Visual markers on the back of the paper arguments allowed to use dragging gestures on the tabletop display for the argument sequencing. In the following, this study will be referred to as ‘*Mixed-Fidelity Prototype Follow-up Study 1*’. The findings suggest that the main difference between the two setups was the awareness of the partner’s arguments during the individual phase: In the ‘one display’ condition participants could read their partner’s arguments already in the individual phase, which they could usually not when the arguments were created on separate displays. This awareness of the partner’s arguments not only eased argument sequencing, it also allows learners to create arguments that fit to their partner’s arguments. This observation led to another follow-up study with a face-to-face seating arrangement (cf. *Mixed-Fidelity Prototype Follow-up Study 2*’ in Figure 4.10). Again, this study could not verify that



the display switch was the main cause of the difference in the number of argument sequences in the ArgueTable vs. ArgueWall study. A comparison between the two Mixed-Fidelity Prototype Studies suggested that the seating arrangement might be a main factor. This was confirmed by a final follow-up study ‘*Mixed-Fidelity Prototype Follow-up Study 3*’. In the following, the studies are presented in more detail.

### 4.3.1 Paper Prototype Follow-Up Study



**Figure 4.11:** Setting in the paper prototype follow-up study.

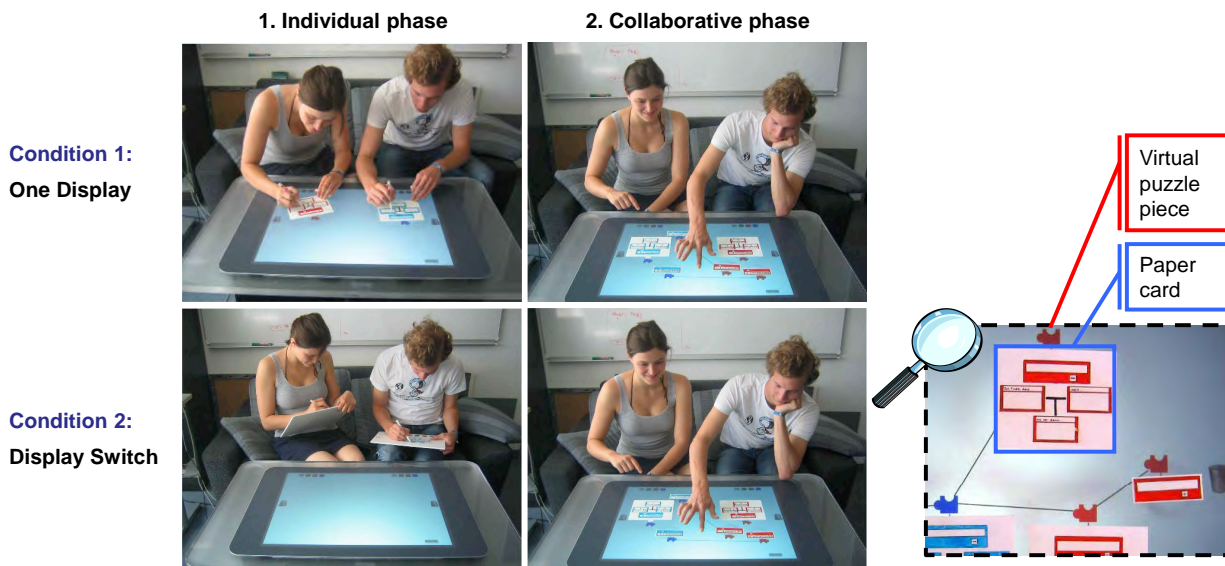
Four dyads (N=8) participated in the study. 38% of the participants were female. Their age ranged from 20 to 58 (34 on average). Five of them were graduate students, one was a house-keeper and two were employees. As explained before the two conditions ‘one display’ and ‘display switch’ were compared<sup>6</sup>. In both conditions arguments were written down on paper cards and linked with wires (as in the paper prototype study that was conducted during the iterative design process of the ArgueTable, which was presented in section 4.1.3). The learning task and procedure was the same as in the ArgueTable vs. ArgueWall study (see section 4.2). In the ‘one display’ condition the individual preparation phase and the discussion phase were done on one table on which the learners sat side-by-side (see Figure 4.11, left). In the ‘display switch’ condition, the preparation phase was done on separate tables with the learners sitting back-to-back (see Figure 4.11, right). For the discussion the learners switched to another table (same setting as in Figure 4.11, left).

Results show that the amount of sequenced arguments was almost the same in the one display (M=3.50, SD=1.29) and display switch condition (M=3.25, SD=1.71). However, two major problems were observed early in the study. Both are related to the nature of the paper prototype, which was chosen to avoid variables such as the text input method, display size and resolution. The first problem was that connecting arguments with wires was obviously more cumbersome than the dragging gestures on the interactive surface. This fiddling with the wires could negatively affect argument sequencing, which is the most important measured variable. In addition, we observed that switching or not switching between tables does not create the same affordance as switching

<sup>6</sup> The paper prototype follow-up study was conducted by Christine Wagner (2011) under supervision of the thesis author.

displays: Whether one is confronted with a computer application that offers a specific functionality or with an empty non-interactive surface seemed to make a significant difference. This difference could not be counteracted by instructing participants to imagine the non-interactive prototype was an application. To eliminate both problems, we decided to conduct another study with a mixed-fidelity prototype.

### 4.3.2 Mixed-Fidelity Prototype Study (Side-by-Side)

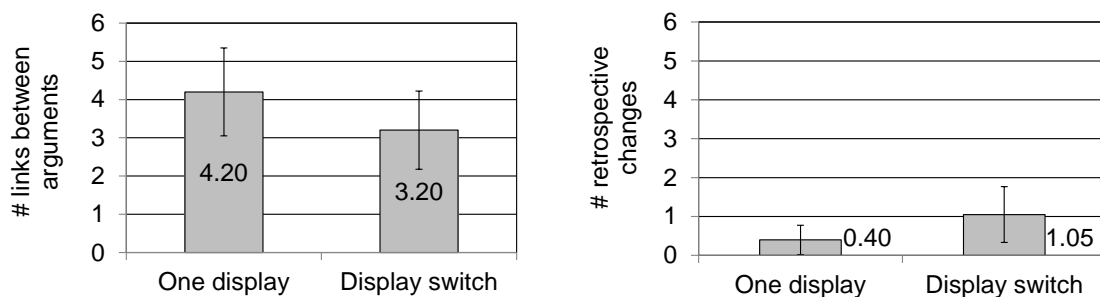


**Figure 4.12:** Setting in the mixed-fidelity prototype follow up study 1.

To solve the problems inflicted by the pure paper prototype a ‘*mixed-fidelity prototype*’ was implemented<sup>7</sup>. It consisted of both low and high fidelity components. As in the paper prototype arguments were created on paper cards (low fidelity). However, the argument sequencing was done using the tabletop functionality (high fidelity). The preparation phase was either done on the tabletop surface (one display condition) or on paperboards, which were designed to represent tablet computers. Both the argument cards and the tablet paperboards were covered with foil. That way arguments stuck to the tablet paperboards due to static electrification even when they were held vertically. To realize the mixed-fidelity prototype, visual markers were placed on the back of each argument paper card. When arguments were placed on the tabletop, the application recognized the marker and showed a puzzle piece above the paper card (see Figure 4.12, right). Dragging gestures from one puzzle piece to another established links between the arguments. Furthermore, unlike the previous paper prototype study, participants could minimize their arguments by placing a finger on the minimize button on the argument card. This was done for two reasons: First, because of space limitations on the tablet paperboards, and second, to remind the participants that they were using a prototype and thus counteract the affordance problem.

<sup>7</sup> The practical part of this work was conducted by Sarah Tausch (2010) under supervision of the thesis author.

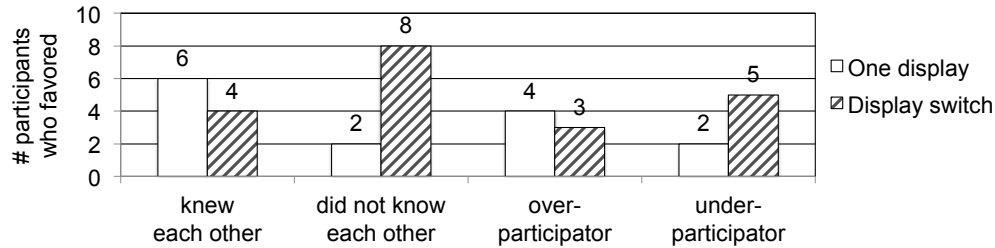
Ten dyads (N=20) participated in the study. 30 % were female. Their age ranged from 20 to 32 (24 on average). Five dyads knew each other, the other five did not. The learning task and procedure was the same as in the study before (see section 4.3.1). The results do not confirm an effect of the display switch on the amount of argument sequences. In the ‘one display’ condition, the number of links between arguments was  $M=4.20$  ( $SD=2.30$ ) compared to  $M=3.20$  ( $SD=2.04$ ) in the ‘display switch’ condition (see Figure 4.13). The difference was not significant. However, we noticed that the awareness for the partner’s arguments played an important role. Whether the learners were able to read their partner’s arguments during the individual phase seemed to influence their ability to create sequences in the collaborative phase. In the first instance this is a confounding factor: It does not explain the difference in the ArgueTable vs. ArgueWall study because the learners were not able to see their partner’s arguments when using laptops (which was the condition with more sequences). Therefore, our observations had two consequences. First, we conducted another follow-up study to investigate the isolated factor ‘display switch’ with a face-to-face seating arrangement, which reduces the awareness of the partner’s arguments during the individual phase (subsequent section 4.3.3). Second, we recognized that the readability is an important factor, which needs further attention (cf. section 4.3.4).



**Figure 4.13:** Number of links between arguments (left) and retrospective changes (right) in the Mixed-Fidelity Prototype Study 1.

In addition, there were other interesting findings. During argument preparation there were significantly more changes made to arguments after they had been created in the display switch condition ( $M=1.05$ ,  $SD=1.43$ ) compared to the one display condition ( $M=.40$ ,  $SD=.75$ ,  $t(19)=-2.16$ ,  $p=.04$ , two-tailed,  $r=.44$ ). This suggests that either participants deliberated their arguments more carefully before writing them down on the shared display or they were reluctant to change arguments retrospectively when the learning partner’s could watch. Furthermore, as half of the groups knew each other and half did not, it was possible to compare their preferences for the display setting. Among the participants that knew their partners beforehand, six favored the one display setting, four favored the display switch. Among the participants that did *not* know their partners the display switch was favored more often (eight times compared to only two preferences for the one display condition). The same tendency was observed with regard to speaking times. Video analyses showed that in seven of the ten dyads one person clearly spoke more than the other. Among the participants who dominated the conversations, four preferred the one display condition, three the display switch. Among the ‘under-participants’ only two preferred the one display condition (compared to five who preferred the display switch condition). Thus, it

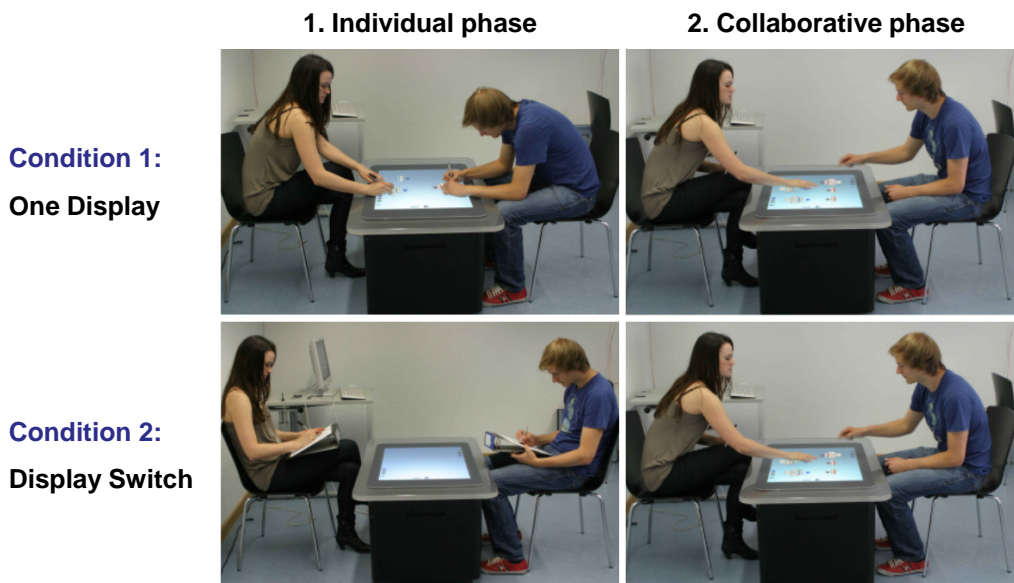
seemed that private workspaces, which are offered in the display switch condition are particularly appreciated by groups who do not know each other as well as rather reticent learners.



**Figure 4.14:** Preferences for the display setups in different subgroups of participants.

### 4.3.3 Mixed-Fidelity Prototype Study (Face-to-Face)

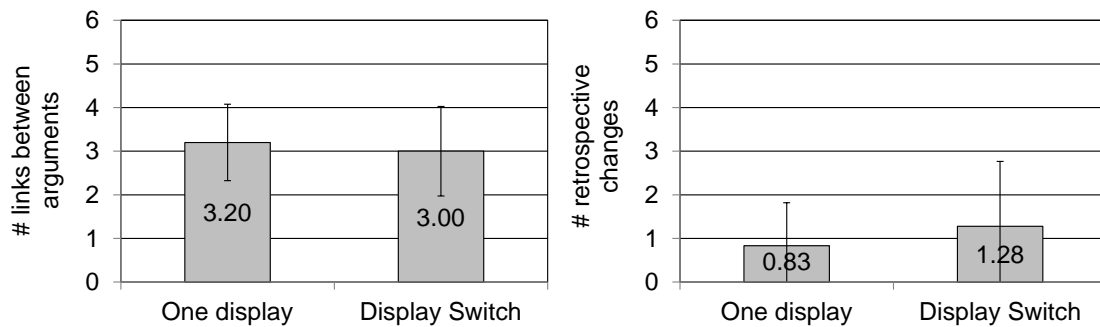
To avoid the awareness of the learning partner's arguments during argument preparation, a similar study was conducted using a face-to-face seating arrangement (see Figure 4.15). Again, ten dyads participated in the study. 40 % were female. Their age was 27 on average. This time the dyads were paired with participants they did not know before the study. The learning task and procedure was the same as in the studies before (see section 4.3.1).



**Figure 4.15:** Setting in the mixed-fidelity prototype follow up study 2.

There was no significant effect of the display switch on the number of argument sequences. In the one display condition the number was  $M=3.20$  ( $SD=1.75$ ), in the display switch condition it was  $M=3.00$  ( $SD=2.05$ , see Figure 4.16). This indicates that the display switch was not the

main cause of the increased number of argument sequencing in the ArgueTable vs. ArgueWall study. Another finding was that - as in the previous study - the number of retrospective changes to argument nodes during the individual phase was significantly higher in the display switch condition ( $M=1.28$ ,  $SD=1.49$ ) compared to the one display condition ( $M=.83$ ,  $SD=.99$ ,  $t(17)=-2.20$ ,  $p=.04$ , two-tailed,  $r=.45$ ).



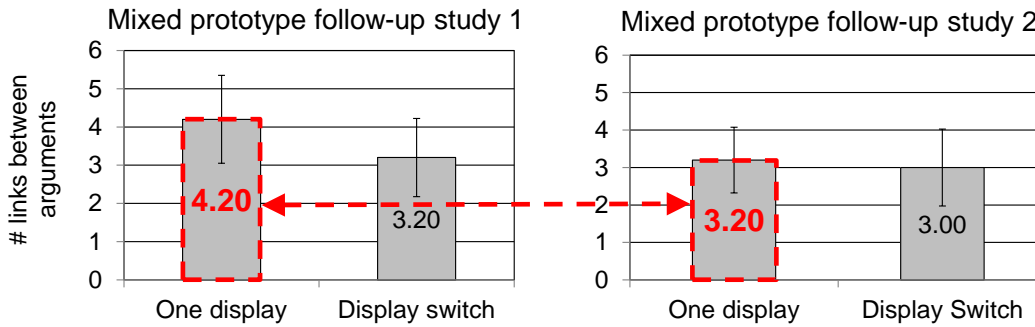
**Figure 4.16:** Number of links between arguments (left) and retrospective changes (right) in the Mixed-Fidelity Prototype Study 2.

#### 4.3.4 Mixed-Fidelity Prototype Study (Side-by-Side vs. Face-to-Face)

As the display switch did not seem to be the crucial factor for the increased amount of argument sequencing in the ArgueWall condition, other factors must have played a significant role. As mentioned before, the first mixed-fidelity prototype study indicated that the awareness of the learning partner's arguments has an effect on argument sequencing. In the second mixed-fidelity prototype study where the participants sat face-to-face this effect was not observed. Therefore, the amount of argument sequences in the one display conditions of both studies were quite different: In the first mixed-fidelity prototype study (side-by-side) the numbers of argument sequences in the side-by-side 'one display' condition was  $M=4.20$  ( $SD=2.30$ , cf. section 4.12) compared to  $M=3.20$  ( $SD=1.75$ , cf. section 4.15) in the second mixed-fidelity prototype study (face-to-face, see Figure 4.17). This supports the hypothesis that the seating arrangement and in particular the readability of the partner's arguments plays a major role. To verify this hypothesis the effect of the seating arrangement on argument sequencing was examined in one last study<sup>8</sup>.

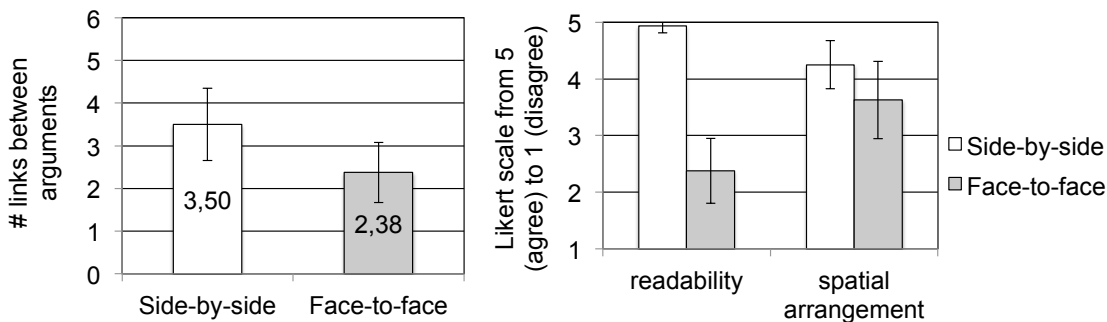
Eight dyads ( $N=16$ ) participated in the study. 44% of the participants were female. Their age ranged from 20 to 26 (23 on average). The learning task and procedure was the same as in the study before (see section 4.3.1). However, this time the seating arrangement was varied. In one condition the participants sat face-to-face, in the other condition they sat side-by-side. In both conditions the individual phase was done on the tablet paperboards. Thus, the conditions are equal to the display switch conditions of the previous two studies respectively.

<sup>8</sup> The practical part of this work was conducted by Susanne Zirker (2011) under supervision of the thesis author.



**Figure 4.17:** Comparison between mixed prototype follow-up studies 1 and 2.

Results show that the number of argument sequences was higher in the side-by-side condition ( $M=3.50$ ,  $SD=1.69$ ) compared to face-to-face ( $M=2.38$ ,  $SD=1.41$ , see Figure 4.18). A paired-sample t-test showed a significant effect ( $t(7)=2.55$ ,  $p=.04$ , two-tailed,  $r=.69$ ). Subjective assessment through questionnaires confirmed that the *readability* was better when sitting side-by-side ( $M=4.94$ ,  $SD=0.25$ ) compared to face-to-face ( $M=2.38$ ,  $SD=1.15$ ). A Wilcoxon signed-rank test showed a significant effect ( $Z=-3.45$ ,  $p=.001$ ). In addition, the question whether the *spatial arrangement* of arguments facilitates argument sequences was significantly more agreed to in the side-by-side condition ( $M=4.25$ ,  $SD=.86$ ) than in the face-to-face condition ( $M=3.63$ ,  $SD=1.36$ ,  $Z=-2.41$ ,  $p=.02$ ).



**Figure 4.18:** Number of links between arguments (left) and subjective results (right) in the Mixed-Fidelity Prototype Study 3.

### 4.3.5 Summary and Discussion

This section examined different display settings for an application that supports co-located collaborative learning. It implements two collaboration scripts that effectively foster argumentative knowledge construction. Based on findings from previous studies in the field of human-computer interaction, our first vision was a tabletop application ('ArgueTable'). However, an iterative design process with small case studies led to a display environment consisting of laptops and a shared interactive wall display ('ArgueWall'). A study showed that the ArgueWall mitigated

a problem that occurred on the ArgueTable through all case studies: the lack of argument sequencing. We hypothesized the display switch between individual preparation on laptops and collaborative phase on the wall display to be the cause of the improvement as the display switch was the motivation for the ArgueWall in the first place. However, ArgueTable and ArgueWall differ in various factors such as the text input method, display size, resolution and orientation. We therefore conducted several follow-up studies to narrow down the cause of the effect.

The studies produced findings on different levels, which are addressed in the following three subsections. At first the difficulties of producing generalizable results are illustrated. Second, the usefulness of iterative design processes is discussed. The focus of both discussions is on the number of argument sequences, a question which we got tied up over and which caused the series of follow-up studies. In the end, the findings that were revealed despite the study design dilemma are summarized.

### *The Study Design Dilemma*

The series of studies unfolds several problems that go along with a comparison of display environments. First, it is extremely difficult to isolate single display factors. For instance, different displays have different text input capabilities (e.g. hardware keyboards on laptops with haptic feedback vs. on-screen keyboard or handwriting recognition on interactive surfaces without haptic feedback). As a consequence the time and effort spent on text input can vary across conditions, leaving more or less time for the actual learning task (in our case argument sequencing). This can in return lead to further consequences, which may be difficult to interpret. The problem of comparability also applies to fairly similar display types. Even the comparison of vertical and horizontal interactive surfaces is not straightforward although they provide the same input and output technologies. Despite the technological equivalence there are significant differences such as ergonomic aspects or the problem of reach, which naturally limits the size of tabletop displays. The size of the display can have various consequences, for example on the overview or the number of items that are created. In our case this would be the number of arguments, which again can affect the number of argument sequences. This throws researchers into a dilemma: For instance, in a clean study design the size of the vertical display needs to be deliberately downsized to match the horizontal display in order to isolate the factor ‘display orientation’. However, in this case the capabilities of the vertical ‘lab’ display do not reflect the capabilities of larger, more realistic wall displays as they would be used in schools.

Due to this dilemma we chose a two-fold approach. We started with a less clean study design by comparing two display environments (ArgueTable and ArgueWall). Neither of them was artificially adapted for the study. As a consequence they varied in several factors. Afterwards, we tried to narrow down the cause of the effect with artificial prototypes in several follow-up studies. However, although each of the follow-up studies specifically investigated one isolated factor, it still turned out to be difficult to produce generalizable effects. For instance, while the handwriting solution used in the follow-up studies was deliberately chosen to avoid a bias caused by different text input methods, it might have amplified the text orientation problem. As handwriting is generally harder to read than typed text, it is not sure whether the effect also

applies to the functional ArgueTable prototype, which recognizes the handwriting and displays block letters. This series of studies depicts a complex of problems, which put into question whether effects of display types or environments on collaborative processes can be generalized.

### *Iterative Design Process*

It is hardly possible to produce general recommendations regarding the choice of display environment. The only alternative for determining the ideal display environment for one's learning application is going through an iterative design process. While the iterative design process of the ArgueTable led to an improved display environment, the ArgueTable studies also raise the question whether small-N case studies can reveal all effects on collaboration. For instance, the role of the seating arrangement was not identified during the iterative design process although the seating arrangement was specifically investigated in one of the case studies: Even though the orientation problem in the face-to-face seating arrangement was acknowledged, it was not considered to be disturbing because the arguments were read out loud. In the end the consideration of all benefits and drawbacks of both seating arrangements spoke in favor of face-to-face. There are two possible explanations. One is that the significance of the problem was not discovered due to the small sample size. Another explanation is that the consequences of the text orientation problem were less severe on the functional prototype case study because of the printed letters. In this case we still do not know what the main cause of the effect on the number of argument sequences in the ArgueTable vs. ArgueWall study was. It may have been a combination of different factors. It could, after all, still have been the display switch but the effect did not occur with the mixed-fidelity prototype due to a lack of affordance (as with the paper prototype). To sum up, although iterative design processes may not reveal all effects, we still argue that (a) it is worthwhile because it led to an improved display environment and (b) it is a more suitable approach than attempting to produce general display recommendations.

### *Findings*

Despite the design dilemma the studies revealed several findings. First and foremost, the studies showed that the display environment can play a significant role in computer-supported collaborative learning. Even though we were not able to identify individual display environment factors as cause of the effect on argument sequencing, there are two aspects, which seemed to be important. Both are problems related to text on tabletop displays, which turned out to be severer than expected. The first one is the text orientation problem, which exists in face-to-face seating arrangements. Although this issue has been addressed before (for instance by Kruger et al., 2003), we underestimated its impact because participants neither placed great importance on it in case studies nor did we observe significant effects in video analyses. Second, text input on tabletop displays using both an on-screen keyboard and handwriting recognition were so unsatisfactory that negative impact on the learning process cannot be ruled out. Again, the text input problem has been addressed before (for instance by Hinrichs et al., 2007) but we underestimated its significance. Considering these issues it seems questionable whether tabletops are ideal for text-based collaborative learning tasks.



Aside from the number of argument sequences, which moved more and more into the focus of attention, there were other interesting findings. For instance, in the last two follow-up studies the amount of retrospective changes to arguments was significantly higher when learners were provided with private workspaces. Subjective results confirm that private workspace are appreciated, in particular by reticent learners and if collaborating learners do not know each other. On the contrary, the awareness for the learning partner's activities seemed to be less important than expected. These findings caused us to have a closer look at assets and drawbacks of different display environments in a different learning scenario.

## 4.4 Examining the Importance of Display Environment Factors

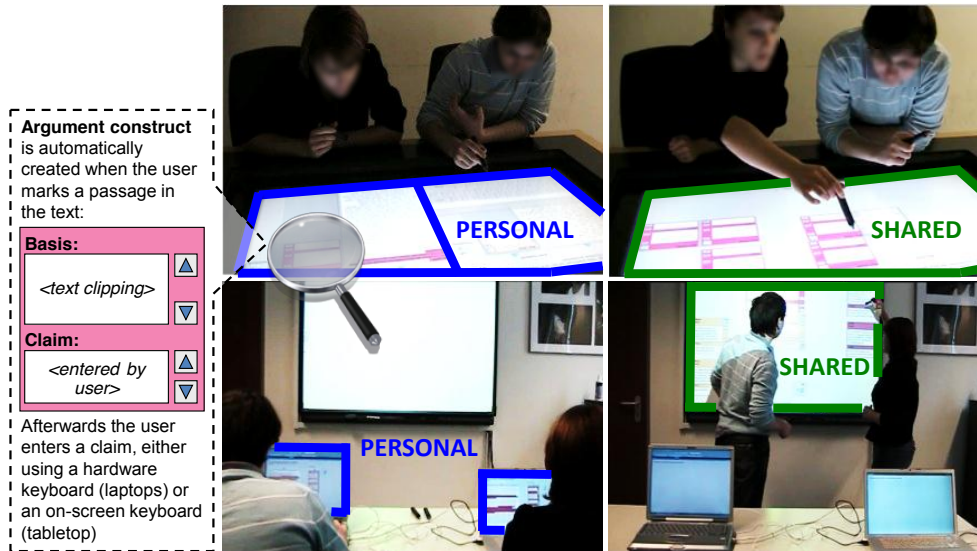
In the ArgueTable and ArgueWall studies the learners' appreciation of different display characteristics (such as having a private workspace, being aware of the learning partner's activities or having to switch display between phases) was different than initially expected. In addition, the awareness of the learning partner's activities (in our case the preparation of arguments) seemed to affect the collaborative phase (in our case argument sequencing). To gain further insights on these matters and to verify whether these findings also apply to other learning tasks another user study was conducted. For this study a different collaborative learning scenario was used, which was adopted from the collaboration script 'structured academic controversies' (SAC). As the name suggests, the script also targets argumentation. However, this time pairs do not consist of two opponents. Instead each pair prepares a joint statement that is later discussed with an opposing pair. Again, two display environments were compared regarding their effects on individual and collaborative work<sup>9</sup>. In one condition, laptops were provided in the individual phase while the collaboration was done on a wall display (see Figure 4.19). In the second condition, a tabletop display was used throughout the whole process. To allow multiple people to work individually at the same time, the tabletop display was split into personal workspaces during the individual phase. This was expected to facilitate a higher awareness of the partner's activities compared to the laptop condition.

The first research question was how the different levels of awareness of the co-learners' activities influence the learner's individual work. Possible effects could be that learners adjust the time spent on the individual work or be more distracted by their partners' activities. The second research question was how the users perceived the different assets and drawbacks of each display. For example, do people accept display switches between individual and collaborative phases if the different displays are more suitable for the tasks at hand? Results show that the awareness was higher in the tabletop condition. However, there was no indication that the increased awareness affects the individual work, neither in a positive way (e.g. by motivating learners to create as much content as the learning partners) nor negatively (by distracting the learners from their work).

---

<sup>9</sup> The practical part of this work was conducted by Mona Metzner (2010) and Marc Mühlbauer (2011) under supervision of the thesis author.

The awareness was also considered less important than all other examined factors such as text input, facilitation of teamwork and communication or a display switch between phases.



**Figure 4.19:** Personal (left) and shared (right) workspaces on the tabletop (top) and distributed application (bottom).

#### 4.4.1 Structured Academic Controversies

For the purpose of the study, the learning strategy 'Structured Academic Controversy' (SAC) (Johnson and Johnson, 1994) was chosen to define the structure of the learning session. Its basic idea is to induce socio-cognitive conflict and to support cognitive strategies to solve the conflict, which is beneficial for individual knowledge construction (Nastasi and Clements, 1992). It is intended for four learners who collaborate in different group formations. In the first phase, all learners individually read handouts to prepare their position for a subsequent discussion. The positions are pre-defined, which means that two learners take the pro and the other two take the counter position. In the second phase, two pairs are built, one for each position. In the final phase, the positions are presented to the other dyad, discussed and merged into one final statement.

#### 4.4.2 Display Environments

Two different applications were built to support the SAC learning strategy using the same display environments as in ArgueTable and ArgueWall: (1) a tabletop application and (2) a distributed application consisting of laptops and a wall-mounted smartboard. In the *tabletop application*, the two learners building a dyad are seated next to each other. They use the same tabletop display for the first (individual) phase as well as during the second (pair) phase. During the individual

phase the tabletop display is graphically split into two personal workspaces, which implies a high awareness of the partner's activities. After the individual phase the two personal workspaces are merged into a joint workspace in which the dyad performs the collaborative task. The *distributed application* provides separate displays for the phases, each providing the workspace that fits best to the according task. The individual phase is done on laptops, which provide private workspaces that the learning partner cannot see. Since a shared workspace is needed in the collaborative phase, the learners switch to an interactive wall display for the second phase. Consequently, learners have to physically relocate and switch displays between the two phases.

The graphical user interfaces were kept as similar as possible to specifically compare the characteristics of the display settings: Both applications allow users to mark text excerpts, which support an argument. Each marked text excerpt is automatically converted into an argument construct that shows the text clipping on the top and allows the user to add an argument claim on the bottom (see Figure 4.19, left). In the collaborative phase these argument constructs are used to prepare a statement.

### 4.4.3 Method

In the study two research questions were examined:

*RQ1: How do the different levels of awareness of the co-learners' activities influence the learner's individual work?* In the tabletop environment the users' interactions with the application are visible to both learners. Therefore, it is easy to detect how many arguments the partner has created or when she is finished with the task. We expected that learners adjust the time spent on the individual work and are more distracted by their partners' activities in the tabletop condition.

*RQ2: How do learners perceive the different assets and drawbacks of each display?* We were interested in the learners' preferences regarding the different display types. For example, do people accept display switches between individual and collaborative phases if the different displays are more suitable for the tasks at hand?

#### *Participants and Design*

A within-subject design was chosen to compare the effects of the two display settings. 26 participants were split into dyads. Each dyad completed two SAC sessions, one with the tabletop application (condition 1) and one with the distributed application (condition 2). The topics were pro/con 'prohibition of first-person shooter video games' and 'school uniforms'. The order of conditions as well as the allocation to the topics was counterbalanced.

#### *Learning Task and Procedure*

The learners were asked to complete the first (individual) and second (pair) phase of the SAC script once per condition. In each condition, the two participants were given one article con-

taining pro and con arguments. In the first phase, the participants individually read the text and extracted arguments that supported their position. When both participants were finished they moved on to the next phase, in which they discussed their arguments and agreed on a list of final arguments. According to the SAC strategy, this list would then be presented to a second dyad, which took the counter position. As we were not interested in any effects occurring in this last phase, the session was stopped after the dyad phase due to time limitations.

### *Measures*

In terms of quantitative data, log files and video recordings were used to measure the number of arguments built by each learner, the time needed for the individual and collaborative phase as well as the number of interactions with the shared display during collaboration. Furthermore, a comparative questionnaire was handed out after each condition to measure the subjective awareness of the partner's activities in the individual phase as well as implications such as motivation and distraction. The questionnaire consisted of several statements, which had to be answered on a 5-point Likert scale (5='completely agree', 1='completely disagree'). In the end, there was a final questionnaire in which participants were asked to weigh the importance of different factors that play a role when choosing a display environment.

### *Statistical tests*

Paired samples t-tests were applied to parametric data. To evaluate the single questionnaire items, Wilcoxon signed-rank tests were used.

## 4.4.4 Results

### *Awareness*

Although there are different ways of measuring awareness on shared displays objectively (e.g. the number of conflicting actions (Hornecker et al., 2008)), these measures were not applicable to our study as we examined the awareness of the partner's activities during individual work. Thus, the subjective awareness was measured using three questions like 'I was always aware of my partner's activities'. The reliability of the awareness scale was sufficiently high (Cronbach's  $\alpha = .52$ ) to be regarded as interval scaled. Therefore, we used a paired-sample t-test to compare the two display settings. Learners using a laptop during the individual phase had a lower subjective awareness ( $M=2.29$ ,  $SD=0.89$ ) compared to learners using a tabletop display ( $M=3.03$ ,  $SD=0.73$ ,  $t(25)=-4.53$ ,  $p<.001$ , two-tailed,  $r=.49$ ). Thus, the perceived awareness was higher in the tabletop condition. However, the question 'I would have liked a better overview of my partner's activities' was rather negated ( $M=2.50$ ,  $SD=1.21$  for laptops and  $M=2.00$ ,  $SD=1.06$  for the tabletop). A Wilcoxon signed-rank test showed no significant difference. When asked which display environment was better in terms of awareness, 76.9% of the participants named the tabletop. The remaining 23.1% abstained.

### *Distraction*

Another possible implication of increased awareness is distraction. When learners are given the possibility to see their partner's activities, their concentration might suffer. To measure the distraction we used two questions, e.g. 'I could always create arguments undistractedly' and reversed the scale. The reliability of the distraction scale was sufficiently high (Cronbach's  $\alpha = .57$ ) to be regarded as interval scaled. A paired-sample t-test showed that learners felt less disturbed when using a laptop during the individual phase ( $M=1.27$ ,  $SD=0.41$ ) than when using a tabletop display ( $M=1.85$ ,  $SD=0.91$ ,  $t(25)=-3.08$ ,  $p=.005$ , two-tailed,  $r=.12$ ). However, the distraction was generally quite low (although higher in the tabletop condition).

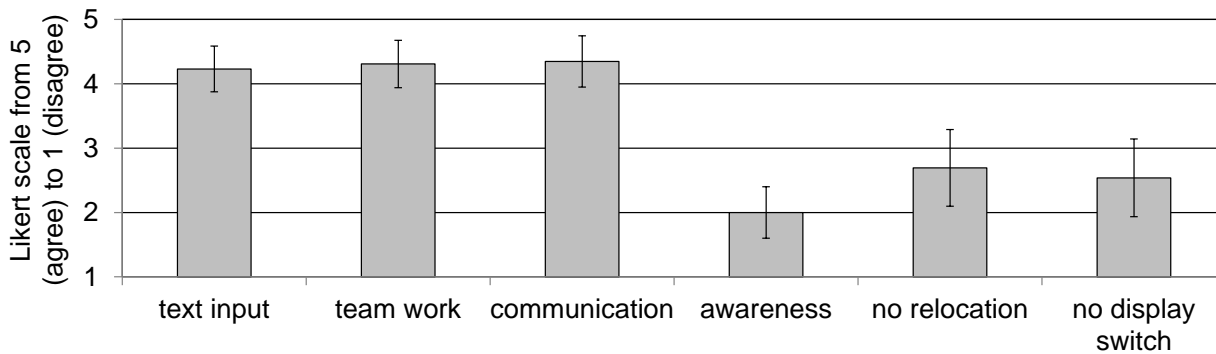
### *Motivation and Participation*

According to the objective results, the mean difference in the time the two learners needed to complete the individual phase was 99.85s ( $SD=113.32s$ ) in the tabletop condition. In the laptop/wall condition the difference tended to be smaller ( $M=63.92s$ ,  $SD=76.38s$ ), but not significantly. Furthermore, the difference in the number of arguments created by the two learners was slightly higher ( $M=.69$ ,  $SD=.75$ ) in the tabletop than in the laptop/wall condition ( $M=.46$ ,  $SD=.66$ ). Although the differences are not significant due to the high standard deviations, the tendency is inverse from what we anticipated (higher discrepancies in the tabletop condition where learners could see their partner's progress). Consequently, there is no indication that the participants adjusted the number of arguments or the time spent on the individual phase to their partner's activities in the tabletop condition (where they were more aware of it).

Regarding the question 'I was influenced by the number of arguments my learning partner created', the answers were generally very low. A Wilcoxon signed-rank test showed that in the tabletop condition the mean answer was significantly higher ( $M=1.71$ ,  $SD=1.00$ ) than in the laptop/wall condition ( $M=1.27$ ,  $SD=.53$ ,  $T=31.00$ ,  $p=.047$ ,  $r=-.28$ ). Regarding the question 'I adjusted my pace of work to my partner', the answer was also significantly higher on for the tabletop ( $M=2.75$ ,  $SD=1.15$ ) compared to the laptop/wall condition ( $M=2.00$ ,  $SD=1.06$ ,  $T=9.00$ ,  $p=.005$ ,  $r=.39$ ). Finally, the question 'I felt observed by my partner' was  $M=1.12$  ( $SD=.33$ ) in the laptop/wall condition compared to  $M=1.67$  ( $SD=.92$ ) on the tabletop. There was no significant effect.

### *Collaborative Phase*

It was also observed whether roles or task sharing patterns occurred in the different conditions. Due to frequent findings from related work (Everitt et al., 2006, Inkpen et al., 2005, Rogers and Lindley, 2004), we were aware of the possibility that only one learner would stand up and interact with the wall display. To measure to what degree the participation in the collaborative phase was balanced, the discrepancy in the number of interactions between the two learners was taken into account. The mean difference in the number of interactions with the wall display was 4.25 ( $SD=4.16$ ) compared to 5.00 ( $SD=4.41$ ) with the tabletop display. (As a reference: The total number of interactions with the shared display per person and condition ranged between 1



**Figure 4.20:** Agreement to the question how important different display factors are.

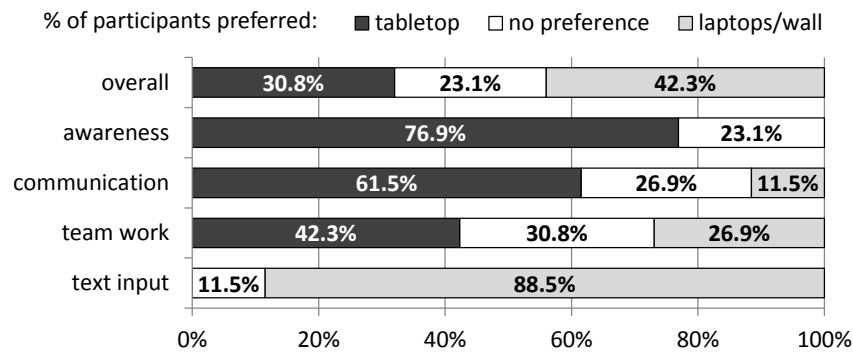
and 29,  $M=9.31$ ,  $SD=6.93$ ). A paired samples t-test showed no significant effect of the display environment on the balance of participation. Observations indicate that the participation was fairly balanced in both display settings. When using the distributed application, all participants except one stood up and interacted with the wall display. In eleven out of 13 cases both group members stood up at the same time. Furthermore, all group members were substantially involved in the discussion. A probable reason for the balanced participation at the wall display is that both learners prepared content in the individual phase, which made them feel equally committed to the follow-up task.

### *Importance of Display Environment Factors*

After both conditions were completed, the participants were asked to rate the importance of six factors on a 5-point Likert scale. The factors were (1) convenience of text input, whether the display supported (2) teamwork and (3) communication, (4) awareness for the partner's activities as well as not having to (5) physically relocate or (6) switch displays. The results are shown in Figure 4.20. A Friedman's ANOVA showed significant differences in the weighting of the different factors ( $\chi^2(5)=91.24$ ,  $p<.001$ ). Wilcoxon tests were used to follow up this finding. The three factors 'text input', 'teamwork' and 'communication' were rated significantly more important than the other factors. Furthermore, the factor 'no relocation' was rated significantly more important than 'awareness' ( $T=113.00$ ,  $p=.017$ ,  $r=-.33$ ). Finally, having 'no display switch' was also significantly more important than 'awareness' ( $T=94.5$ ,  $p=.043$ ,  $r=-.28$ ). The preferences for the two display environments are shown in Figure 4.21.

## 4.4.5 Summary and Discussion

The study compared two display settings (tabletop vs. laptops and wall display) and its effect on collaborative as well as individual work. The main differences between these display environments were the level of awareness of the partner's activities in the individual phase and whether there was a display switch between the individual and collaborative phase. Observations and questionnaire results indicate that an increased awareness on the tabletop display existed but did



**Figure 4.21:** Preferences for display environments.

not have the expected impact on the individual work. Learners were neither distracted by their partner's activities, nor did they try to create as many arguments as their learning partner or finish in the same time. Furthermore, awareness was rated significantly less important than all other factors such as difficulties in entering text or having to switch displays. The factors that participants considered most important were (1) text input (which was the major usability issue in the tabletop condition), as well as whether the display facilitates (2) teamwork and (3) communication. Having to switch displays and physically relocating was rated significantly less important than these factors (but still significantly more important than awareness of the learning partner's individual activities). In other words, people are willing to switch displays between phases if they are provided with workspaces that are optimized for the task at hand.

## 4.5 Lessons Learned

A major decision in the design of co-located collaborative learning applications is the choice of the display environment. Several previous studies have shown that different types of displays have various advantages and disadvantages (e.g. Inkpen et al., 2005). However, it is hard to foresee how important the individual pros and cons of a certain display environment in a specific collaborative scenario are, which makes the choice very difficult. This particularly applies to *scripted* collaborative learning scenarios, which often involve individual *and* collaborative phases. For instance, should individual work take place on personal devices, which offer private, undisturbed workspaces, or should a shared display be divided into personal workspaces to enhance the awareness of the partner's activities? Moreover, should collaboration be supported with tabletops because they facilitate balanced participation and natural communication, or with wall displays as they provide a better overview?

The studies presented in this chapter show that the choice of display environment plays an important role in the development of collaborative learning applications. In the following, the main findings of both studies are summarized and contrasted with previous findings from related work. In the process the collection of studies on the ArgueTable and ArgueWall (sections 4.1, 4.2 and

4.3) will be referred to as '*ArgueTable studies*'. The last study (cf. section 4.4) will be called '*SAC study*'.

With regard to objective results, the *ArgueTable* studies showed that the choice of display environment affected how many arguments were sequenced, which was the main task in the collaborative phase. While we were not able to determine exactly which display characteristic was responsible for this effect, the finding illustrates the role of the display environment in co-located collaborative learning. Another interesting finding is that in the *ArgueTable* vs. *ArgueWall* study as well as in the *SAC* study, both participants stood up and engaged with the interactive wall display. Although we expected imbalances due to findings from related work (e.g. Rogers and Lindley, 2004 or Everitt et al., 2006), we could neither observe an imbalance of participation nor a role distribution such as thinker and writer. A likely explanation is that both learners equally prepared for the collaborative phase using their laptops. Consequently, after the switch to the wall display it contained arguments created by both group members, which might have invited both learners to interact. Therefore, even though several previous studies have found collaboration on tabletop displays to be more democratic than on wall displays, this cannot be generalized to all collaborative scenarios.

In addition, there were some interesting findings with regard to learners' preferences for different display characteristics. The *SAC* study indicates that learners are willing to switch displays and even physically relocate when this provides them with more suitable display types for the respective tasks. An increased awareness of the learner's activities, which is facilitated by shareable user interfaces, was against our expectations rated significantly less important in our scenario. At the same time, private workspaces were appreciated, in particular by reticent learners and in groups of learners who do not know each other. However, there were indications that the desire for private workspaces depends on the learning task: Participants did not feel observed by their learning partners in either of the conditions of the *SAC* study. On the contrary, there was a significant difference for the same question in the *ArgueTable* vs. *ArgueWall* study. Interviews with participants suggest that the reason for these ambivalent findings might be the nature of the task. While in the *ArgueTable* studies the task was to argue *against* each other, learners had to prepare a *joint* position in the *SAC* study. This might explain why learners did not feel observed in the *SAC* study although they were sitting side-by-side and had a better overview of their partner's activities compared to both *ArgueTable* and *ArgueWall*.

While it is not sure whether these results can be generalized to other learning scenarios, it is important to be aware of such potential effects when choosing a display environment for collaborative learning applications. In particular, we would like to stress the benefits of private workspaces as literature in human-computer interaction disproportionately emphasizes benefits of shareable user interfaces such as awareness.



# Chapter 5

---

## Implicit Guidance in Multi-Display Environments

During the iterative design process of the ArgueTable the idea of using different physical displays to implicitly guide learners through different phases emerged. The underlying hypothesis is that each display or workspace has its own affordance: Its design and functionality suggests which activities are expected from the learners. Consequently, using different displays for different phases might emphasize which activities should be done in which phase. If this assumption proves true, the choice of display could provide a valuable supplement in addition to other explicit guidance mechanisms (such as those presented in the subsequent chapters 6 and 7).

Guidance through the display environment is comparable to ‘software wizards’ or ‘assistants’, which lead the user through a sequence of virtual screens. Several HCI patterns come into play here: Each screen makes a *simple impression* on the user, information is presented *just in time* and *incrementally revealed* (Borchers, 2001). While software wizards and other applications make use of this step-by-step guidance to improve usability, the same principle can be used to guide learners through collaboration scripts using physical displays. For instance, private workspaces could be used to support phases in which learners should work individually (e.g. reading and preparation phase) while shared workspaces are provided for collaborative phases. We consider this an *implicit* guidance mechanism as opposed to explicit guidance through graphical user interface components (for example the triangular argument layout in the ArgueTable, which explicitly prompts learners to fill in claim, grounds and qualification).

**Chapter Question:**

How can multi-display environments with private and shared workspaces be used to implicitly guide activities of collaborating learners?

This chapter<sup>1</sup> investigates such implicit guidance, which happens through switching from one physical display to another. At first, we present a distributed application called Note&Share, which was designed to facilitate knowledge sharing. Afterwards two experiments are described, which examined the effects of Note&Share on the knowledge sharing process. In the first study Note&Share reduced the number of misunderstandings and increased the participant's confidence in the solution. In the second study significantly more knowledge was shared with the group as compared to the whiteboard or pen-and-paper conditions. Therefore, both studies indicate that Note&Share guides learners through activities, which facilitate knowledge sharing processes.

## 5.1 Note&Share: Facilitating Knowledge Sharing

Note&Share uses separate displays to enforce a protocol in order to enhance knowledge sharing. The application consists of private workspaces for each learner (laptops) and a shared interactive wall display. In the following it is described how these workspaces are used to support knowledge sharing (section 5.1.1) and how Note&Share is implemented (section 5.1.2)<sup>2</sup>.

### 5.1.1 Concept

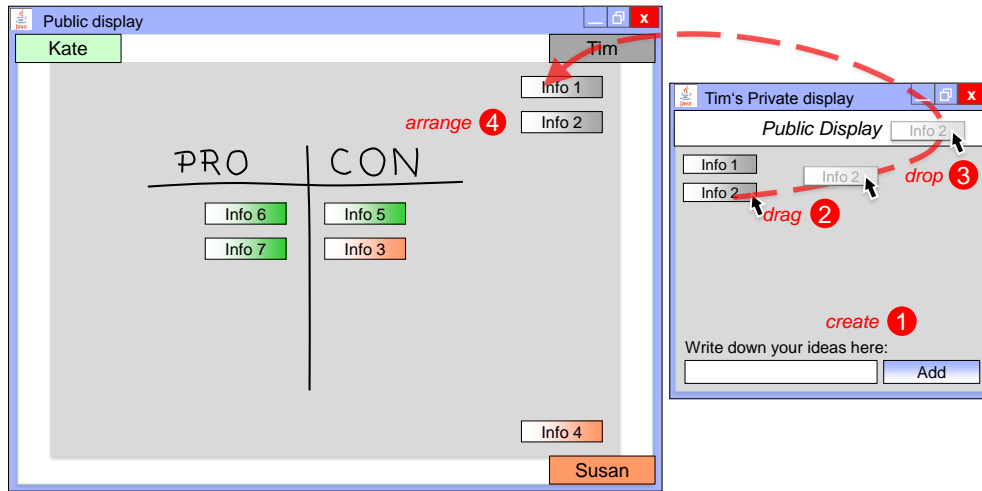
Knowledge sharing is regarded as an important process of effective collaborative learning (Weinberger et al., 2007b). In meetings that strive for a convergence of people's knowledge, opinions or ideas, a common problem is that collaborators share their knowledge ineffectively (Weinberger et al., 2010). Certain information that is held by only one group member is likely to be ignored and not regarded during decision making (Stasser and Stewart, 1992). Furthermore, it has been found that factors like the order of speakers have a comparatively high impact on group decisions (Weisband, 1992).

Such effects may be countervailed by a protocol that tells each group member to first individually structure their knowledge (prior to or at the beginning of the meeting) and to share it with the group in a second phase (Stegmann et al., 2007b). We have built a distributed application called Note&Share, which is designed to facilitate such a protocol. It consists of a personal workspace (laptop) for each group member in addition to a shared workspace (smartboard) for the collaborative phase. The laptops can be used to structure knowledge by creating keywords privately and thus without being influenced by other group members. Later the keywords can be sent to the shared smartboard with a simple drag-and-drop gesture. That way collaborators externalize their individual knowledge based on keywords, which they verbally explain during the discussion. On the touch-sensitive wall display the keywords can be freely arranged and linked to each other,

---

<sup>1</sup> Parts of this chapter have been published. The design of Note&Share as well as the first study have been presented in (Streng et al., 2010).

<sup>2</sup> The practical part of this work was conducted by Sonja Böhm (2009).



**Figure 5.1:** Interaction with Note&Share (Schematically).

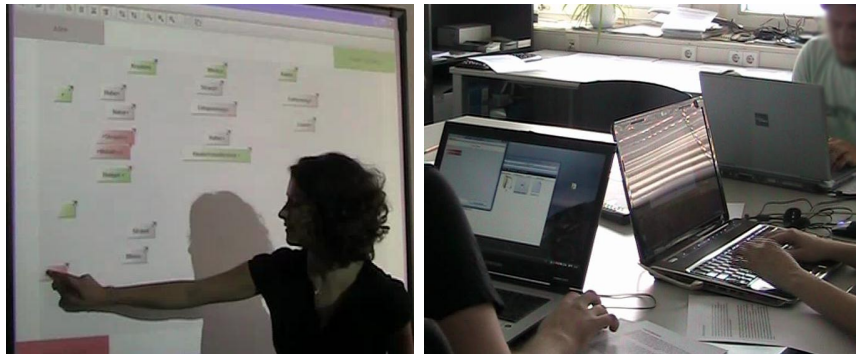
for instance in concept maps, which have been successfully used to visualize knowledge (Cañas et al., 2004).

Depending on the task at hand the keywords can comprise ideas, simple facts or stand for clusters of ideas and knowledge. Thus, Note&Share is a flexible tool that may not only be used for knowledge sharing but also for various other purposes such as brainstorming or decision making.

### 5.1.2 Implementation

Note&Share is designed to support effective knowledge sharing where the knowledge of all participants is equally taken into account. It consists of one laptop per person and a wall-mounted, interactive smartboard for the whole group (see Figure 5.2). Prior to the discussion the laptops are used to create a list of keywords, which represent the owner's knowledge. The smartboard serves as a shared, touch-sensitive display, which represents the group knowledge and thus supports knowledge exchange.

Clients running on all laptops are connected to the smartboard via wireless LAN to allow an easy information exchange. Information transfer is initiated using the Gateway Interaction Technique (Guinard et al., 2007). Each client window has a 'gateway' bar on the top edge, which represents the shared display. After creating keywords that represent relevant knowledge (step 1 in Figure 5.1), they can be dragged onto the gateway in order to transfer it to the smartboard. Using the same interaction technique, keywords can be sent back to the laptops using the smartboard's gateways, which are distributed along the edges (see Figure 5.1, left). Keywords sent by a group member pop up next to the corresponding gateway. The moderator who is operating the smartboard can arrange the keywords in the center and, for instance, create a mind map to visualize and structure the shared knowledge.



**Figure 5.2:** Note&Share Smartboard (left) and Laptops (right).

## 5.2 Hidden Profile Experiment

### 5.2.1 Method

To examine the effects of shared as well as personal displays on knowledge sharing processes a study was conducted in which Note&Share was compared to two other tools that are commonly used during knowledge sharing processes: pen and paper and a dry-erase whiteboard. The following research questions were investigated:

*RQ1: To what extent does the availability of private and shared displays affect knowledge sharing processes?*

*RQ2: To what extent does the availability of private and shared displays affect the quality of decisions?*

#### *Participants and Design*

The three different tools pen-and-paper, whiteboard and Note&Share were compared in a within-subject design. To avoid a bias caused by the order of conditions the tools were used in a counterbalanced order. Eight groups of four ( $N=32$ ) participated in the study. All participants were university students, 14 female and 18 male. Most of them ( $n=26$ ) were between 19 and 25 years old, the others between 26 and 30.

#### *Learning Task and Procedure*

For the purpose of this study, three so-called hidden profile tasks with well-defined solutions were created. In hidden profile experiments, each person is given a different subset of information on several options which the group has to choose from (Stasser and Stewart, 1992). In this experiment the tasks were: Finding the best (1) job candidate, (2) holiday destination and (3) wall paint for a nursery. The information was provided in case descriptions from which the learners had to extract the information that was relevant for the decision. The cases were randomly assigned

to the conditions. The key property of hidden-profile experiments is that the reader of a single case description would choose the wrong option (e.g. not the best job candidate or wall paint) because only a subset of information is considered. Only if all aspects are taken into account the best candidate is found. Thus, knowledge sharing is required to find the best solution.

Each group of four had to solve all of the three hidden profile cases. Three group members, the so-called ‘analysts’, were given the different case descriptions. The fourth group member was assigned the role of a moderator. Moderators did not get case descriptions but were in charge of consolidating all information. After the case descriptions were handed out, the analysts were given as much time as they needed to read them and create keywords for pro and con arguments of all candidates. Afterwards the moderator was in charge of a discussion in which the group had to agree on one option.

In all conditions case descriptions were handed out on paper. Additional paper and pens were available to allow all analysts and moderators to take notes. This was done because it is considered important to provide users with “[...] *their existing repertoire of representations and artefacts when interacting with shared displays. This includes everyday physical tools like pens and paper, communication devices, resources like notes and documents.*” (Rogers and Rodden, 2003). In the pen-and-paper condition no additional tools were provided. Learners only worked with pen and paper in their private workspace and no shared workspace was available. In the whiteboard condition the moderator was allowed to write and draw on a whiteboard. The whiteboard served as shared workspace for the groups in addition to the paper-based private workspaces. In the Note&Share condition analysts had to note down keywords on laptops (i.e. their private workspace) and later send them to the smartboard (the shared workspace), which was controlled by the moderator. Thus, a shared representation of the group’s knowledge only existed in the whiteboard and Note&Share conditions. Another important difference between the conditions is that (according to the core idea of Note&Share) the laptops provide private workspaces, which peers cannot see, whereas there is a higher awareness of the peers’ activities in the other conditions. Working a case took between six and 20 minutes (13 minutes on average) per tool. The whole session could take up to one hour, including introductions in the beginning and prior to each condition.

### *Measures*

There were two sources of quantitative data. In order to assess the quality of the collaboration, the *number of shared arguments* and the *correctness of decision* were recorded. The total number of arguments in each hidden profile case was 15, some being mentioned in only one case description, others being redundant. Thus, if less than 15 arguments were shared information was lost. If more than 15 arguments were identified, different statements for the same arguments were misunderstood as different arguments.

In addition, post-questionnaires were used to measure the participants’ subjective assessment of the three tools. The participants’ *confidence* in the solution was measured with the item ‘The best solution was found’. The item ‘I had a good overview of all arguments’ was used as indicator for the *overview*. Furthermore, the participants were asked to assess different *usability* criteria. All of

these questionnaire items had to be answered on a 5-point Likert scale ranging from 1 (completely disagree) to 5 (completely agree). Finally, they were asked to rank the tools according to which one they liked best (*popularity*).

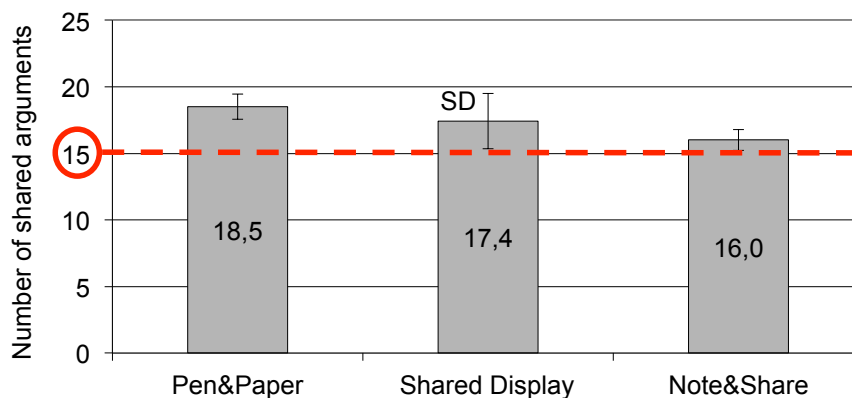
### Statistical Analyses

Analysis of variance with repeated measurements were used to reveal overall effects of the tools (within-subject factor). For further pairwise comparisons paired sample t-tests were applied. If other statistical tests were applied (e.g. to compare frequencies), the type of the statistical test is mentioned in the results section. The quality of collaboration and outcome was measured per group, whereas the subjective evaluation was done with the individual participants as unit.

## 5.2.2 Results

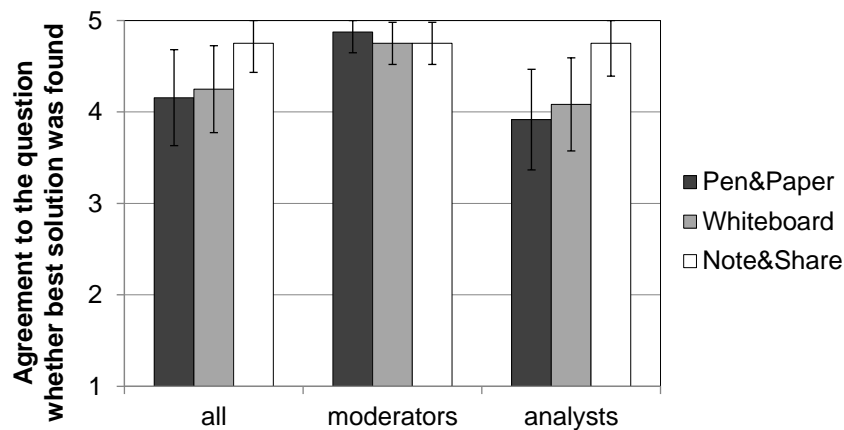
### Quality of collaboration and outcome

Using pen-and-paper the mean *number of shared arguments* per group was 18.5 (SD=1.87), 17.4 (SD=4.16) in the whiteboard and 16.0 (SD=1.55) in the Note&Share condition (see Figure 5.3). Consequently, the main problem was identifying redundant arguments (not information loss as expected). Note&Share tended to be closer to the optimum than the other two conditions, which suggests that misunderstandings occurred less frequently.



**Figure 5.3:** Number of shared arguments.

Looking at the *correctness of decision*, the best solution was found in 19 of 24 cases. Among the other five cases, three were in the pen-and-paper condition and one in each of the other conditions. Three times the wrong conclusion was made in the group's first case, twice in the last case. A  $\chi^2$ -test revealed no significant differences between the tools regarding the amount of best solutions. The reason for the wrong outcome was either unidentified redundancies (occurred three times) and/or groups started to weigh arguments according to their personal judgment instead of comparing the number of pro and con arguments (which happened twice). Information loss,



**Figure 5.4:** Results of study 1. Participants' confidence in the quality of the result.

which is the effect we expected in the first place, only occurred once. Interestingly, unidentified argument redundancies were also quite common in cases where the final result was correct. Only in five cases the number of shared arguments was correct, four of them were in the Note&Share condition.

### *Subjective evaluation*

Regarding the *confidence* that the best solution was found (cf. Figure 5.4), a within participants analysis of variance showed a significant effect for the tool ( $F(2,62)=5.92$ ,  $p<.05$ ). Paired samples t-tests revealed that the agreement was significantly higher for Note&Share ( $M=4.75$ ,  $SD=.57$ ) compared to whiteboard ( $M=4.25$ ,  $SD=.95$ ,  $t(31)=-3.09$ ,  $p=.004$ , two-tailed,  $r=.49$ ) and pen-and-paper ( $M=4.16$ ,  $SD=1.05$ ,  $t(31)=-2.77$ ,  $p=.01$ , two-tailed,  $r=.44$ ). No significant effect was found between whiteboard and pen-and-paper ( $t(31)=-.53$ ,  $p=.60$ , two-tailed,  $r=.69$ ). An unpaired samples t-test showed that moderators rated the pen-and-paper condition significantly higher ( $t(30)=2.40$ ,  $p=.001$ , two-tailed,  $r=.56$ ) than the analysts. Figure 5.4 illustrates an interesting interaction effect: Moderators generally agreed more to the question whether the best solution was found than analysts (which makes sense). However, in the Note&Share condition the analysts' agreement was as high as the moderators'.

The tool also had a significant effect on the *overview* of all arguments ( $F(2,62)=9.03$ ,  $p<.001$ ). According to paired samples t-tests learners rated the overview of shared arguments significantly better in the Note&Share condition ( $M=4.61$ ,  $SD=.80$ ) as compared to pen-and-paper ( $M=3.53$ ,  $SD=1.32$ ,  $t(31)=-3.72$ ,  $p=.001$ , two-tailed,  $r=.56$ ). Furthermore, the overview was significantly better on the whiteboard ( $M=4.28$ ,  $SD=.99$ ) compared to pen-and-paper condition ( $t(31)=-2.64$ ,  $p=.01$ , two-tailed,  $r=.43$ ). There was no significant effect between whiteboard and Note&Share ( $t(31)=-1.69$ ,  $p=.10$ , two-tailed,  $r=.29$ ).

Finally, participants were asked to rank which tool best fulfilled different *usability* criteria. The whiteboard was considered as being the easiest to use (first-ranked by 48%), followed by

Note&Share (35%) and pen-and-paper (16%). However, Note&Share was considered the most helpful (best-ranked by 65%) followed by the whiteboard (35%). Note&Share was also by far the most *popular* tool (first-ranked by 84%, the other 16% voted for the whiteboard). A Friedman-Test for related samples showed a significant difference between the ranks of the three tools ( $N=32$ ,  $\chi^2=40.11$ ,  $df=2$ ,  $p<.001$ , two-tailed). A post hoc Wilcoxon-Test revealed significant differences between whiteboard and pen-and-paper ( $N=32$ ,  $Z=4.51$ ,  $p<.001$ ) as well as between Note&Share and pen-and-paper ( $N=32$ ,  $Z=5.015$ ,  $p<.001$ ).

### 5.2.3 Summary and Consequences

The experiment showed that analysts were significantly more confident in the quality of the results when using Note&Share. Looking at the objective results, the number of shared arguments tended to be closer to the number of the arguments in the hidden profile case when using Note&Share (although not significantly). This results from the fact that more redundant arguments were identified, which means that group members were able to express themselves more clearly leading to fewer misunderstandings during the discussion.

Although a reduction of misunderstandings is an important aspect of supporting knowledge sharing processes, we were not able to prove that Note&Share increases knowledge sharing. The reason was a combination of several factors. First, we underestimated the power of pen and paper as a strong aid of memory, which helps structuring decision-making tasks (Brodbeck et al., 2007). Second, knowledge sharing was required to find the correct solution - a fact that was known to the participants. In addition, the moderators served as information pool. As a consequence of these factors, information loss did not occur at all and potential benefits of Note&Share compared to traditional tools could not be shown. We therefore conducted a follow-up experiment with a more natural open-ended problem solving task.

## 5.3 Experiment with an Open Problem Solving Task

To examine whether Note&Share increases the degree to which knowledge is shared with the group, a follow-up experiment with an open problem solving task was conducted<sup>3</sup>. Prior to the second study, the usability and functionality of Note&Share was improved according to the feedback gained in the first study. Most importantly, drawing functionality was added, which provided moderators with additional means to structure the shared workspace. The first prototype had only allowed moderators to arrange and connect keywords. Second, scrollbars were added to the shared workspace to increase the available space. Furthermore, analysts were enabled to send multiple keywords to the smartboard simultaneously. Finally, the limitation of characters per keyword was increased from 30 to 45.

---

<sup>3</sup> The practical part of this work was conducted by Günther Obexer (2009) under the supervision of the thesis author.



### 5.3.1 Method

In the follow-up study the following research questions were investigated:

*RQ1: To what extent does the availability of private and shared displays affect knowledge sharing processes?*

*RQ2: To what extent does the tool (pen and paper vs. regular whiteboard vs. Note&Share) affect how much knowledge is retained until after the discussion?*

#### *Participants and Design*

As in the first experiment Note&Share was compared to a dry-erase whiteboard and pen and paper using a within subject design. Again 32 people participated in groups of four. To avoid a bias caused by the order of conditions, the three tools were used in a counterbalanced order. 28 participants were female, four male. 53% were university students, the others employees or high school students. Seven participants were between 16-18 years old, 13 between 19-25, 10 between 26-30 and two between 30-40.

#### *Learning Task and Procedure*

To avoid the above-mentioned problems caused by the well-defined hidden profile tasks, a discussion of open-ended problems was chosen as task for the second study. The discussion was more realistic than the rather artificial hidden profile task that turned out to prevent information loss (at least in combination with pen-and-paper). The groups again consisted of three analysts and a moderator, who was in charge of the knowledge sharing process. The three analysts were given essays with different subsets of arguments as preparation for the verbal discussion. This allowed us to reconstruct the processing of arguments. Each analyst was assigned a position regarding a certain statement. That way a controversial discussion was ensured.

For each topic ten arguments were distributed among the analysts, five pro and five con arguments (see Figure 5.5). Of the five pro arguments, two were *'joint arguments'*, which means that they were included in all handouts and thus given to all three analysts (cf. 'Pro 1' and 'Pro 2' in Figure 5.6). The others were *'exclusive arguments'* that were only covered in one case description. 'Pro 3' and 'Pro 4' were only provided to the analyst who was assigned the pro position. 'Pro 5' was only provided to the undecided analyst. The con arguments were distributed contrariwise. In total each analyst had six arguments. The discussion topics were (1) speed limit on highways, (2) non-smoker protection, and (3) nuclear power phase-out. As in the hidden profile experiment the topics were randomly assigned to the condition.

The procedure was similar as in the hidden profile experiment. However, this time fixed time frames were prescribed to make sure that the groups worked equally thorough on their task. The analysts were given four minutes to read the essay and structure their arguments using keywords for all pro and con arguments. Afterwards the moderator led a discussion which took up to eight minutes. If the group did not finish by then, the discussion was stopped by the moderator or experimenter. After that the group was given additional two minutes to reach a conclusion.

Arguments	Analyst 1 (Pro)	Analyst 2 (Undecided)	Analyst 3 (Con)
joint	Pro 1	Pro 1	Pro 1
joint	Pro 2	Pro 2	Pro 2
exclusive	Pro 3		
exclusive	Pro 4		
exclusive		Pro 5	
joint	Con 1	Con 1	Con 1
joint	Con 2	Con 2	Con 2
exclusive		Con 3	
exclusive			Con 4
exclusive			Con 5

Figure 5.5: Number of joint and exclusive arguments to analysts.

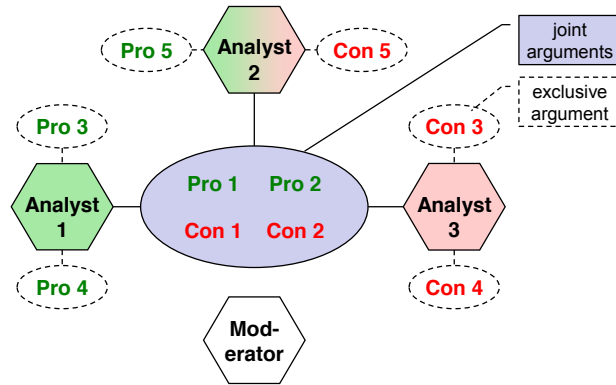


Figure 5.6: Distribution of joint and exclusive arguments to analysts.

The conditions were the same as in the hidden profile experiment: Each group used three tools: pen and paper only, a regular dry-erase whiteboard and Note&Share. Again, the case descriptions were handed out on paper in all conditions. Working a case took up to 14 minutes per tool. The whole session could take up to one hour including introductions in the beginning and prior to each condition.

Measures

There were two sources of quantitative data. First, the sessions were recorded on video and later analyzed. In line with our first study, the number of forwarded arguments was counted as an indicator for the degree of *knowledge sharing*. Second, a *retention* test was conducted after each condition to measure to what degree learners internalized the arguments. Retention tasks have been used to determine how much was learnt in various cases (e.g. by Kirschner et al., 2009). Participants were asked to note down all arguments they remember. The retained arguments can be split into four categories. Joint arguments that were provided to all analysts (cf. Figure 5.6 ‘Pro 1’, ‘Pro 2’, ‘Con 1’ and ‘Con 2’), exclusive arguments that were only provided to oneself (e.g. when ‘Pro 3’ is remembered by ‘Analyst 1’) or only to another analyst (e.g. when ‘Pro 5’ is remembered by ‘Analyst 1’). The fourth category is arguments that were remembered by moderators. As moderators were not provided with an essay containing pro and con arguments, there is only a single category for moderators.

To retrieve a subjective assessment questionnaires were handed out after each condition. They contained the same set of items. Each item consisted of a statement and a 5-point Likert scale from 1 (completely disagree) to 5 (completely agree). The subjective *effort* was measured with the item ‘Forwarding arguments to the group did not cause a great effort’. The item ‘The information was shared effectively in the group’ was used as indicator for *effectiveness*. In addition, the *overview of arguments* was measured with the items ‘I had a good overview of my own arguments’, ‘I had a good overview of my group members’ arguments’, ‘My group members had a good overview of my arguments’ and ‘I did not forget to forward arguments to the group’.

The reliability of the scale was above Cronbach's  $\alpha = .8$  for each condition. Finally, whether the participants were sure that there were no *misunderstandings* was measured with the item 'My arguments were not misunderstood'.

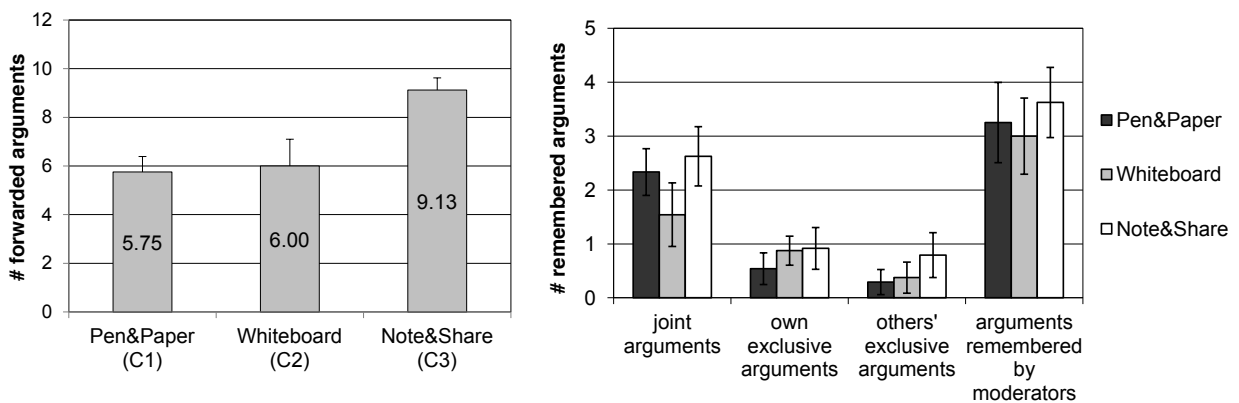
### Statistical analyses

Analysis of variance with repeated measurements were used to reveal overall effects of the tools (within-subject factor). For further pairwise comparisons paired samples t-tests were applied. If other statistical tests were applied (e.g. to compare frequencies), the type of the statistical test is mentioned in the results section.

## 5.3.2 Results

### Knowledge Sharing

Regarding the number of arguments that analysts forwarded to the group, a within participants analysis of variance showed a significant effect for the tool ( $F(2,14)=9.56$ ,  $p=.002$ , see Figure 5.7, left). Paired samples t-test revealed that significantly more arguments were forwarded to the group using Note&Share ( $M=9.13$ ,  $SD=0.99$ ) as compared to pen and paper ( $M=5.75$ ,  $SD=1.28$ ,  $t(7)=-6.34$ ,  $p<.001$ , two-tailed,  $r=.92$ ) or the whiteboard ( $M=6.00$ ,  $SD=2.20$ ,  $t(7)=-3.57$ ,  $p=.01$ , two-tailed,  $r=.80$ ). This number refers to the total number of arguments shared in the group, so the maximum was 10 arguments. In contrast to the first experiment no redundancies occurred because the case descriptions were specifically designed to avoid ambiguity. There were no significant differences between pen and paper and whiteboard ( $t(7) = 0.23$ ,  $p=.824$ , two-tailed,  $r=.09$ ). Consequently, Note&Share seems to have positively affected knowledge sharing.



**Figure 5.7:** Left: Number of arguments that analysts forwarded to the group. Right: Retention test results.

To better understand which feature of Note&Share caused this effect, the video recordings were further analyzed. The transcriptions of two groups are compared in Appendix E. One of the groups performed well across all conditions, the other one performed significantly better when

using Note&Share. The qualitative video analysis indicates that Note&Share streamlines the knowledge sharing process. Analysts presented their arguments and sent keywords to the wall display during the process. This allowed moderators to concentrate on steering the conversation. While rather skilled groups behaved similarly across all conditions, less skilled groups tended to engage in rather unstructured conversations in the whiteboard and pen-and-paper conditions. For instance, moderators would rephrase analysts' statements and further elaborate on them. This made it difficult to identify the arguments (or keywords) in the end and sometimes prevented analysts from presenting all arguments. Consequently, the rephrasing seemed to have a negative effect on the knowledge sharing process.

In the other conditions there was much more variance: Sometimes the moderator would listen to the analyst and write down the keywords when the speaker was finished, at other times the moderator would write down the argument while the analyst spoke. There were also cases in which the moderator listened to several arguments and wrote them down later. Consequently, the chain of events seemed to be less consistent compared to Note&Share.

Furthermore, in cases of ambiguity, skilled moderators would rephrase the statement and ask whether they understood it correctly. This would happen across all conditions, although less frequently in Note&Share because the analyst's keywords provide clarification beyond the verbal explanation. In addition, while in the Note&Share condition it was clear which phrase was discussed (because the keyword was already defined), discussions in the other conditions often resulted in multiple phrases that could be written down in the end. This might explain the higher confidence that no misunderstandings occurred when using Note&Share.

### *Retention test*

Figure 5.7 (right) shows the results of the retention test, which shed light on the cognitive processing of arguments. The first group concerns *joint arguments*, which were given to all analysts. A within participants analysis of variance showed a significant effect for the tool ( $F(2,46)=7.09$ ,  $p=.002$ ). Paired samples t-tests revealed that significantly fewer joint arguments were remembered by analysts when using the whiteboard ( $M=1.54$ ,  $SD=1.18$ ) compared to pen-and-paper, ( $M=2.33$ ,  $SD=.87$ ,  $t(23)=2.53$ ,  $p=.02$ , two-tailed,  $r=.47$ ) or Note&Share ( $M=2.63$ ,  $SD=1.10$ ,  $t(23)=-2.96$ ,  $p=.01$ , two-tailed,  $r=.53$ ). No significant difference was found between pen-and-paper and Note&Share ( $t(23)=-1.57$ ,  $p=.13$ , two-tailed,  $r=.31$ ).

The second group was arguments that were remembered by the analyst who received it exclusively (*own exclusive arguments*). A within participants analysis of variance showed no significant effect for the tool ( $F(2,46)=3.05$ ,  $p=.06$ ). Paired samples t-tests showed that using pen-and-paper significantly less individual arguments were retained ( $M=.54$ ,  $SD=.59$ ) compared to Note&Share ( $M=.92$ ,  $SD=.78$ ,  $t(23)=-2.10$ ,  $p=.05$ , two-tailed,  $d=-.55$ ) or whiteboard ( $M=.88$ ,  $SD=.54$ ,  $t(23)=-2.33$ ,  $p=.03$ , two-tailed,  $d=-.59$ ). There was no significant difference between Note&Share and whiteboard ( $t(23)=.81$ ,  $p=.81$ , two-tailed,  $r=.17$ ).

The third group regarded *exclusive arguments* that were owned by other analysts. A within participants analysis of variance showed a significant effect for the tool ( $F(2,46)=3.73$ ,  $p=.031$ ).

Paired-samples t-tests revealed that significantly more exclusive arguments owned by other analysts were remembered using Note&Share ( $M=.79$ ,  $SD=.83$ ) than using pen-and-paper ( $M=.29$ ,  $SD=.46$ ,  $t(23)=-2.40$ ,  $p=.025$ , two-tailed,  $r=.45$ ). No significant effects were found between whiteboard ( $M=.38$ ,  $SD=.58$ ) and pen-and-paper ( $t(23)=-.53$ ,  $p=.60$ , two-tailed,  $r=.11$ ) as well as between whiteboard and Note&Share ( $t(23)=-1.93$ ,  $p=.07$ , two-tailed,  $r=.37$ ).

The last group was the number of arguments remembered by *moderators*. Here, no distinction was made between shared/unshared or own/others' arguments because the moderators did not have any arguments themselves. Using Note&Share they remembered 3.63 ( $SD=1.30$ ) arguments, using pen and paper 3.25 ( $SD=1.49$ ), and using the whiteboard 3.00 ( $SD=1.41$ ). No significant effects for the tool were found in this group ( $F(2,46)=.40$ ,  $p=.68$ ). Paired-samples t-tests showed that the differences between pen-and-paper and whiteboard was not significant ( $t(23)=.28$ ,  $p=.79$ , two-tailed,  $r=.06$ ), neither were the difference between pen-and-paper and Note&Share ( $t(23)=-.53$ ,  $p=.61$ , two-tailed,  $r=.11$ ) or between whiteboard and Note&Share ( $t(23)=-1.36$ ,  $p=.22$ , two-tailed,  $r=.27$ ).

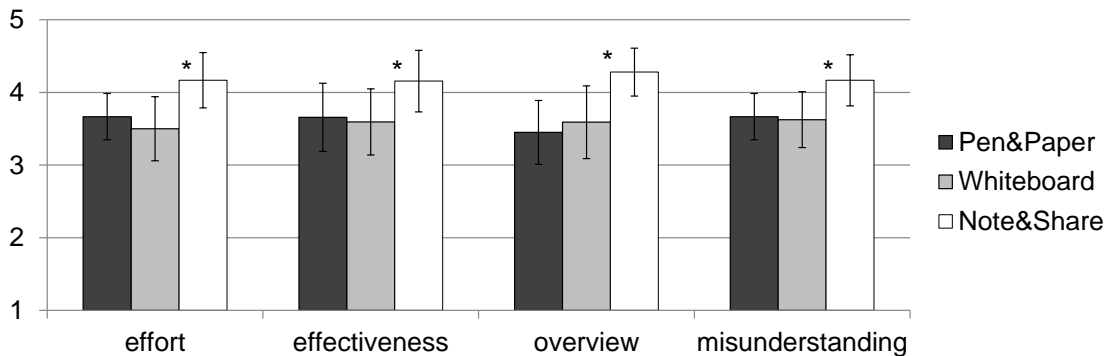
Some of these results are surprising. For instance, there is no obvious explanation why Note&Share and pen-and-paper outperformed the whiteboard with regard to *joint* arguments. We would have expected the worst results in pen-and-paper due to the lack of a shared representation. For arguments owned by *other* analysts this was the case. In addition, we would have expected that Note&Share improves the individual preparation since learners are forced to identify keywords. This explains why the retention of one's *own* arguments was better in Note&Share compared to pen-and-paper but it does not explain the significant difference between whiteboard and pen-and-paper. Despite the fact that some of the effects are inexplicable, it is promising that Note&Share overall performed best.

### *Subjective evaluation*

Figure 5.8 shows the questionnaire results. Regarding the subjective *effort*, a within participants analysis of variance showed a significant effect for the tool ( $F(2,46)=5.64$ ,  $p=.006$ ). Paired samples t-tests showed that the effort was rated significantly higher for Note&Share ( $M=4.17$ ,  $SD=.76$ ) as compared to pen and paper ( $M=3.67$ ,  $SD=.64$ ,  $t(23)=-3.14$ ,  $p=.005$ , two-tailed,  $r=.54$ ) and whiteboard ( $M=3.50$ ,  $SD=.88$ ,  $t(23)=-2.50$ ,  $p=.020$ , two-tailed,  $r=.46$ ). There was no significant effect between pen-and-paper and whiteboard ( $t(23)=.94$ ,  $p=.36$ , two-tailed,  $r=.19$ ). This questionnaire item was only answered by analysts.

Regarding the subjective *effectiveness*, a within participants analysis of variance showed a significant effect for the tool ( $F(2,62)=5.56$ ,  $p=.01$ ). Paired samples t-tests showed that the effectiveness was rated significantly higher for Note&Share ( $M=4.16$ ,  $SD=.85$ ) as compared to pen-and-paper ( $M=3.66$ ,  $SD=.94$ ,  $t(31)=-2.98$ ,  $p=.01$ , two-tailed,  $r=.47$ ) and whiteboard ( $M=3.59$ ,  $SD=.91$ ,  $t(31)=-2.81$ ,  $p=.01$ , two-tailed,  $r=.45$ ). There was no significant effect between pen-and-paper and whiteboard ( $t(31)=.34$ ,  $p=.74$ , two-tailed,  $r=.06$ ).

According to the subjective results, the *overview* of arguments was better using Note&Share compared to pen-and-paper as well as the whiteboard. A within participants analysis of variance



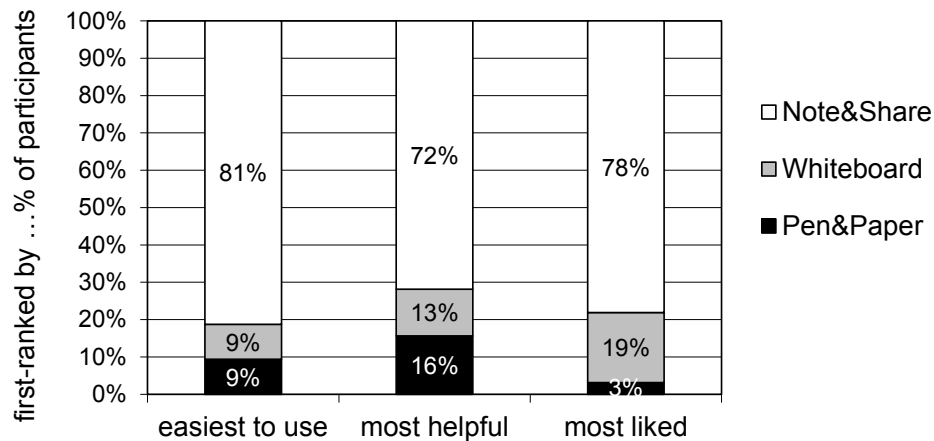
**Figure 5.8:** Questionnaire results on a Likert scale from 1 (completely disagree) to 5 (completely agree).

showed a significant effect for the tool ( $F(2,46)=10.08$ ,  $p<.001$ ). Paired samples t-tests showed that the overview was significantly better rated in the Note&Share condition ( $M=4.28$ ,  $SD=0.66$ ) as compared to pen and paper ( $M=3.45$ ,  $SD=.88$ ,  $t(31)=-5.08$ ,  $p<.001$ , two-tailed,  $r=.67$ ) or the whiteboard ( $M=3.59$ ,  $SD=1.00$ ,  $t(31)=-3.74$ ,  $p=.001$ , two-tailed,  $d=-.95$ ). There were no significant differences between pen-and-paper and whiteboard ( $t(31)=0.23$ ,  $p=.90$ , two-tailed,  $r=.02$ ).

Regarding the subjective *misunderstanding*, a within participants analysis of variance showed a significant effect for the tool ( $F(2,46)=5.29$ ,  $p=.009$ ). Paired samples t-tests showed that the matter of misunderstanding was significantly better rated in the Note&Share condition ( $M=4.17$ ,  $SD=.70$ ) as compared to pen and paper ( $M=3.67$ ,  $SD=.64$ ,  $t(23)=-3.39$ ,  $p=.003$ , two-tailed,  $r=.58$ ) or the whiteboard ( $M=3.63$ ,  $SD=.77$ ,  $t(23)=-2.50$ ,  $p=.020$ , two-tailed,  $r=.56$ ). There were no significant differences between pen-and-paper and whiteboard ( $t(23)=0.225$ ,  $p=.91$ , two-tailed,  $r=.05$ ). This questionnaire item was only answered by analysts.

Finally, participants were asked to rank which tool best fulfilled different *usability* criteria (see Figure 5.9). Note&Share was considered as being the easiest to use (first-ranked by 81%), followed by the whiteboard and pen-and-paper (each 9%). A Friedman-Test for related samples showed a significant difference between the ranks of the three tools ( $N=32$ ,  $\chi^2=31.19$ ,  $df=2$ ,  $p<.001$ , two-tailed). A post hoc Wilcoxon-Test revealed significant differences between Note&Share and pen-and-paper ( $N=32$ ,  $Z=-4.39$ ,  $p<.001$ ), between Note&Share and whiteboard ( $N=32$ ,  $Z=-4.23$ ,  $p<.001$ ). No significant differences were found between pen-and-paper and whiteboard ( $N=32$ ,  $Z=-1.58$ ,  $p=.12$ ). Furthermore, Note&Share was considered the most *helpful* (best-ranked by 72%) followed by pen-and-paper (16%) and the whiteboard (13%). A Friedman-Test for related samples showed a significant difference between the ranks of the three tools ( $N=32$ ,  $\chi^2=15.25$ ,  $df=2$ ,  $p<.001$ , two-tailed). A post hoc Wilcoxon-Test revealed significant differences between Note&Share and pen-and-paper ( $N=32$ ,  $Z=-2.73$ ,  $p=.01$ ) between Note&Share and whiteboard ( $N=32$ ,  $Z=-3.29$ ,  $p=.001$ ). No significant differences were found between pen-and-paper and whiteboard ( $N=32$ ,  $Z=-.36$ ,  $p=.72$ ). Note&Share was also the most *popular* tool that was liked best (first-ranked by 78%). 19% voted for the whiteboard and only 3% for pen-and-paper. A Friedman-Test for related samples showed a significant difference between the ranks of the three tools ( $N=32$ ,  $\chi^2=23.31$ ,  $df=2$ ,  $p<.001$ , two-tailed). A post hoc Wilcoxon-Test re-

vealed significant differences between whiteboard and pen-and-paper ( $N=32$ ,  $Z=-2.02$ ,  $p=.043$ ), between Note&Share and whiteboard ( $N=32$ ,  $Z=-2.71$ ,  $p=.007$ ) as well as between Note&Share and pen-and-paper ( $N=32$ ,  $Z=-4.27$ ,  $p<.001$ ).



**Figure 5.9:** Comparison of tools in terms of usability criteria in study 2.

### 5.3.3 Summary and Discussion

Using Note&Share significantly more arguments were forwarded to the group than using pen and paper or the whiteboard. This finding can be explained by the lower subjective effort and a higher effectiveness in sharing arguments with Note&Share. Furthermore, the overview of arguments was better rated and participants were more confident that no misunderstandings occurred during the information exchange. Thus, Note&Share seems to significantly enhance knowledge sharing.

The qualitative video analysis provides further insights on the cause of these effects. Note&Share seems to facilitate a pattern of usage, which is beneficial for knowledge sharing processes. This pattern stems from the analysts' behavior of sending each keyword during their verbal statement. Furthermore, analysts often started with the first keyword item and worked their way through their keywords list. This explains why the overview of arguments was significantly better rated in the Note&Share condition, which is a likely cause of the higher number of shared arguments. In addition, the video analysis indicates that less skilled groups considerably benefited from Note&Share whereas rather skilled groups behaved similarly across all conditions independent of the order of the conditions.

Due to these findings, we believe that the excitement of using a new technology was not the main cause of the better results in the Note&Share condition. First and most importantly, the qualitative video analyses showed that Note&Share did not lead to more playful interaction but quite to the contrary streamlined the working process. Another indicator is the fact that participants did not rate Note&Share more useful than the whiteboard in the first experiment, but only in the second experiment where an improved version was used. This suggests that the participants critically

reflected on the application. Nonetheless, a long-term study would be needed to rule out a novelty effect.

## 5.4 Related Work

This work combines ideas from different research directions. First, Note&Share builds on previous studies that investigated effects of display environments on collaborative processes, in particular with regard to the role of private, personal and public workspaces. This literature was presented in detail in section 2.3.1 and 2.3.2. Unlike previous studies the presented work on Note&Share demonstrates the potential of combining private and shared workspaces in co-located knowledge sharing processes. In addition, there is related work on knowledge sharing tools as well as brainstorming and group decision support systems, which is presented in the following.

### *Knowledge Sharing Tools*

The term ‘knowledge sharing’ is mainly used in the context of knowledge management in organizations or distributed teams (cf. Alavi and Tiwana, 2002 and Cross et al., 2001). Many tools make use of information technology as means of sharing knowledge in networks such as Intranets. Huysman and Wulf (2006) provide an overview of such tools. Some Web 2.0 sites even allow for knowledge repositories beyond organizations boundaries. For instance, Yahoo Answers (YA) is a large question-and-answer forum, which allows users to submit or answer questions. The platform not only facilitates knowledge sharing - it also enables its users to seek advice or gather opinions (Adamic et al., 2008). Such systems focus on asynchronous computer-mediated communication and therefore are fundamentally different from our co-located scenario.

Another related research field is knowledge visualization. This matter regards all systems that facilitate knowledge sharing: synchronous or asynchronous as well as co-located or distributed. According to Burkhard (2004), knowledge visualization aims to improve the transfer of knowledge between two or more people. It differs from the more established term ‘information visualization’, which supports the exploration of large amounts of data. One form of representation that has been successfully applied in knowledge modelling and sharing is concept maps. Concept maps have shown to facilitate the capturing and sharing of knowledge (Coffey et al., 2002). A practical example is CmapTools (Cañas et al., 2004), a software environment that allows users to represent and share their knowledge in form of concept maps. The maps can be created individually or collaboratively in both synchronous and asynchronous manners. The feature of collaboratively creating concept maps is similar to our approach. However, CmapTools is (as most knowledge sharing tools) designed for computer-mediated communication where the concept map is the means of communication. Therefore, a main challenge in designing such systems is the creation of self-explanatory knowledge repositories. As for Note&Share, the map is merely a visualization of what is discussed verbally.



In both cases, an important feature is the free arrangement of items. Allowing users to create their own organization of representations can support them in creating mental models of the underlying dataset (Isenberg and Carpendale, 2007). Note&Share builds on this literature, in particular regarding the use of mindmaps to visualize knowledge in co-located settings.

### *Brainstorming and Group Decision Support Systems*

There are other forms of computer applications which are related to our work although their primary goal is not knowledge sharing. For instance, there has been intensive research in the fields of brainstorming and group decision support systems. For example, Ackermann and Eden (1994) examined design aspects for group decision support systems, such as location and flexibility of design. They conclude that computerized systems consisting of laptops and portable shared displays could be valuable tools to support group decision processes. Furthermore, studies have shown that computer support can have positive effects on brainstorming results (Gallupe et al., 1991). A recent example for an electronic brainstorming system is GroupMind (Shih et al., 2009), which supports collaborative brainstorming processes in both co-located and distributed settings (also using mind-maps). While this research is highly relevant because it examines computer support in face-to-face collaboration, the primary goals of these systems differ. For instance, group decision support systems help groups to reach consensus, whereas brainstorming systems aim at furthering creativity.

## 5.5 Lessons Learned

Multi-display environments with shared and private displays have the potential to facilitate knowledge sharing. In this chapter, Note&Share was presented as one example of a multi-display environment that was specifically designed for this purpose. It consists of several laptops, which provide private workspaces for the group members, and a smartboard that serves as shared workspace. Two studies showed that Note&Share facilitates knowledge sharing processes better than the commonly used whiteboard or pen-and-paper. The key benefit seems to be the combination of (1) private workspaces, (2) a shared workspace and (3) an interaction technique that allows to transfer items between the workspaces with a simple drag-and-drop gesture. This combination afforded a behavior pattern that positively affected the knowledge sharing process: First, group members create a list of keywords to summarize their individual knowledge prior to the discussion. This is done in private workspaces so that learners cannot see their peer's keywords and are not influenced by them. During the discussion the same keywords are sent to the smartboard one by one and each keyword is verbally explained. Compared to the rather varying behavior in the whiteboard or pen-and-paper conditions, there was little room for misunderstandings. Furthermore, group members tended to share all keywords in their list instead of only parts of the information.

Due to its simplicity, Note&Share might also be useful for other collaborative tasks. For instance, brainstorming sessions could highly benefit from the combination of private and shared

workspaces: First, group members individually brainstorm for ideas and create keywords for each of them. In the second phase the keywords are shared and discussed with the group. This bisection seems particularly useful for brainstorming sessions because it has frequently been found that groups create fewer ideas when brainstorming together than when brainstorming individually (Diehl and Stroebe, 1987). With a tool like Note&Share the underlying effects such as free riding and production blocking could be avoided. For similar reasons decision making processes could benefit from Note&Share.

Furthermore, the tool could also be used in classrooms. Preliminary results from a pilot study in two seminar sessions with up to eleven participants were promising. After a short demonstration of the tool, the students were provided with laptops and a text on the seminar topic. The text was silently read bit by bit. For each paragraph the students created keywords which summarized the content. During the discussion students sent the keywords to the smartboard and the teacher arranged them in a concept map. Again, such classroom situations could benefit from the two-fold workspaces, because the students first have to individually deal with the text in order to be prepared to contribute to the discussion. Both students and teacher found the tool very helpful and easy to use.

Note&Share uses concepts, which have been established in previous work, for example the use of mindmaps to visualize and structure knowledge. Also, the technologies and user interface concepts are adopted from related work (e.g. the Gateway Interaction Technique). Our main contribution is to demonstrate the potential of combining individual and group workspaces to further collaborative learning through implicit guidance. While this implicit guidance can be valuable, it also has limitations. For instance, it is generally not available in single-display applications and it only supports certain types of guidance such as guiding learners through a sequence of activities and assigning tasks to different group members. Due to these limitations the following chapters explore *explicit* guidance mechanisms (to be used instead of or on top of implicit guidance).

# IV

## EXPLICIT GUIDANCE MECHANISMS



# Chapter 6

---

## Guiding Widgets and GUI Concepts for Shareable User Interfaces

The last section discussed the possibility of implicit guidance by using different displays in different learning phases. However, it is still important to look into ways of guidance on single shared displays for three reasons. First, there are still many scenarios in which a single shared display is sufficient or even beneficial. This particularly applies to tabletop applications as they can be used continuously for longer periods. For instance, *MatchingTable* and *PoetryTable* (which were presented in section 2.2.1) depend on the availability of one shared workspace that is continuously used throughout the entire learning process. Second, even if different displays are used in different phases, there may be need for further guidance *within* a phase. Third, the effect of implicit guidance could be strengthened by adding explicit means of guidance. This chapter therefore pursues the question how learners can be guided through collaboration scripts using interactive graphical widgets.

**Chapter Question:**

How can learners be guided through collaboration scripts explicitly using interactive graphical widgets?

In terms of collaboration script components and mechanisms (see section 2.1.1) there are two guidance aspects, which are particularly suitable to be integrated in the learning application: guiding learners through a sequence of learning activities or tasks and distributing different roles to the group members. In the following we present two tabletop prototypes that pursue these goals. The first prototype is ‘*Task Tokens*’, which guide a group of learners through a sequence

of tasks (section 6.1). A study showed that the Task Tokens work very well in terms of assigning tasks in a specific order. However, from an educational perspective, it would be desirable to assign different roles for each of the tasks. Therefore, a second prototype ‘Ludo’ was built, which is presented in section 6.2. While the Task Tokens are restricted to tabletop scenarios with around-the-table seating arrangements, Ludo was built for two different display environments: as tabletop application and as application consisting of laptops and wall display. Finally, section 6.3 illustrates how Guiding Widgets could be integrated in a development process with automatic code generation (such as facilitated by the XSS Framework).

## 6.1 Task Assignment

Research on computer-supported collaborative learning has emphasized that collaboration needs to be structured to lead to optimal learning outcomes (Kobbe et al., 2007). One way of structuring collaborative learning is by specifying a sequence of tasks that learners should engage in. As an example, the learning strategy ‘Reciprocal Teaching’ (Palincsar and Brown, 1984) defines a sequence of learning activities, which has shown to improve text comprehension: clarifying, summarizing, questioning, and predicting. In a study Reciprocal Teaching was compared to traditional classroom practice. Results showed a significant improvement in the quality of the summaries and questions. Furthermore, Reciprocal Teaching led to greater gains on comprehension tests and maintenance over time. Without guidance, learners rarely engage in such activities on their own accord, and if they do, the same task is often done by the same learner (cf. Kobbe et al., 2007). To overcome these problems we propose a GUI concept for task assignment on tabletop displays. The goal is to guide a group of learners through a sequence of learning activities (or tasks) and to distribute tasks among the learners in a balanced way. The central GUI elements are graphical representations of tasks (‘Task Tokens’) that can be moved across a tabletop display to assign the task to specific learners.

The general concept of Task Tokens as a mechanism of guiding learners through a sequence of tasks was evaluated in a user study. While it was likely that the Task Tokens cause an increased adherence to Reciprocal Teaching (as compared to learning sessions without scaffold), an interesting question was how strict the guidance should be. On the one hand, it is - from an educational perspective - desirable to direct learners according to a chosen learning strategy. On the other hand Dillenbourg (2002) pointed out that the ‘fun and richness of group interactions’ can be endangered by applying overly strict guidance mechanisms. To investigate this issue, we implemented three assignment modes for Task Tokens: Automated assignment by the tabletop application, assignment by one learner who adopts the role of a moderator and a free assignment mode in which learners have to volunteer. In the user study these three levels of guidance were compared to find a good balance between guiding the learners enough to trigger effective learning activities and creating a too constrained learning environment <sup>1</sup>.

---

<sup>1</sup> The practical part of this work was conducted by Wolfgang Matzke (2010).

### 6.1.1 Task Tokens

#### *User Interface Concept*

As usual, the first decision in the user interface design process was the choice of display (environment). A tabletop scenario in which learners are seated around the table was chosen because this is considered to be beneficial for collaborative learning (Rick and Rogers, 2008). For instance, tabletop displays have shown to facilitate partner gaze and natural face-to-face communication (Inkpen et al., 2005) as well as balanced participation (Rogers and Lindley, 2004, cf. section 2.3.2).

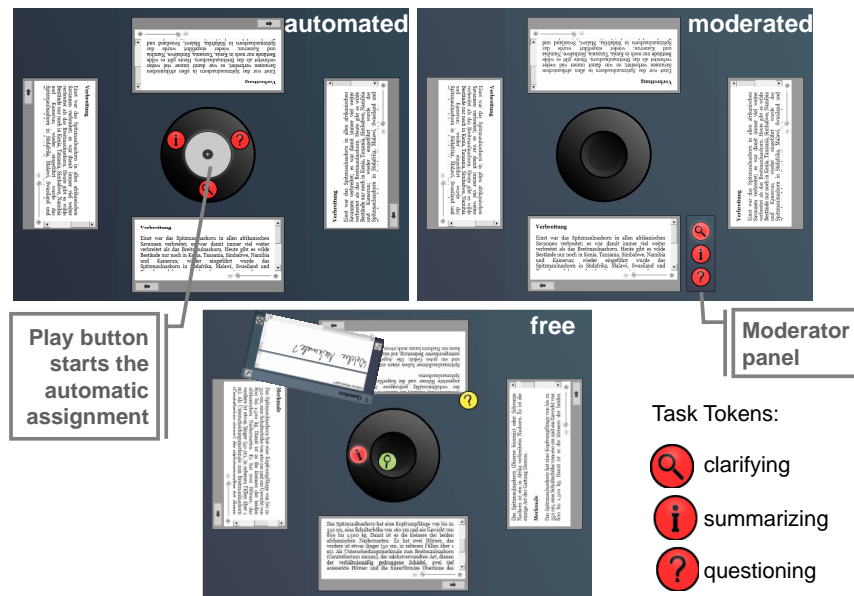
The ‘Task Tokens’ are graphical representations of tasks. For each task type there is one Task Token, which is characterized by a specific icon. In the beginning all Task Tokens are located in a central depot. To assign a task the according Task Token is moved towards one specific learner. A tap on the token opens an editing window that provides a workspace in which the task can be accomplished. When the task is finished, the learner closes the editing window and the token changes its color from initially red to now yellow. Then the token is moved back to the central depot and the color changes to green. Once all tasks are accomplished and thus a complete cycle is finished, a new set of Task Tokens is shown for the next learning phase.

This GUI concept was chosen for several reasons. The depot in the center provides an overview of the finished and remaining tasks and thereby serves as a central visualization of the task sequencing and assignment process. The learning material as well as the actual editing is, however, done in the proximity of one learner, which is an implicit representation of the task assignment. We chose virtual tokens instead of physical tokens. This allowed us to implement an automatic assignment mode in which Task Tokens automatically move on the table towards a specific learner.

In this case the learning tasks and therefore also the corresponding token types were clarifying, summarizing and questioning, which were adopted from the Reciprocal Teaching learning strategy (Palincsar and Brown, 1984). However, by adapting the Task Token symbols as well as the content of the editing window and learning material, the GUI concept can be applied to various learning scenarios, which involve small groups who collaboratively elaborate on any kind of learning material. Typical use cases are groups of learners who study for an exam or breakout sessions (e.g. during a seminar), in which learners are handed out learning material that they need to read and elaborate on in groups.

#### *Assignment Modes*

The application provides three modes of token assignment. In the (1) automated mode the tokens initially lie in the center but are automatically moved towards one group member when the previous task is finished (see Figure 6.1, left). Groups can overrule the initial assignment by passing on the token to another group member. In this mode a ‘Play’-button is needed to initiate the automatic assignment. In the (2) moderated mode there is an additional panel for the moderator that contains the Task Tokens in the beginning (see Figure 6.1, right). The moderator can choose a token and pass it to a team member. The token representing the next task is highlighted



**Figure 6.1:** Task Tokens in the automated (left), moderated (right) and free mode (bottom). The screenshot at the bottom also contains an editing window as well as finished tasks (green), current tasks (yellow) and left tasks (red).

by a pulsing glow, however, the proposed order can be overruled. The role of the moderator is assigned by the application and rotates after each round. Finally, in the (3) free mode the tokens remain in the central depot until learners voluntarily take them. As in the moderated mode the task sequence is indicated by highlighting.

### 6.1.2 Method

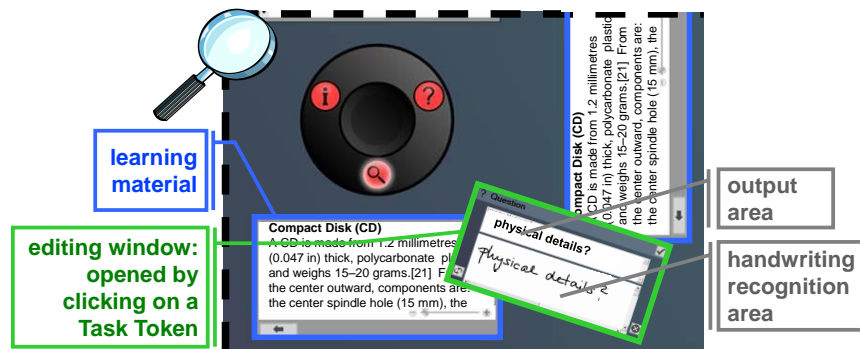
A user study was conducted to investigate the following research questions:

- RQ 1: To what extent does the assignment mode affect the adherence to the task sequence?
- RQ 2: To what extent does the assignment mode affect the balance of role distribution?
- RQ 3: To what extent does the assignment mode affect the subjective assessment of the applied guidance?

#### *Participants and Design*

The goal of the study was to find out which assignment mode provides the most suitable guidance in comprehension training sessions without tutor. Therefore, the three modes were compared to each other as three different conditions in a within-subject design. Five groups of three (N=15) each used all three modes in a counterbalanced order. The age ranged from 24 to 51 (30 on





**Figure 6.2:** Personal workspaces containing (1) learning material and (2) an editing window, which is used to enter the results (using handwriting recognition).

average). Seven participants were female. Except for one participant all group members knew each other as they were recruited in groups during seminars.

In each of the conditions, the group was given different text passages which they should read and then apply the learning tasks to. For each text passage the group worked on three tasks, each represented by one Task Token. At first, clarifying questions should be discussed regarding misleading or complex sections of the text. Second, the content of the text was to be summarized, and third, examining questions had to be asked and answered by the group. The solution to each task (e.g. summary or questions and answers) was entered in separate editing windows (Figure 6.2). For the purpose of our study handwriting recognition was used for text input.

### *Learning Task and Procedure*

In the beginning, there was a brief introduction including a demonstration of how the tokens are used, how text can be entered and what the differences in the three modes are. The participants were instructed to outline the solution to each task using the editing window. It was not specified how the group should collaborate verbally.

The order of the text passages remained constant. That way the allocation of condition and text passages was automatically counterbalanced. This also allowed us to take a continuous text instead of several different ones, which would have been an additional independent variable and less realistic. Consequently, the text had to be divided into three parts (for each condition) and each of these parts was again divided into three parts to allow each participant to take each role once per condition.

After each condition the same questionnaire was handed out. In the end there was an additional questionnaire for demographic information and comparative questions.

### *Measures*

It had already been shown that Reciprocal Teaching improves comprehension (Palincsar and Brown, 1984). Therefore, the focus of this study was not to measure learning outcomes. As-

suming that the learning outcome increases when the learners engage in the proposed sequence of tasks, the goal was to investigate whether Task Tokens guide the learners accordingly. In terms of objective data, we recorded the sessions and analyzed the videos to find out whether the suggested task sequence was adhered to or overruled (*adherence to task sequence*). Second, we measured the *balance of role distribution*.

In addition, the subjective opinion was examined. After each condition a comparative questionnaire was handed out, each containing the same sets of 5-point Likert scale items. One set of items related to the *general understanding* of the Task Tokens, i.e. whether the functionality and instructions were well understood. Another item set gained insights on the participants' subjective opinions on the three modes in particular regarding the level of guidance (*perceived guidance vs. constraint*). In the end, a final questionnaire was handed out to determine the *popularity* of the assignment modes.

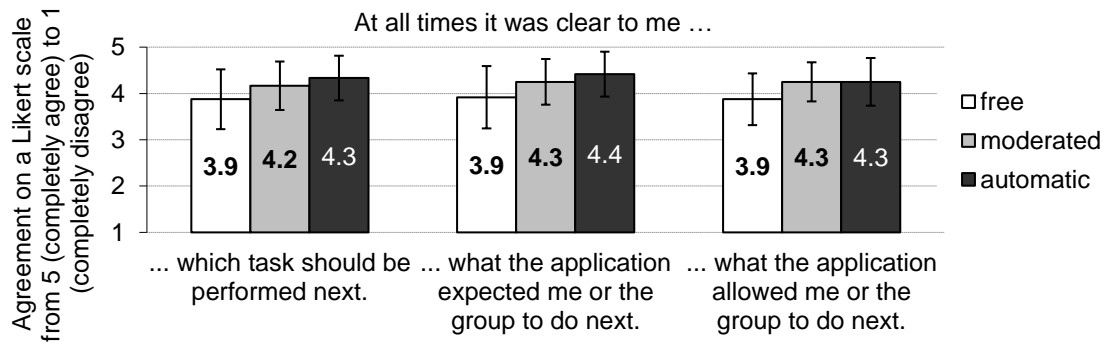
### 6.1.3 Results

#### *Adherence to Task Sequence*

While the assignment of tasks to learners varied across the conditions, the task sequence was specified in all conditions (either by highlighting the next token or by moving the next token to a learner). The video analysis showed that the predetermined sequence was significantly more often adhered to when the tokens were automatically passed to a person (100%) as opposed to tokens that call attention through highlighting (which was the case in the free (66.2%, SD=6.3) and the moderated mode (78.2%, SD=7.7). An ANOVA with repeated measures showed a significant effect for the mode ( $F(2,28)=10.52$ ;  $p<.05$ ; partial  $\chi^2=.43$ ). A Bonferroni post hoc test revealed that the adherence was significantly higher in the automated mode compared to the moderated ( $p=.041$ ) and the free mode ( $p=.000$ ). Because the highlighting was by far too clear to be easily overlooked, our assumption is that the highlighting is rather taken as a recommendation whereas moving tokens from the center to one of the learners was interpreted as a command. Although this does not seem surprising in hindsight, it was not our intention in the first place. Our goal was to vary the assignment of tasks to learners, not the order of tasks. It seems that by changing the way tasks are assigned to learners, the urgency of other guidance aspects is inevitably changed as well.

#### *Balance of Role Distribution*

Across all conditions the task distribution among team members was fairly balanced. On average 86.7% of the group members contributed to all three tasks in a cycle. Interestingly, we observed that the tokens created an increased awareness regarding how often each participant was assigned a token. In the free mode where the tokens needed to be taken from the center, people would make suggestions like 'ok, now it's my/your turn', by which they would refer to the person that had not done this task for the longest time. On the contrary, people would not prompt their peers to make more contributions with regards to content when that person was very reticent in terms of the



**Figure 6.3:** Understanding of functionality and instructions.

verbal output. It seems that by introducing a virtual task object that can be moved to the person responsible, the group becomes more aware of a balance of distribution regarding this task.

### *General Understanding*

The idea of using Task Tokens, which represent specific learning activities, was easy to understand. Figure 6.3 shows the post-questionnaire results regarding the functionality and instructions of the Task Tokens. Across all modes the participants agreed that it was at all times clear what the application allowed and expected them to do next.

### *Perceived guidance vs. constraint*

In the post-questionnaire the participants were also asked whether the modes met their expectation regarding the appropriate level of guidance. When asked whether each mode kept the balance between enough guidance and not too much constraint, there were hardly any differences between the modes. On a scale from 1 (too restrictive) to 5 (not enough guidance) the automated mode was closest to the ideal value of three (mean=2.9, SD=1.0). The moderated mode (mean=3.3, SD=0.9) and the free mode (mean=3.3, SD=1.0) were also considered to have a quite good balance with a slight tendency to a lack of guidance. Thus, the automated mode did not seem to be too restrictive.

### *Popularity*

Regarding the popularity of the different modes we hypothesized that the automatic mode is liked best because no social pressure with regard to choosing or assigning roles was involved. We also expected that choosing a role (in the free mode) is preferred over assigning roles to others (in the moderator mode). In the post-questionnaire the participants were asked to rank the modes according to their preference. As expected, 46.7% favored the automatic mode, 33.3% the free mode and 20% the moderated mode.

### 6.1.4 Summary and Discussion

The study showed that Task Tokens can guide learners through a sequence of learning activities. Three assignments modes were compared in the study. The results show that the automated mode not only performed best in terms of adherence to the suggested task sequence, it was also not considered to be too restrictive. The moderated mode performed second-best in terms of adherence to the task sequence but was liked least by the participants. This clearly speaks in favor of an automatic task distribution. Furthermore, we observed that using Task Tokens as an explicit representation of the tasks as well as the task assignment process created an increased awareness of the task distribution within the group. Therefore, they seem to support the group in keeping a balanced task distribution in which every person takes each learning task equally often.

While guiding learners through a sequence of learning activities is an important part of guidance, collaboration scripts also assign roles within the group. We therefore extended this GUI concept by assigning complementary roles such as teacher and student who asks/answers questions (see section 6.2).

## 6.2 Role Assignment

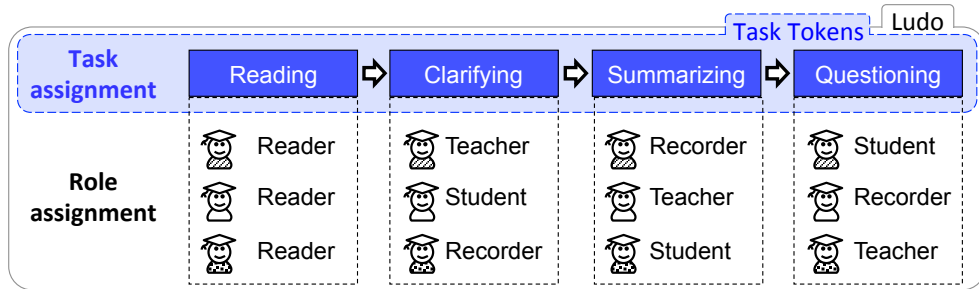
This section presents an extended concept, which not only provides guidance on tasks but also assigns different roles to different group members for each of these tasks. As this extended concept contains more functionality, we chose an iterative design process to ensure that it is easy to understand and to use. First, we implemented two concepts ‘Spin the Bottle’ and ‘Draw a Card’ as paper prototypes (section 6.2.1) and compared them in a study (section 6.2.2). Based on the feedback, the concept was then adapted to ‘Ludo’, which was implemented as functional prototype (section 6.2.3). In a second study, three different Ludo versions (‘Strict Ludo’, ‘Free Ludo’ and ‘Distributed Ludo’) were evaluated with regard to their effects on role distribution, awareness of the learning partners, group interactions and individual work <sup>2</sup>.

### 6.2.1 First Concepts: ‘Spin the Bottle’ and ‘Draw a Card’

The goal was to extend the guidance provided by the Task Tokens by assigning roles for each task. Thus, the original task sequence (reading, clarifying, summarizing and questioning) is maintained (see Figure 6.4). In the reading phase all learners have the role of a ‘reader’. For the other tasks three different roles are assigned to different group members: (1) recorder, (2) teacher and (3) student. The *recorder* writes a protocol of every noteworthy information that is verbalized in the collaborative process (independent of the task). The notes could be in the form of keywords or sentences. The responsibilities of *teacher* and *student* depend on the task. In the clarifying phase all group members can ask and answer clarifying questions. In the summarizing phase the

---

<sup>2</sup> The practical part of this work was conducted by Maraike Stuffer (2011) and Sarah Torma (2011).



**Figure 6.4:** Functionality of the Task Tokens (task assignment) vs. Ludo (task and role assignment).

student summarizes the text and the teacher subsequently gives feedback on the summary (similar to the M.U.R.D.E.R. script presented in section 3.2.1). In the questioning phase the teacher asks questions, which the student has to answer. The teacher then assesses the correctness of the student's answer. If the student is not able to answer the question, the teacher provides the correct answer.

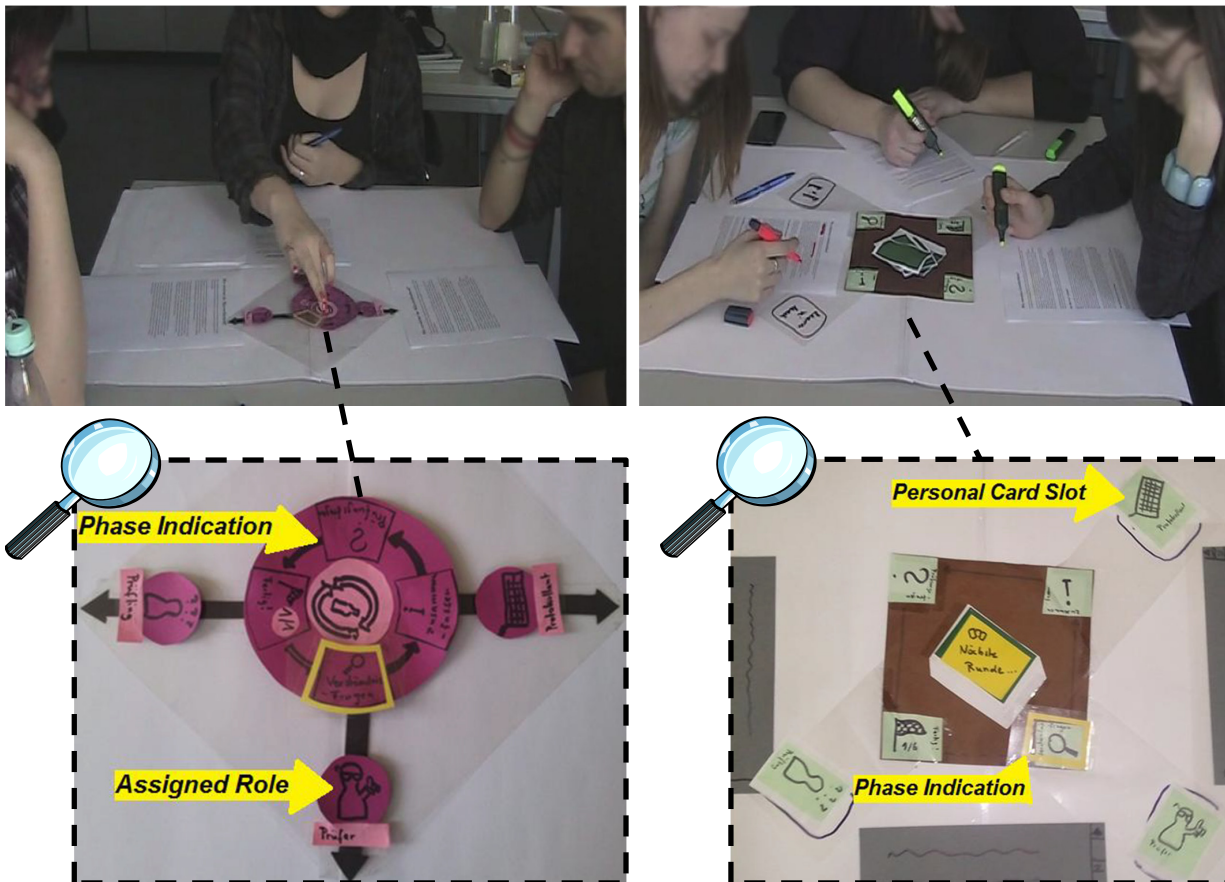
To facilitate task *and* role assignment, the user interface obviously needs to be more complex compared to the Task Tokens, which only conveyed tasks. In this sense, an important underlying question is how much guidance can be carried out by an application without overloading its user interface. Therefore, it was particularly important to carefully design the user interface and get early feedback on its usability. We therefore created two user interface concepts, which we implemented as paper prototypes. Both concepts metaphorically relate to popular games. The first concept is 'Spin the Bottle' where roles are assigned by rotating arrows. The second concept is called 'Draw a Card'. What both concepts have in common is a fixed, central visualization of the task sequence. This replaces the central depot of the Task Tokens. In contrast to the Task Tokens, which are moved out of the central depot to open an editing window, the position of the task representations is fixed. The reason is that the editing window is now attached to the *role* of the recorder instead of the according *task*.

### *Spin the Bottle*

The first prototype is Spin the Bottle. In the center of the table there is a guiding widget which assigns both tasks and roles (see Figure 6.5, bottom left). The task sequence is shown in the inner circle. The current phase is highlighted by a yellow frame. Due to the results of the Task Token study, in which the automatic task assignment performed best, the order of the tasks is fixed. The only way of interacting with the circular element is switching to the next task, which is done by tapping on the circle's center. The arrows, which originate from the center, randomly assign the roles of teacher, student and examiner to the group members.

### *Draw a Card*

The second prototype is 'Draw a Card'. Again, there is a central widget, which shows the task sequence (see Figure 6.5, bottom right). In addition, there is a card stack. Instead of rotating



**Figure 6.5:** Spin the Bottle (left) and Draw a Card (right) in the paper prototype study.

arrows, each learner has to drag a card from the central card stack and drag it into her personal card slot. Consequently, a major difference is that the roles are distributed one after another compared to the simultaneous role assignment in Spin the Bottle.

## 6.2.2 Paper Prototype Study

Spin the Bottle and Draw a Card were implemented as paper prototypes and compared in a user study. The goal was to get qualitative feedback on the quality of the metaphors, the acceptance of the roles as well as the comprehensibility of the concepts. Five groups of three ( $N=15$ ) with an average age of 23 participated in the study. A within-subject design with counterbalanced order was used to rule out learning effects. The experimenter explained the general idea of reciprocal teaching, the tasks and roles as well as the functionality of the prototype. After each condition, questionnaires were filled out. In the end the group discussed benefits and drawbacks of both prototypes and suggestions for improvement.

While the participants mentioned many pros and cons of both approaches, there was no clear trend. For example, some participants argued that drawing cards one after another creates a higher awareness of the partners' roles. Others said they only looked at their own card but were aware of all arrows in Spin the Bottle. Furthermore, the usability as well as the quality of metaphor was rated fairly equal. Consequently, the amount of participants who overall favored Spin the Bottle was about the same as those preferring Draw a Card. In addition, we noticed that many participants lifted the card from the central stack, although they were instructed how to interact with the paper prototype. Finally, many participants suggested to change the name of roles to better refer to the learning activities. Instead of using the names 'teacher' and 'student' across phases two through four (as suggested in section 6.2.1, cf. Figure 6.4), it would be more helpful to use the role name 'questioner' in the clarifying phase as well as 'summarizer' and 'listener' in the summarizing phase. As this general suggestion of improvement is not restricted to any particular concept, the role names were revised accordingly (see Figure 6.6).

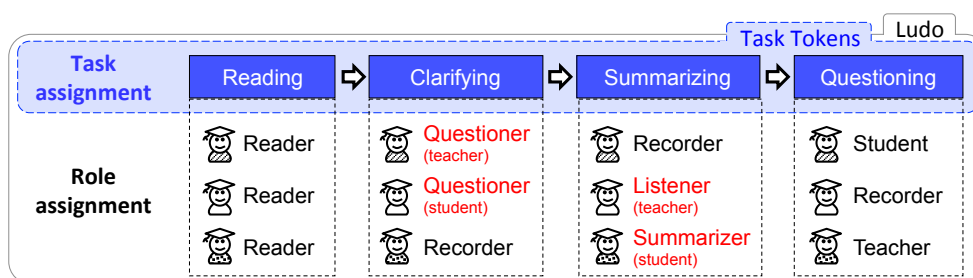


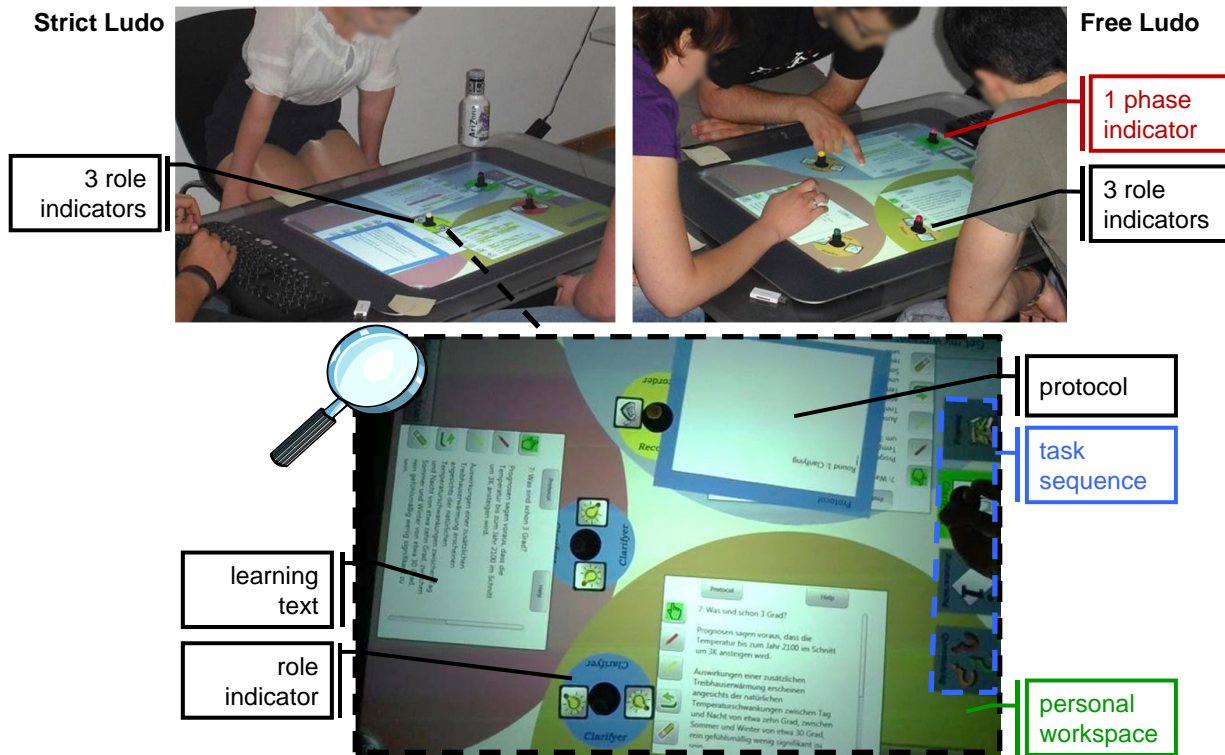
Figure 6.6: Revised role names (original names in brackets).

### 6.2.3 Conceptual Redesign: Ludo

The findings of the paper prototype study were used to revise the concept. Overall both concepts seemed to perform equally well. However, Draw a Card left more room for different assignment modes. For instance, you could allow learners to choose when to switch cards (each representing a specific role) or enforce an automatic role assignment. Different assignment modes could also be implemented with Spin the Bottle but would be less intuitive. We therefore decided to use a modified version of Draw a Card. As the cards were often picked up during the paper prototype study, we chose tangible objects instead of virtual cards. Instead of paper cards we used gaming pieces, such as the ones used in the traditional 'Ludo' game<sup>3</sup> because they are easier to grasp. Since the name 'Draw a Card' no longer fitted to the tangible gaming pieces, we renamed the functional prototype 'Ludo'.

The functional Ludo prototype was implemented on a Microsoft Surface. To better use the limited space of the Microsoft Surface the task sequence controls were moved from the center to one edge (see Figure 6.7). We implemented two different ways of using the tangibles for role and

<sup>3</sup> in German 'Mensch Ärgere dich nicht!'



**Figure 6.7:** Ludo: Phase transition and role distribution with tangibles.

task assignment. One facilitates a strict assignment of roles (*Strict Ludo*), the other one gives learners more freedom to choose when to switch roles and with whom (*Free Ludo*).

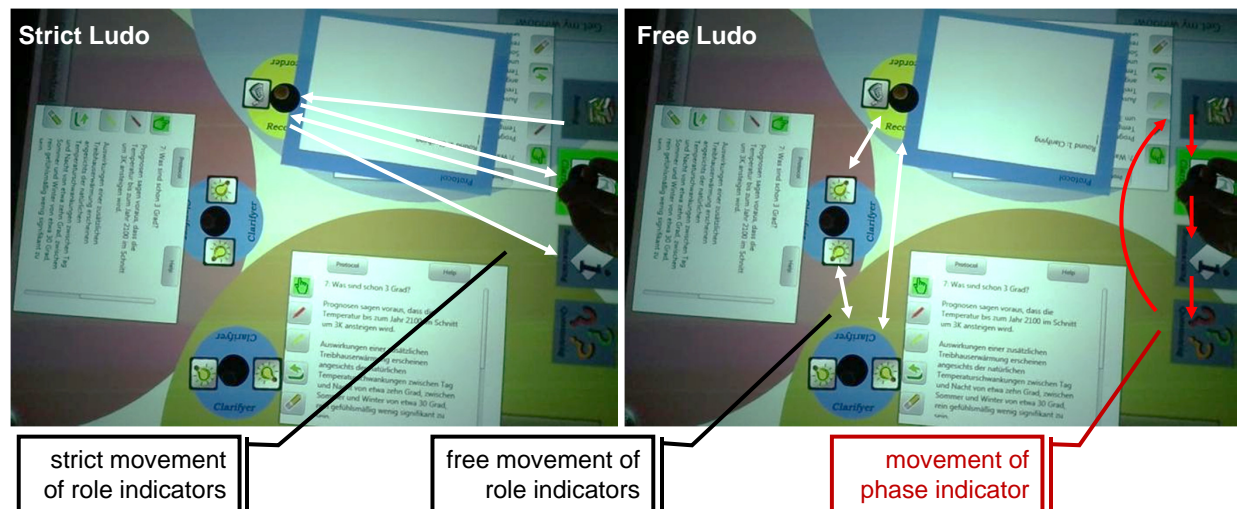
### *Strict Ludo*

In *Strict Ludo* each learner has an own dedicated gaming piece. In the beginning all learners take their gaming pieces from the first task field (reading) and place it within their personal workspace. The application detects the visual marker on the bottom of the tangible gaming piece, automatically assigns a role to the learner and shows the role in a circle, which is displayed around the gaming piece (cf. ‘role indicator’ in Figure 6.7). The task description is linked to the role and can be looked up by tapping on the circle. Once a learner is finished, he places his gaming piece on the next task sequence field. When all gaming pieces are reunited the gaming pieces are moved back to the personal workspaces and new roles are assigned automatically. The movement pattern is shown in Figure 6.8 on the left.

### *Free Ludo*

In *Free Ludo*, there is an additional gaming piece for the task distribution. It is moved from one field in the task sequence to the next. Consequently, the group members need to agree on the time to switch to the next task. The three tangibles that assign the roles do not belong to a fixed person (as in *Strict Ludo*). Instead, each tangible represents a fixed role. Roles are switched by





**Figure 6.8:** Movements of the Ludo tangibles.

moving the tangibles to other personal workspaces. That way, learners are free to decide when the roles are switched and to whom the roles are assigned.

### *Distributed Ludo*

In addition to the two tabletop versions, the concept was transferred to a distributed setup consisting of laptops and walls (*'Distributed Ludo'*). Some benefits of this combination have already been shown in studies on the ArgueWall (section 4.2) and Note&Share (section 5.1). A distributed setup could have several advantages to the tabletop in this scenario. First, learners could benefit from having a private display, which allows the learners to use their own working style (Inkpen et al., 2005). Moreover, different elements (e.g. protocol window and text) overlap on the tabletop, which can hinder the workflow (Everitt et al., 2006). This problem can easily be circumvented on the laptop. Another major advantage is the text orientation problem on tabletops (cf. Everitt et al., 2006), which does not occur in the distributed setup. Finally, due to the indirect input devices, working on laptops has ergonomic advantages compared to direct touch on tabletops, where usually not all areas can be reached comfortably (Ha et al., 2006). On the other hand, the awareness of the learning partners could be higher on the tabletop where all activity is visible to the entire group.

For the development of Distributed Ludo the major goal was to create an application that is as similar to the tabletop application as possible. Consequently, most elements and their representations (e.g. the icons that represent tasks and roles) remained. The task sequence is shown on the wall display (see Figure 6.9, top right). Below there are visualizations of the different group members and their roles. Whether they are finished with their current task is indicated with a green check mark icon. On the left side of the wall display the protocol is shown.

The laptops contain the learning text (Figure 6.10, left). Next to it is an editable text field for the protocol, which is synchronized with the protocol on the wall display. The protocol area is

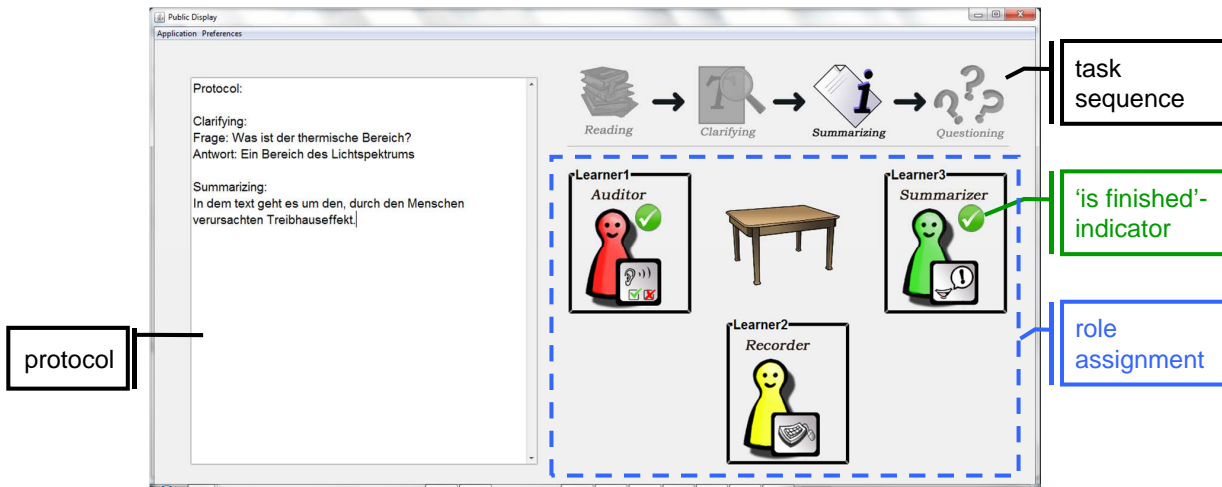


Figure 6.9: Screenshot of the shared wall display.

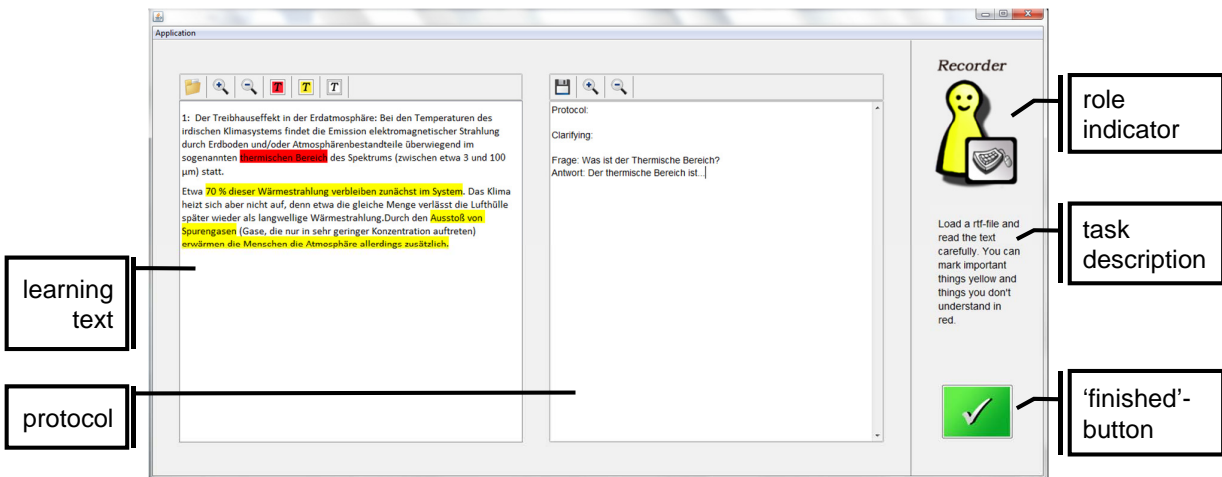


Figure 6.10: Screenshot of the recorder's private display (laptop).

only shown on the recorder's laptop. On the right there is a representation of the laptop owner's role and a 'finish' button. When the mouse is placed over the role indicator a task description is presented below. The role distribution is done automatically once all learners have pressed the 'finish' button.

### 6.2.4 Method

The three Ludo versions (Strict, Free and Distributed Ludo) were compared in a user study. They differ in two dimensions (cf. Figure 6.11). The first dimension is the display environment: Strict Ludo and Free Ludo are both tabletop applications, while Distributed Ludo uses laptops and a wall display. The second dimension is the strictness of guidance: Strict Ludo and Distributed

		Display Environment	
		Tabletop	Laptops & Wall
Strictness of Guidance	Strict	Strict Ludo	Distributed Ludo
	Free	Free Ludo	

**Figure 6.11:** Dimensions in which the three Ludo versions vary.

Ludo apply a strict guidance, whereas Free Ludo leaves freedom with regard to role switching. One combination of the two dimensions is missing. A distributed application with freedom regarding the role distribution was omitted for two reasons. First, we considered it to be the combination which makes the least sense because the choice of roles would have to be implemented as additional feature on the laptops. Consequently, individuals would choose roles in their private workspace which does not support the feeling of mutual agreement as the shared tabletop workspace. Second, a fourth condition could not have been included in the study anyway due to time constraints.

The study examined four research questions. RQ1 investigates effects caused by the strictness of role distribution and therefore compares Strict and Free Ludo. RQ2 examines effects of strictness (Strict vs. Free Ludo) as well as the display setting (Strict vs. Distributed Ludo). RQ3 and RQ4 only focus on the display setting.

*RQ1: To what extent does the strictness of role assignment affect the role distribution?* The strictness of guidance is likely to change the role distribution within the group. Strict Ludo and Distributed Ludo automatically assign roles to learners in a balanced way. Free Ludo, on the other hand, lets the learners decide when to switch roles and with whom. Therefore, we expected Strict and Distributed Ludo to facilitate a more balanced role distribution than Free Ludo. Furthermore, in accord to the findings of the Task Token study, where the strictest guidance was also liked best, we expected that the satisfaction with the role assignment is higher in Strict and Distributed Ludo.

*RQ2: To what extent do the display setting and the strictness of role distribution affect the awareness of the learning partners?* This regards several aspects. The first one is the *awareness of the learner partners' roles*, which we expected to be higher in the two tabletop versions. The reason is that the tabletop display is always the center of attention whereas in Distributed Ludo learners have to divide their attention between the laptops and the wall display (where the roles of all group members are shown). The second aspect is the *awareness of the learning partners' progress*. Again, we expected this awareness to be higher in both tabletop versions where all activities are visible to the group. On the wall, the only indicators in Distributed Ludo are (1) the green check mark symbol and (2) whether one is still interacting with the laptop (e.g. typing). In addition, we expected the awareness of the learning partners' progress to be higher in Strict Ludo compared to Free Ludo because in Strict Ludo learners must move their tangibles to the next task field which generates an additional awareness.

*RQ3: To what extent does the display setting affect the group interaction?* The display setup can affect the interaction between the group members. We expected that due to the laptops' private workspaces the subjective concentration on the team members is lower in Distributed Ludo. Respectively, the distraction from the group interaction caused by the application is expected to be higher in Distributed Ludo.

*RQ4: To what extent does the display setting affect the individual work?* Whether each learner is provided with a private workspace (laptops) or whether all activities are visible to the group members can affect the individual work. We expected that learners work at their own pace and feel less observed by their learning partners when using laptops compared to the two tabletop conditions.

### *Participants and Design*

To compare the three Ludo versions, a within-subject design was chosen. 24 people participated in groups of three in a counterbalanced order. Their age ranged from 17 to 28 years. 15 participants were female. Three of them were high school students, the others were enrolled in college or at university. The goal was to recruit groups that did not know each other, however, it occurred that participants knew each other casually.

### *Learning Task and Procedure*

The procedure was very similar to the Task Token study (cf. section 6.1.2). In addition to instructions on the learning tasks, the roles were explained. In the beginning of each condition it was also demonstrated how the group could move on to the next task and how to switch roles.

This time the learning text was on scientific aspects of the greenhouse effect. The text was again divided in nine segments, three for each condition. Consequently, the group was asked to perform three complete cycles of the tasks (reading, clarifying, summarizing, questioning). This was done to allow each group member to take each role once for each task. The order remained fixed independent of the order of conditions. The whole session was recorded on video. After each condition a questionnaire was handed out. In the end there were semi-structured interviews. No time limits were given since this could have interfered with our measures. Therefore, depending on the group the whole session could take up to two hours including introductions, questionnaires and interviews.

The Strict and Free Ludo conditions were done on a Microsoft Surface Tabletop (cf. Figure 6.12). In the Distributed Ludo condition the group was seated on different sides of a meeting table. Each of the three group members was provided with a laptop and mouse. A beamer was used to project the shared workspace onto a wall next to the table.

### *Measures*

To evaluate our research questions all sessions were recorded on video. The videos were analyzed to get an objective assessment on the group interaction. As an objective measure for *role*



**Figure 6.12:** Strict and Free Ludo tabletop setting (left) and Distributed Ludo setting (right).

*assignment*, the number of different roles per learner was counted. Furthermore, the number of positive and negative awareness indicators for partners' roles and tasks was counted. Positive indicators would be statements such as '[...] you are the <name of the role> [...]' or '[...] your task is [...]'. A negative indicator would be 'What is your role?' or 'What do you have to do?'.

In addition, identical questionnaires were handed out after each condition containing 5-point Likert scale items. The scale was from 1 ('completely disagree') to 5 ('completely agree'). The questionnaire contained different sets of items on *role assignment*, on the participants' *awareness* of their partners, on the *group interaction* as well as on the *individual work*. Finally, there was one last questionnaire in which participants were also asked demographic questions and ranked the Ludo versions according to their *popularity*.

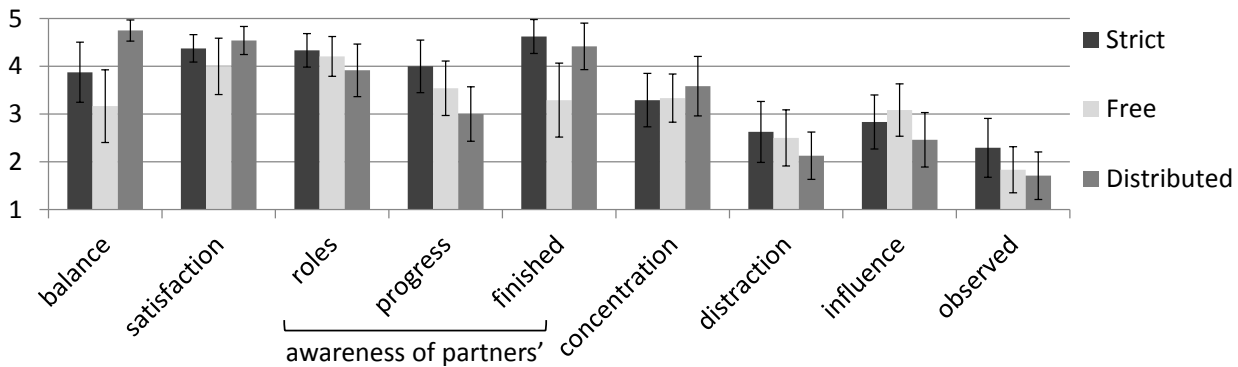
### *Statistical Tests*

To evaluate the questionnaire results Friedman two-way analysis of variance tests were applied to the 5-point Likert scale data. In cases of significant effects pairwise comparisons were done using Wilcoxon signed-rank tests.

## 6.2.5 Results

### *Role Assignment*

The maximum number of roles that one participant could take was seven (reader, questioner, recorder, summarizer, listener, student and teacher, cf. Figure 6.6). In Distributed Ludo the number of roles per person was always  $M=7.00$  ( $SD=0.00$ ) due to the automatic role distribution. In Strict Ludo it should have been the same. However, due to an implementation error, which was only corrected after the first four trials, participants only took  $M=6.25$  ( $SD=.94$ ) roles. In Free Ludo the number of roles was only 5.71 ( $SD=1.46$ ).



**Figure 6.13:** Ludo questionnaire results on a Likert scale from 1 (completely disagree) to 5 (completely agree).

Regarding the subjective *balance* of role distribution (see Figure 6.13 Distributed Ludo was rated significantly better ( $M=4.75$ ,  $SD=0.44$ ) than Strict Ludo ( $M=3.88$ ,  $SD=1.26$ ) and Free Ludo ( $M=3.17$ ,  $SD=1.52$ ,  $\chi^2=14.69$ ,  $p=.001$ ). Post-hoc Wilcoxon tests revealed significant differences between Free and Distributed Ludo ( $Z=-3.44$ ,  $p=.001$ ), and between Strict and Distributed Ludo ( $Z=-2.63$ ,  $p=.009$ ). Thus, Distributed Ludo performed best regarding the perceived balance of role distribution. This is not what we expected because both Distributed and Strict Ludo have an equally strict guidance with regard to the role distribution. An explanation is the above-mentioned implementation error in Strict Ludo, which led to an unbalanced role distribution. It is likely that without this error Strict Ludo would have performed equally well. The finding that learners perceived the role distribution as less balanced in Free Ludo is conform to our expectations.

The *satisfaction* with the role assignment was rated with  $M=4.38$  ( $SD=0.58$ ) for Strict Ludo,  $M=4.00$  ( $SD=1.18$ ) for Free Ludo and  $M=4.54$  ( $SD=0.59$ ) for Distributed Ludo. There was no significant effect.

### *Awareness of Learning Partners' Roles*

The subjective *awareness of the partners' roles* was rated  $M=4.33$  ( $SD=.70$ ) for Strict Ludo,  $M=4.21$  ( $SD=.83$ ) for Free Ludo and  $M=3.92$  ( $SD=1.10$ ) for Distributed Ludo. There was no significant effect. The video analysis revealed two positive indicators for an awareness of a partner's role in Strict Ludo compared to six in Free Ludo and five in Distributed Ludo across all groups. As for negative indicators (i.e. statements that indicate a lack of awareness of a partner's role), there was one in Strict Ludo, none in Free Ludo and two in Distributed Ludo. There were also positive indicators for partner's tasks: 28 in Strict Ludo, 16 in Free Ludo and 19 in the Distributed Ludo conditions (again across all groups). Furthermore, there were two negative indicators, which show a lack of awareness for the partner's tasks, in Free Ludo, none in the other two conditions. These numbers do not reveal a clear trend. Consequently, neither the subjective nor the video analysis indicated that the Ludo version had a significant effect on the awareness of the learning partner's roles.

### *Awareness of Learning Partners' Progress*

The subjective *awareness of the learning partners' progress* was  $M=4.0$  ( $SD=1.10$ ) for Strict Ludo,  $M=3.54$  ( $SD=1.14$ ) for Free Ludo and  $M=3.00$  ( $SD=1.14$ ) for Distributed Ludo. There was a significant effect,  $\chi^2=10.60$ ,  $p=.005$ . Post-hoc Wilcoxon tests revealed a significant difference between Strict and Distributed Ludo,  $Z=-3.27$ ,  $p=.001$ . Therefore, the awareness of the learning partner's progress was significantly higher in Strict Ludo than in Distributed Ludo.

Regarding the question whether the participants were *aware which partners had finished* their current task, there was also a significant effect,  $\chi^2=6.86$ ,  $p=0.32$ . Post-hoc Wilcoxon tests revealed that Free Ludo was significantly lower rated  $M=3.29$  ( $SD=1.55$ ) than Strict Ludo  $M=4.63$  ( $SD=0.71$ ),  $Z=3.01$ ,  $p=.003$  as well as Distributed Ludo  $M=4.42$  ( $SD=0.97$ ),  $Z=-2.82$ ,  $p=.005$ .

### *Group Interaction*

The subjective *concentration* on the partners was  $M=3.29$  ( $SD=1.12$ ) for Strict Ludo,  $M=3.33$  ( $SD=1.01$ ) for Free Ludo and  $M=3.58$  ( $SD=1.25$ ) for Distributed Ludo. There was no significant difference. In the interviews conducted at the end of the study three people stated that the laptops provide a barrier to the peers. However, other participants considered the laptop condition to be more communicative than the tabletop because the gaze is directed towards the surface while there is more attention shifting in the laptop setup.

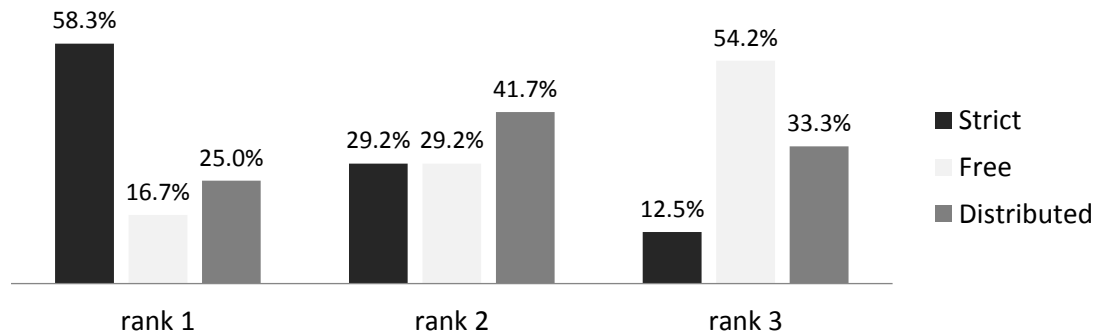
The subjective *distraction* caused by the application was also rated fairly equal across all conditions. For Strict Ludo the answer was  $M=2.63$  ( $SD=1.28$ ), for Free Ludo  $M=2.50$  ( $SD=1.18$ ) and for Distributed Ludo  $2.13$  ( $SD=0.99$ ). There was no significant effect. During the interviews, six people declared the tabletop to be distracting, some explicitly mentioned the playfulness or the tangibles as reason. Two other participants stated that Distributed Ludo is less distracting due to the moderate use of color and the missing direct touch input. In summary, there was no indication that the Ludo versions had a significant effect on the group interaction.

### *Individual Work*

The subjective *influence* of the partner's activities was rated  $M=2.83$  ( $SD=1.13$ ) for Strict Ludo,  $M=3.08$  ( $SD=1.10$ ) and  $M=2.46$  ( $SD=1.14$ ) for Distributed Ludo. There was no significant effect. Regarding the question whether the participants felt *observed* by their learning partner's there was a significant effect,  $\chi^2=7.39$ ,  $p=.025$ . Post-hoc Wilcoxon tests revealed that the answers were significantly higher for Strict Ludo ( $M=2.29$ ,  $SD=1.23$ ) than for Distributed Ludo ( $M=1.71$ ,  $SD=1.00$ ,  $Z=-2.01$ ,  $p=.45$ ). To the question whether it was convenient to have a private workspace 71% agreed or even strongly agreed. 17% were neutral. The remaining 12% disagreed.

### *Popularity*

In the final questionnaire the participants were asked which Ludo version they liked best. 58.3% of the participants preferred Strict Ludo, 16.7% chose Free Ludo and 25.0% favored Distributed



**Figure 6.14:** Popularity of the three Ludo versions.

Ludo. Figure 6.14 shows the complete popularity rankings: Strict Ludo was best ranked, Distributed Ludo second-best and Free Ludo was the least popular version.

The semi-structured interviews provide further insights on the reasons for these preferences. The experimenter asked the participants to reflect on benefits and drawbacks of the different solution. Nine participants judged the strict guidance as good. Typical comments were ‘*A predetermined role distribution allows setting the focus on the learning activities*’, ‘*[...] it ensures a fair role distribution*’ or ‘*A strict role distribution is better because the least popular roles are most beneficial for learning*’. There were no comments that spoke in favor of a free role distribution.

Furthermore, there was diverse feedback on the display environment. Figure 6.15 lists all positive and negative comments on the two display environments, which were mentioned by at least three participants. Seven participants mentioned independent from each other that they liked the playfulness of the tabletop applications and thought it was suitable for the learning application. Five participants argued that the tabletop setting had a positive effect on the group interaction. There were different reasons for this statement, for instance that the group members sit together more closely and that the direct interaction on the shared workspace without any physical barrier facilitates a better group interaction. The better awareness regarding the partners’ roles and activities was also mentioned.

Negative comments on the tabletop versions regarded the distraction through the application. Of the five comments in this direction, one specifically held the colorfulness responsible for the distraction, two the playfulness and another one the shared workspace, which seduced him to look around. Moreover, five participants found the tangibles annoying.

The most frequent positive comments on the distributed application regarded the better ergonomics (mentioned by seven participants) as well as less distraction due to the private workspaces (mentioned six times). In addition, five participants found clicking a button easier than moving around tangibles. Furthermore, it was appreciated that one felt less observed (mentioned four times), that the overview was better (mentioned 3 times) and that laptops are a more accustomed working environment. The only negative comment on the distributed setup, which was mentioned more than twice, was that the laptops are a barrier to group interaction.



	Tabletop	Laptops & Wall
Positive comments	<ul style="list-style-type: none"> <li>+ Playfulness (7 comments)</li> <li>+ Better group interaction (5)</li> <li>+ Better awareness regarding partners' roles (4)</li> <li>+ Better awareness regarding partners' activities (4)</li> </ul>	<ul style="list-style-type: none"> <li>+ Better ergonomics (7 comments)</li> <li>+ Less distraction due to private workspace (6)</li> <li>+ Clicking a button is easier (than moving around tangibles) (5)</li> <li>+ Not feeling observed (4)</li> <li>+ Good overview (3)</li> <li>+ Laptop is an accustomed working environment (3)</li> </ul>
Negative comments	<ul style="list-style-type: none"> <li>- Distraction (5 comments)</li> <li>- Tangibles are annoying (5)</li> </ul>	<ul style="list-style-type: none"> <li>- Laptops build a barrier to the group interaction (3 comments)</li> </ul>

**Figure 6.15:** Participants' comments on the display environments (and the number of times they were mentioned by different participants).

## 6.2.6 Summary and Discussion

Ludo is an extension of the Task Tokens. While both concepts aim at assigning learning tasks to a group of learners, Ludo also assigns different roles to group members for each of these tasks. Three versions of Ludo were compared in a user study: Strict Ludo and Free Ludo are both tabletop applications, which differ in the strictness of guidance. Distributed Ludo runs on laptops and a shared wall projection and applies a strict guidance similar to Strict Ludo.

Regarding the strictness of role assignment, there was a significant effect on the perceived balance of role distribution, which speaks in favor of the strict assignment. More importantly, the semi-structured interviews revealed that the strict assignment was preferred over the free assignment, although it implicated more movement of tangibles, which the participants found annoying. Nine out of 24 participants elaborated on the benefits of a strict, automatic role distribution, none advanced the opposite view. Objective measurements were not the main purpose of this study for two reasons. First, the frequency of role changes in the laboratory setting was fairly high due to time constraints. Therefore, a decreased frequency of role changes in Free Ludo would not be a meaningful result - it would merely reflect the laboratory constraints. Second, a strict, automatic role distribution causes by definition an equal balance, which is beneficial for collaborative learning (Zafeiriou et al., 2001). Therefore, the interesting question was neither whether Strict Ludo causes balanced role distribution nor whether it is beneficial for learning (which was shown in previous work, cf. Palincsar and Brown, 1984). We were interested in finding negative implications of strict role assignment such as a lack of acceptance by the learners, which was ruled out by the study.

Furthermore, effects on the awareness of the learning partners were examined. There was no significant difference regarding the subjective awareness for the partners' roles. However, the perceived awareness for the partners' progress was significantly higher in Strict Ludo compared to Distributed Ludo. This is not surprising because of the private workspaces in Distributed

Ludo. Furthermore, participants were asked whether they were aware which learning partners were finished with their current task. This kind of awareness was worse rated in Free Ludo compared to both other conditions. The higher awareness in Strict Ludo compared to Free Ludo can be explained by the personal tangibles, which needed to be placed on the next task by each learner once she was finished with the task. However, we were surprised that Distributed Ludo also performed significantly better than Free Ludo in this regard. We expected the awareness to be lowest in Distributed Ludo because the only indicator is the green checkmark icon next to the avatar on the wall display (and whether the group member is still interacting with the laptop at all). Overall, the effects on the different aspects of awareness were by far smaller than expected.

Next to the somewhat better awareness other characteristics of the tabletop versions that were appreciated by the participants were the playfulness of the application as well as the better group interaction. While we consider the better group interaction as clear benefit, the playfulness also has its downside. First, more participants mentioned negative implications of the playfulness, e.g. that they were distracted or that the tangibles were annoying. Second, research has shown that playfulness can distract learners from their tasks (cf. Hinrichs et al., 2008 or Sugimoto, 2009).

In addition, there were many clear advantages of Distributed Ludo. Most important was the ergonomical aspect. Many participants complained that leaning over the surface was uncomfortable and exhausting over time. This problem is particularly severe on the Microsoft Surface due to its coffee table height but has been found in other tabletop environments as well (e.g. by Morris et al., 2008). On the contrary, the laptops were not only more comfortable to use but also the more familiar working environment. Furthermore, most participants appreciated the private workspace and felt less observed.

### 6.3 Towards Automatic Code Generation

Guiding widgets provide an intuitive way of assigning tasks and roles to co-located collaborative learners. This functionality is currently not provided by the XSS Framework, which was presented in chapter 3. Currently, a typical application created with the XSS Framework provides guidance through a task title and description. However, there is no way of representing the learning partners and their roles (other than mentioning them in the task description). In contrast to the Task Tokens or Ludo learners need to verbally communicate their tasks and roles which are also automatically assigned by the application. If the group only consists of two learners (as in our example application M.U.R.D.E.R. in section 3.2) the coordinative effort, which is imposed on the learners, is still acceptable. However, the larger the learning group is, the more useful are Guiding Widgets as they ease coordination. In addition, applications developed with the XSS Framework do not provide a visualization of the task sequence - they merely guide the learners through the sequence 'blindly' by presenting a sequence of screens each of which provides the means to fulfill the task. There is also no way of knowing when the learning partner is finished - the application just asks finished learner to wait.

Consequently, the functionality of the XSS Framework (as well as of the applications developed with it) could benefit from integrating Guiding Widgets. Distributed Ludo provides a good example of a solution for distributed applications (which are currently supported by the XSS Framework). The role indicator of the Ludo laptop (on the right of Figure 6.10) could be added to the XSS Dummy Clients (cf. Figure 3.10). In addition, there could be a shared representation, such as the Ludo wall (Figure 6.10). All of this could be generated automatically.

As the XSS Framework at its current state only supports distributed applications, a concept for an extension towards shareable user interfaces was proposed in section 3.5. On top of this extension, the XSS Framework for shareable user interface could be further extended with regard to Guiding Widgets. Strict Ludo provides a good example how this could look like. As in Strict Ludo, the shared workspaces (cf. Figure 3.18, right) could show a representation of the task sequence, whereas the private workspaces contain (placeholders for) role indicators and task descriptions. Again, this functionality could be automatically generated by an extended XSS framework.

## 6.4 Related Work

This work is related to several research areas. First, there is the large body of research on collaboration scripts, which has been introduced in section 2.1. Of particular interest is the learning strategy ‘Reciprocal Teaching’ (Palincsar and Brown, 1984), which defines the learning tasks used in this application. Also related are other tabletop applications that support collaborative learning (cf. section 2.2.1), even though these applications are not based on collaboration scripts.

Second, in the field of computer-supported collaborative work (CSCW), there are a number of studies on social protocols that many tabletop applications rely on, for example asking team members before taking action (Izadi et al., 2003). In order to avoid conflicts that might occur despite such protocols, Morris et al. (2004) established a set of coordination policies, which are graphical metaphors shown on a shared display to clarify the rights of each participant. Another approach is using personal workspaces that are either visually highlighted or implicitly defined by the proximity to the workspace owner. For instance, on the DiamondSpin (Shen et al., 2004) ownership is implicitly defined by windows that automatically turn towards the closest outside edge of the tabletop. The Task Tokens and Ludo use this research as a foundation for intuitive task and role assignment. However, to our knowledge there have not yet been any tabletop or shareable user interface applications that specifically target task distribution. A probable reason is that scripted applications are often realized on traditional personal computers (e.g. Clark et al., 2010 or Stegmann et al., 2007b). On personal computers roles are usually implicitly distributed by showing different task descriptions on different computers, which is an entirely different approach than ours.

## 6.5 Lessons Learned

Two important aspects of scripted collaborative learning are guiding learners through a sequence of learning activities and assigning roles to different group members to induce socio-cognitive conflict. This chapter presented two GUI concepts that use Guiding Widgets to assign tasks and roles to co-located learners. At first, the Task Tokens were developed and examined in a user study. The study showed that the Task Tokens are very easy to understand, easy to use and effectively guide learners through a sequence of tasks. However, from an educational perspective, further guidance regarding the assignment of different roles for each task was desirable. We therefore extended the concept and created Ludo, which assigns roles for each task. A second study showed that Ludo effectively assigns tasks as well as roles. These examples demonstrate that Guiding Widgets can successfully guide learners with regard to certain components of collaboration scripts. Our experience also shows that more than one kind of Guiding Widgets (e.g. for tasks *and* roles) can be integrated in carefully designed GUI concepts without overwhelming the learners.

Furthermore, different versions were compared, which varied in the applied strictness of guidance. In both studies, learners preferred an automatic task (study 1) and role (study 2) assignment. They considered it fair and appreciated that the coordinative effort is taken care of by the application. Of course, this finding cannot be generalized to ‘the stricter the learning application, the better’. However, it is reasonable to assume that strict role and task assignment is generally preferred in learning settings with complementary roles and/or with roles of different popularity. Learners neither appreciated their freedom of choice in the Task Token study, where they are forced to volunteer for tasks, nor in Ludo, where there was no social pressure (as learners could simply keep their role for as long as they wanted).

Finally, while the Task Tokens and the original version of Ludo were both designed for tabletop displays, we showed that Guiding Widgets (and the according GUI concepts) can be transferred to distributed applications (e.g. to a combination of laptops and wall display) - leading to comparable success. Contrary to expectations the level of awareness regarding the learning partners’ roles was almost equally high in Distributed Ludo, despite the fact that the learners’ attention shifted between the private workspaces and the shared wall display. Only the awareness of the partners’ progress was higher in the tabletop versions. Consequently, the advantages of the tabletop setup were lower than expected. In addition, there were some major benefits that the participants saw in the distributed setting. For instance the laptops were found to be less distractive than the tabletop application. Also, laptops are the more familiar working environment, provide easy means for text input and are more ergonomic. Furthermore, the availability of private workspaces was appreciated by the majority of the participants. In addition, there is an organizational benefit as laptops and wall displays are ubiquitously available. These findings leave serious doubt as to whether tabletops are the preferable display environment for co-located scripted collaborative learning.

# Chapter 7

---

## Using Ambient Feedback for Mirroring and Guidance

The previous work examined guidance mechanisms, which are integrated in learning applications. A different approach is using separate mirroring or guidance systems, which are dedicated to the purpose of providing feedback. Such systems collect information about the learners' collaboration and visualize it or give advice based on the data (Jermann et al., 2001). For instance, participation levels of all group members can be captured via microphones and displayed on the wall next to the collaborating group (DiMicco et al., 2004). We consider this form of guidance as *explicit* because the feedback is unambiguously perceived as such. Furthermore, the guidance is less strict than previously discussed mechanisms because the feedback can easily be ignored - in particular because it is not integrated in the learning system. On the contrary, Guiding Widgets, for example, can be overruled but not completely disregarded as they are an integral part of the learning application. At the same time the independence of any learning application makes mirroring and guidance systems flexible tools that can be used in combination with learning applications (and therefore in combination with other guidance mechanisms such as Guiding Widgets) as well as in pen-and-paper-based learning settings.

**Chapter Question:**

How can learners be guided through collaboration scripts explicitly using non-interactive mirroring and guidance systems?

This chapter investigates how dedicated mirroring and guidance systems can be applied to scripted collaborative learning scenarios. After presenting related work on mirroring and guidance systems in general (section 7.1), we describe two studies that examined effects of *script-based* mirroring and guidance systems. In the first user study (cf. section 7.2) two visualizations of the same data (metaphor and diagram) were compared to a condition without group mirror. While both visualizations had a significant positive effect on the collaboration, the metaphor led

to significantly faster reactions compared to the diagram. As a follow-up project we moved out of the laboratory settings and implemented a similar system in a practical seminar at a German high school. The experience in the classroom is presented in the section 7.3.

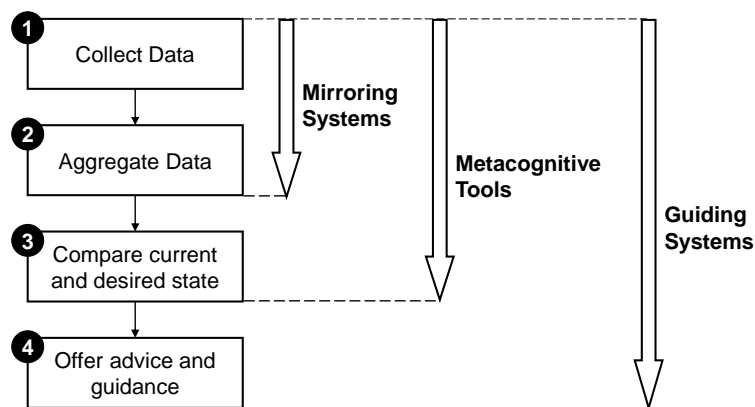
## 7.1 Mirroring and Guidance Systems

Feedback plays an important role in collaborative processes. Usually collaborators self-regulate their behavior according to feedback they get during collaboration. In an argument, for example, a participant will add additional backing for a claim if another participant shakes her head. Furthermore, several empirical studies showed a positive effect of feedback on knowledge acquisition in learning scenarios (Hattie and Timperley, 2007).

This section reviews existing mirroring and guidance systems, which provide feedback to improve learners' collaboration. At first, different types are characterized by their functionality (mirroring vs. guiding). Afterwards, several mirroring and guidance systems for co-located groups are presented and categorized.

### *Mirroring vs. Guidance*

While all mirroring and guidance systems collect data on the group's behavior, there are different ways of using the data. The simplest form is '*mirroring systems*' or '*group mirrors*', which reflect the aggregated data to the group, assuming that the group members know how to adapt their behavior. More complex systems use the data to determine the current state of collaboration and compare it with the desired state (cf. Figure 7.1). Jermann et al. (2004) call these systems '*metacognitive tools*'. The most advanced type is '*guidance system*', which offer advice or guidance based on the collected data.



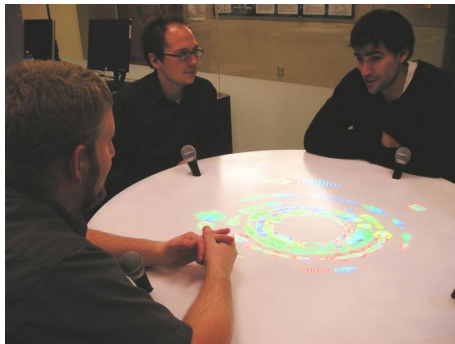
**Figure 7.1:** Types of mirroring and guidance systems (according to Jermann et al., 2004).

Soller et al. (2005) provide an extensive review of different types of mirroring and guidance systems and analyze their functionality from an educational perspective. Most of these systems are

distance learning applications such as online discussions or newsgroup-like systems. More relevant for this thesis are mirroring and guidance systems for co-located settings. In the following such systems are presented from a user interface perspective.

### *Mirroring and Guidance Systems for Co-Located Groups*

In co-located settings mirroring and guidance systems often make use of shared displays such as wall or tabletop displays. Tabletop displays are positioned in the center of the group (see Figures 7.2 and 7.3) whereas wall displays are usually peripheral. There are several mirroring systems that reflect participation levels. For example, DiMicco et al. (2004) used a diagram to show participation shares of all group members on a large wall display. Experiments showed that the presence of such visualizations has an effect on the participation levels (DiMicco and Bender, 2007). A table-based approach with a similar goal is *Reflect* (Bachour et al., 2008), which uses colored LEDs that are embedded in the table to show participation levels. Bergstrom and Karahalios (2007b) presented *Conversation Clock*, a circular representation of speaking turns. It focused less on balance but rather on the participation history and conversation flow. In an experiment the system led to increased awareness for the participant's own and the group's interaction. A more sophisticated group mirror is the *Meeting Mediator* (Kim et al., 2008). It incorporates information about speaking times, overall balance, and interactivity (i.e. the frequency of turn changes) in group collaboration. The multi-dimensional information is aggregated in a rather complex visualization, using a colored circle to indicate the degree of interactivity.



**Figure 7.2:** Conversation Clock (Bergstrom and Karahalios, 2007a)



**Figure 7.3:** Reflect (Bachour et al., 2010)

Most group mirror approaches provide feedback about quantitative data, and therefore mainly focus on speaking times and participation. This data is fairly easy to measure with microphones. At the same time the information is rather superficial. An approach focusing only on quantity does not differ, for example, between off-topic and on-topic contributions. Therefore, some approaches pursue the idea of qualitative feedback. For instance, *Conversation Votes* (Bergstrom and Karahalios, 2007c) allow users to highlight certain moments of conversation by anonymous votes. In addition, Sturm et al. (2005) presented two approaches for feedback on attention and interest. They are intended for usage in group meetings or classrooms. While the group tool illustrates speaker and listener attention of each group member, the classroom tool provides the

teacher with indicators for the student's attention and level of interest. The indicators are controlled by the students.

In summary, there are several mirroring and guidance systems, which vary in three main aspects: (1) the type of information (qualitative vs. quantitative), (2) the visualization (ranging from simple diagrams to rather complex visualizations) and (3) the placement of the feedback (usually on wall or tabletop displays).

*In the remainder of this chapter we will use the term 'group mirrors' for our systems although they also have some guidance components. This term was chosen to avoid the more adequate but long term 'mirroring and guidance system' (and because we find the term 'regulation system' ambiguous).*

## 7.2 An Ambient Group Mirror for Qualitative Feedback

Traditionally feedback is given by teachers or tutors. Group mirrors may provide feedback in a subtler and less interruptive way as the feedback is provided visually instead of verbally. However, if the feedback provided by the group mirror requires too much attention, collaborators may be distracted from the task as well as require more time to react to the feedback. Therefore, the type of feedback representation plays an important role<sup>1</sup>.

Previous work has shown that group mirrors can affect group behavior (see section 7.1). However, group mirrors have many different characteristics, not all of which have been equally examined so far. The following study specifically investigates effects of different feedback representations. Unlike the previous chapters, which examined actively used shared displays, a passively used ambient display was chosen to mirror the quality of argumentation in small groups. Two different representations are compared with a baseline condition without group mirror in a controlled experiment<sup>2</sup>. The first is a diagram, which is commonly used for group mirrors. Second, a metaphoric representation visualizes the same data by changing parts of a pictorial scenery as shown in Figure 7.4. The feedback is implicitly shown by the weather, which can range from cloudless sky to heavy rain, as well as the condition of the trees, which blossom when the quality of argumentation is very good or lose their leaves when the performance is poor. In the baseline condition no group mirror is used at all.

In the following, the collaboration scripts that were used to define ideal collaborative activities in terms of tasks, roles and phases are introduced (section 7.2.1). Afterwards, the different feedback representations are described as well as the control interface, which is used to manually enter the quality of contributions (section 7.2.2). In the end, the user study is presented and conclusions on the development of effective group mirrors are drawn.

---

<sup>1</sup> Section 7.2 contains extracts of the publication by Streng et al. (2009a).

<sup>2</sup> The group mirror was implemented by Younes Alj (2009) under the supervision of the thesis author.



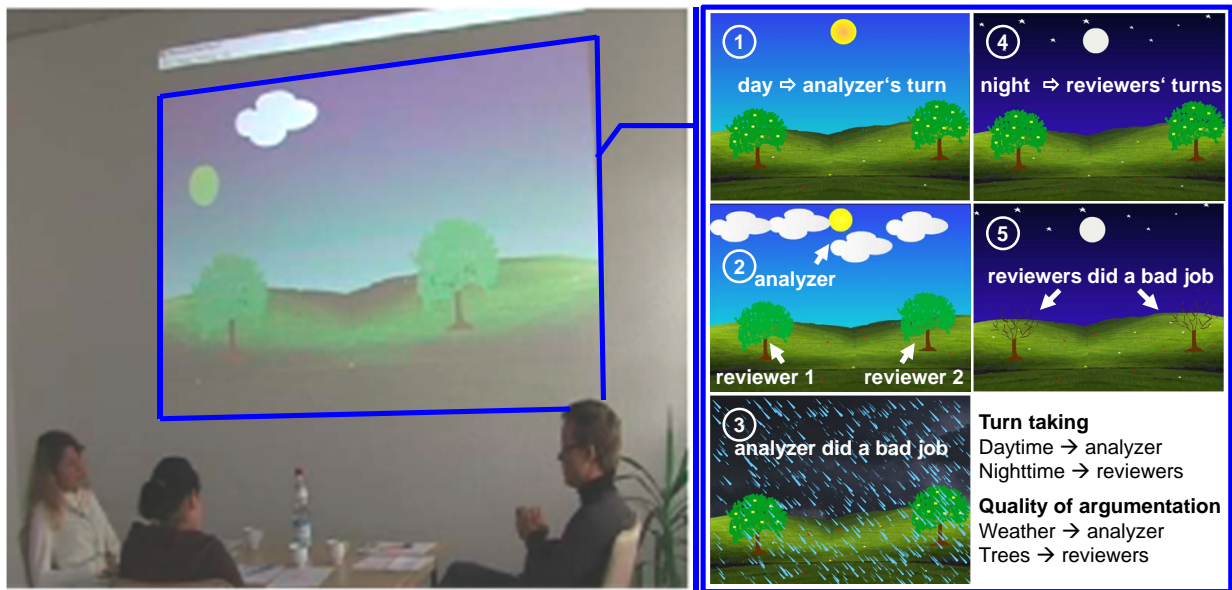


Figure 7.4: Metaphoric Group Mirror in Use.

### 7.2.1 Concept

In comparison to most group mirrors, which reflect quantitative information, our group mirror reflects the *quality* of learners' contributions. As pointed out by Dillenbourg (2005) the usefulness of mirroring pure quantitative data is questionable. Taking the example of participation rates, while unbalanced participation is an indicator for a deficient (learning) outcome, the opposite is not necessarily true. In order to leave a good impression on the group mirror, participants could even produce useless arguments, which would undermine the purpose. Therefore, group mirrors should ideally (also) offer feedback on the quality of collaborative processes. However, qualitative data is by far more difficult to acquire. While directing a microphone to each person allows for easy capturing of participation shares, it is much more complicated to assess the quality of collaborative processes. The first problem is the definition of the criteria that are used to assess the quality of collaboration. Second, it is unclear how to measure whether the criteria are fulfilled.

#### *Collaboration Scripts as Criteria for Qualitative Assessment*

To be helpful, feedback has to include information on the difference between the target state and current state. The target state has to be known by the feedback recipient. To define the target state a peer-review script (Weinberger, 2003) was used. It distributes the roles of problem analysts and critics and specifies the activities for each role. The problem analyst has to provide an analysis for a given problem and has to make proposals how to solve it. The critics have to provide constructive critique including alternative proposals. Each activity has a defined duration (two minutes for the analyzer and two for both critics together). The learners are advised not to

interrupt the speaker except for clarification questions. Usually, the script includes a rotation of roles but for reasons of simplicity role reversal was abstained from in this experiment. Furthermore, based on a second script by Stegmann et al. (2007b) participants were asked to provide reasons and qualifications for their claims. The same script was already used by the ArgueTable and ArgueWall (see section 4).

### *Measuring the Quality of Collaborative Processes*

Current interdisciplinary approaches of learning scientists and computer linguists show that the automated assessment of quality of collaborative processes is already possible with high accuracy for written discussions. Rosé et al. (2008) trained automated classifiers for different dimensions of qualities of collaborative processes, e.g. content or quality of argumentation. The authors used classified written online discussions and trained automated classifiers with this data. Therefore, a system that transcribes spoken discussions and applies such automated classifiers to the transcribed discussion does not seem unrealistic. However, it is still a long way to a reliable qualitative real-time analysis of spoken arguments. In the meantime, a moderator or teacher can do the assessment by manually rating the quality of arguments. The advantage of using a group mirror instead of spoken feedback is the unobtrusiveness. The tutor can silently give feedback without having to interrupt the ongoing discussion.

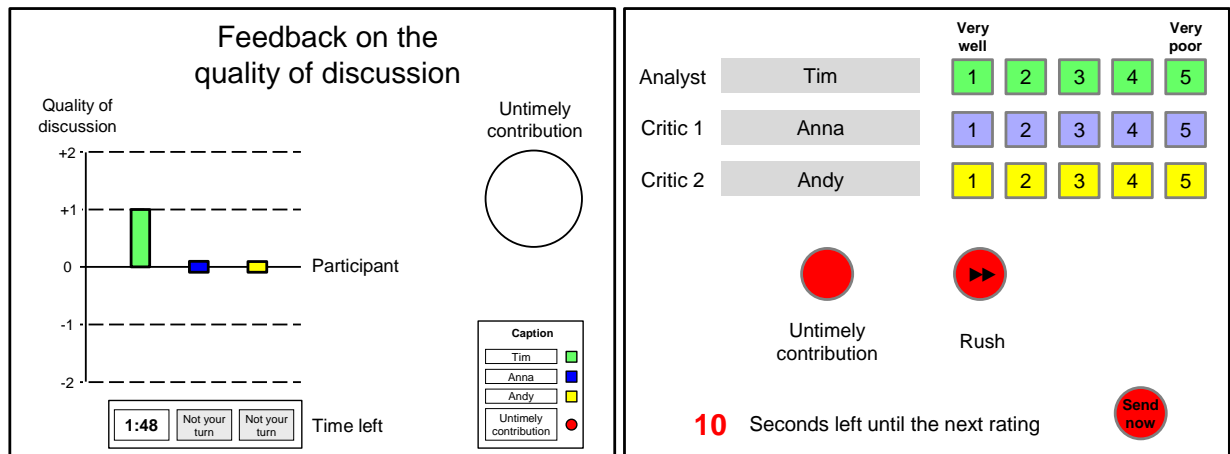
## 7.2.2 Design and Implementation

The prototypical implementation consisted of two connected laptops. One of them showed the group mirror representation, which was projected to a wall. The other laptop was used by the wizard to manually enter the quality assessment that was visualized by the group mirror. Both graphical user interfaces were implemented using Flash and ActionScript.

The group mirror representations have to match the quality criteria defined in the collaboration script. In this respect we agree with literature on awareness tools (Espinosa et al., 2000), which also came to the conclusion that the representation highly depends on the task at hand. In our case the group mirror has to provide feedback on the compliance with the peer review script (turn taking, untimely contributions, duration of contribution) and the quality of argumentation (bare claim vs. grounded claims with qualification). In the following two different visualizations are presented to provide this feedback. One is a diagram, which is commonly used for group mirrors. The other is a metaphorical representation of the same information.

### *Metaphoric Group Mirror*

Figure 7.4 shows five screenshots of the metaphoric group mirror. It was designed to provide feedback to three people regarding the quality of their argumentation. The analyst's quality of argumentation is represented by the weather. There are five stages from cloudless sky (the analyst is providing grounded claims with qualifications as in picture 1) to strong rain (the analyst is providing bare claims as in picture 3). The analyst's phase starts with sunrise and ends with



**Figure 7.5:** Group mirror diagram (left) and control interface (right).

sunset. However, the timeline is just a suggestion. If the analyst takes less time, the sunset is accelerated. If the analyst takes more time, the sunset ‘waits’ until the learners are finished with their turns.

Each of the two critics is represented by a tree. The names of the critics are placed below their tree. The trees also have five stages to reflect the quality of argumentation. If a critic provides grounded claims with qualifications the corresponding tree is flourishing (as in picture 4). If the critic provides just bare claims, the tree loses its leaves (see picture 5). The critics share the night-time, which means that the handover should happen when the moon is at the uppermost point. Again, this is only a lead, not an instruction. Initially average ratings are shown: partly cloudy sky and trees with leaves but without blossoms (both matching a 3 out of 5 stages as in picture 2).

A sudden lightning covers the whole display when one participant interrupts another. The lightning lasts until the interruption ends. As the lightning covers the whole display, the emitted warning is directed towards the whole group whereas all other information is mapped to a specific part (tree or weather) and thus to a specific person. However, we expect that the interrupter is likely to feel addressed no matter whether the bolt will hit one of the trees or not.

### *Diagram Group Mirror*

In order to compare the effectiveness of diagram and metaphor, the mirrored information must be the same for the diagram. The quality of people’s arguments are shown in a column chart (see Figure 7.5, left). Each of the three columns represents one person. The five rating stages are centered around the neutral value 3. Below the bar is a count down that shows the time intended for the corresponding participant. Of course, only one of three count downs is active at a time.

In case of an interruption the circle at the right side starts flashing in a bright red color. The large size of the circle was chosen to call for a similar attention as the lightning in the metaphoric version. All components are explained in the legend at the bottom right.

### *Control Interface*

The group mirror can be controlled with a simple user interface as shown in Figure 7.5 on the right. For the purpose of this experiment the Wizard of Oz approach (Kelley, 1983) was used. Participants thought that the experimenter takes notes on the laptop. They were not told that the experimenter is the Wizard who rates the quality of their argumentation and their compliance to the peer review script. In learning settings the same control interface can also be used by a teacher or tutor to give feedback to a group of learners without interrupting them.

In the upper part of the control interface the quality of argumentation for each of the three participants can be entered on a scale from 1 (grounded and qualified claims) to 5 (bare claims). The ratings are automatically sent to the group mirror every 10 seconds. If the wizard/tutor wants an immediate update, the ‘send now’ button can be used. The ‘untimely contribution’ button turns the lightning or blinking red circle on and off, which signals an interruption by one of the participants. The ‘rush’ button can be used to accelerate the timer (or sun- or moonset respectively) when the current speaker finishes early.

### 7.2.3 Method

A Wizard of Oz experiment was conducted to evaluate the effectiveness of the group mirror in general as well as the different feedback representations. The following research questions were investigated:

*RQ1: To what extent does the type of representation (none vs. diagram vs. metaphor) facilitate the self-regulation of participants in a group discussion?* We expected that both the diagram and the metaphor have a positive effect on the self-regulation compared to the baseline condition without feedback. Furthermore, we expected learners to react faster to the metaphoric feedback than to the diagram.

*RQ2: To what extent does the type of representation (none vs. diagram vs. metaphor) affect the subjective estimation of compliance to the collaboration script?*

### *Participants and Design*

Each discussion was held with groups of three participants. Altogether 27 people at the age of 19 to 53 participated in the study. 50% were female. Two participants were university students, three were apprentices and the remaining participants were employees with various levels of education and backgrounds (secretaries, hairdressers, branch managers, ...). To compare the three *types of representations* (without vs. diagram vs. metaphor) a within subject design was chosen. Therefore, the same group of participants had three discussions, one in each condition.

The baseline condition without feedback representation was always first. The order of diagram and metaphor was randomized. The roles of the peer review script were fixed so that each person had the same role across all conditions. In the control condition no mirroring system was used at all. During the condition with group mirror the diagram or the metaphor was projected onto a wall next to the table where the group of three discussed.

### *Learning Task and Procedure*

The topics that were chosen for discussion had attracted public attention in Germany over a long period of time. Therefore, no additional material and preparation time was needed to discuss these topics. The topics were (1) data security and online search, (2) nuclear and green energy and (3) speed limits and carbon emission. In the beginning the experimenter gave a short overview on the different steps of the study. Before the participants were asked to discuss the first topic, there was a short introduction on the collaboration script and the quality criteria. In addition, before each condition with a group mirror the experimenter explained the feedback representation. After each discussion the participants filled out a questionnaire.

### *Measures*

We measured *self-regulation* of collaborative processes by computing the time between a negative feedback and the moment when the process was improved or the end of a turn was reached. In the baseline condition without group mirror, the quality of the discussion was assessed in the very same way but the information was not reflected to the group. After each discussion with a group mirror representation the same questionnaire was filled out. In the questionnaire participants were asked for the subjective estimation of *compliance to the role* as well as *compliance to the timing*. In the end the students were asked which of the representations they preferred.

### *Statistical Tests*

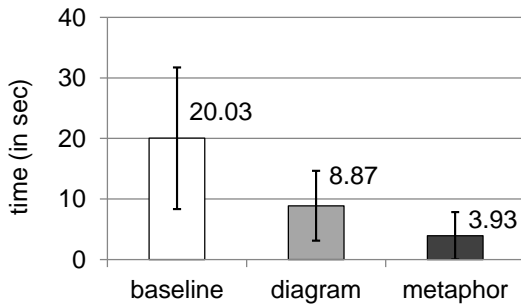
The three conditions were compared using the General Linear Model (GLM) for repeated measures. To compare contrasts post-hoc between conditions without vs. with group mirror or between the diagram condition vs. the metaphor condition, t-tests for dependent variables were applied. Effect sizes are either reported as  $\eta^2$  (GLM) or  $d$  (t-test).

## 7.2.4 Results

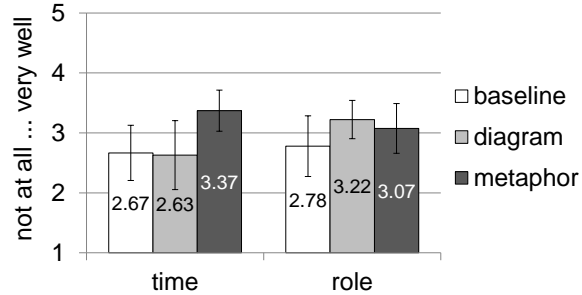
### *Self-Regulation*

The statistical analyses show that the type of representation had a significant and strong effect on the self-regulation of collaborative processes ( $F(2)=8.63$ ,  $p<.05$ ,  $\eta^2=.25$ ). Post-hoc comparisons show that the self-regulation was significantly slower in the baseline condition compared to both, the diagram condition ( $t(26)=2.74$ ,  $p<.05$ , two-tailed,  $d=.53$ ) and the metaphor condition ( $t(26)=3.16$ ,  $p<.05$ , two-tailed,  $d=.62$ ). Furthermore, the self-regulation was significantly slower

in the diagram condition compared to the metaphor condition ( $t(26)=1.98$ ,  $p<.05$ , two-tailed,  $d=.39$ , see Figure 7.6).



**Figure 7.6:** Time taken from a negative feedback until a measured improvement (in seconds).



**Figure 7.7:** Self-assessment of how well the person stuck to time and role (5 = very well, 1 = not at all).

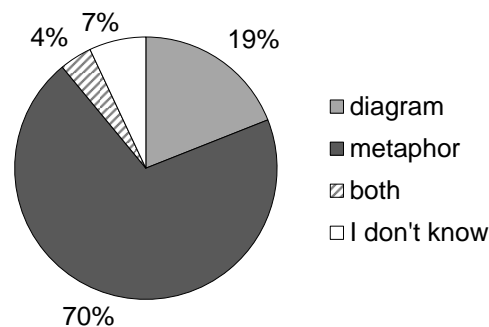
### *Subjective Compliance to the Collaboration Script*

Regarding subjective estimation of *compliance to the role* the GLM shows a significant effect with a medium effect size ( $F(2)=3.10$ ,  $p<.05$ ,  $\eta^2=.11$ , see Figure 7.7). Post-hoc comparisons show that the subjective estimation of compliance to the role was significantly better in the diagram condition compared to the baseline condition ( $t(26)=2.06$ ,  $p<.05$ , two-tailed,  $d=.39$ ). Neither the difference between the baseline condition and the metaphor condition ( $t(26)=1.77$ , two-tailed), nor the difference between the diagram condition and the metaphor condition ( $t(26)=-0.94$ , two-tailed) were significant.

In addition, the type of representation had a significant and strong effect on the subjective estimation of *compliance to the timing* ( $F(2)=7.66$ ,  $p<.05$ ,  $\eta^2=.23$ ). Post-hoc comparisons show that the subjective estimation of compliance to the timing was significantly better in the metaphor condition, compared to both, the baseline condition ( $t(26)=3.68$ ,  $p<.05$ , two-tailed,  $d=.71$ ) and the diagram condition ( $t(26)=3.41$ ,  $p<.05$ , two-tailed,  $d=.65$ ). No significant difference was found between the baseline condition and the diagram condition ( $t(26)=-0.16$ , two-tailed).

### *Popularity*

A significant majority of 70% ( $\chi^2(3)=30.93$ ,  $p<.05$ , two-tailed) would prefer the metaphor (see Figure 7.8). The most frequent reasons given were that it was ‘more vivid’, ‘fun’, ‘aesthetic’, ‘easier to understand’, ‘friendlier’ and ‘better noticeable’. Another positive comment concerned the lack of a clear distinction between positive and negative ratings, which was regarded as ‘pedagogically valuable’.



**Figure 7.8:** Preference for Visualization Types.

### 7.2.5 Summary and Discussion

To investigate effects of group mirror representation an ambient display application was presented. The group mirror reflects whether learners fulfill quality criteria that were defined in two collaboration scripts. In a Wizard of Oz experiment two different representations were compared to a baseline condition without group mirror. The results show that both types of representations had a positive effect on the self-regulation of group processes. However, deficient behavior was corrected significantly faster when receiving negative feedback from the metaphoric group mirror as compared to the diagram condition. Furthermore, a majority of 70% would choose the metaphor over the diagram. These results demonstrate the potential of metaphoric visualizations: They can communicate feedback in a subtler and more playful way while still catching the eye with events that affect larger parts of the screen. Although the transferability of the results to other scenarios needs to be confirmed in further studies, it can be assumed that the advantages of metaphors apply to a wider range of settings than the specific scenario that was used here.

In addition, we have presented an approach that allows to examine other ambient displays in controlled experiments. According to Mankoff et al. (2003) '*ambient displays are abstract and aesthetic peripheral displays portraying non-critical information on the periphery of a user's attention*'. Although numerous ambient display applications have been developed, there has been little in-depth evaluation on them. One major reason is that they are designed to be unobtrusive and interaction occurs spontaneously, which makes it hard to measure effects. With our script-based approach interactions with ambient displays can be condensed to sessions of 10 minutes. That way, it is not only easier to observe the effects of the group mirror, it is also possible to compare different visualizations in controlled experiments.

## 7.3 Group Mirrors in the Classroom

After the promising results of the study, the metaphoric group mirror was used in a project seminar at a German high school. The project seminar was a particularly suitable course because it

involves a high amount of small-group collaboration. In the following, the objective of the project seminar is introduced. Then the concept and design process of the group mirror are presented, followed by the methodology and results of the classroom study<sup>3</sup>.

### 7.3.1 Project Seminar

Project seminars were introduced at Bavarian high schools in the course of the Bavarian secondary school reforms. The goal was to prepare students for their academic studies as well as their professional life (Staatsinstitut für Schulqualität und Bildungsforschung, 2011). It is part of the eleventh school year. The class works on a large project in small groups. Each group is responsible for a specific task in the scope of the project.

The project seminar was designed to teach social, personal and professional skills. *Social skills* include reliability, respectful behavior with team mates and giving appreciative feedback. *Personal skills* target at an awareness of one's own strengths, weaknesses and interests as well as planning one's future after high school graduation. *Professional skills* include methodological aspects such as researching, interviewing and presentation techniques as well as knowledge in a specific field.

Each project seminar is bound to a school subject. Teachers have to come up with ideas for projects that fit their subject. In our case, an English teacher at the 'Gymnasium Tutzing' near Munich set up a project, in which her class had to dub an English movie. The class was split into different teams. Some teams did the translation, others were responsible for organizational tasks such as finding a sound studio and native speakers who do the dubbing.

### 7.3.2 Concept, Design and Implementation

Several months before the project seminar started, the developers of the group mirror met with the teacher. The teacher explained the structure of project seminar sessions and her motivation for using a guidance system. There were some features of the original guidance system, which the teacher wanted to keep. First, avatars should be used to represent the quality of a learner's contributions. The teacher specifically requested abstract metaphors instead of lifelike characters. For instance, she would prefer negative feedback that is visualized by a tree without leaves to a dying human-like character. A second feature that the teacher found useful was interruptive warnings that are sent when fundamental rules are broken. Similar to the lightning that appears when group members cut each other off in the middle of a sentence, a warning could appear when students disrespect each other.

However, a lot of information was still missing at this stage. First, this was the first year the project seminar took place, so there was little experience with it. Second, the teacher did not

---

<sup>3</sup> The practical part of this work was conducted together with Tim Kugler (2011) and Fabian Winter (2011) under the supervision of the thesis author.



yet know the participants of the seminar or how many would attend. Therefore, it was not clear whether the students would be disrespectful at all. If not, warnings about disrespectful behavior would be needless. The same goes for the qualitative feedback. It was impossible to foresee where problems would occur and where the learners needed support.

To deal with this problem the following approach was agreed on: In the first seminar sessions the groups were recorded on video, which were then analyzed by a team of educators and educational psychology students to identify areas of improvement. Based on these findings a collaboration script was developed, which describes the ideal collaborative process to the students. In the end, the developers created a group mirror representation that fits to the collaboration script.

### *A Collaboration Script for Collaborative Translation Sessions*

The groups consisted of three to four group members. The recordings showed that the collaboration was already fairly good without prior instructions. The groups translated sentence by sentence. One learner held the sheet with the English text and read each sentence out loud. Another team member was in charge of the dictionary and looked up words when necessary. A third learner wrote down the translation. Therefore, the groups automatically established a useful role distribution. However, three main problems were identified across all groups to different degrees. First, the roles were switched rarely. Second, the translation was often done by the learner who had the best English skills (for instance, students who had lived in English speaking countries). Consequently, not all team members could equally improve their English skills. Third, some sentences were discussed for a long time because one or more words were misunderstood. This could be avoided if ambiguous words would always be looked up in the dictionary and all possible meanings would be read out loud.

To overcome these deficits the team of educators and educational psychology students defined a collaboration script. It consists of three roles (reader, writer and consultant) who had different tasks across five phases:

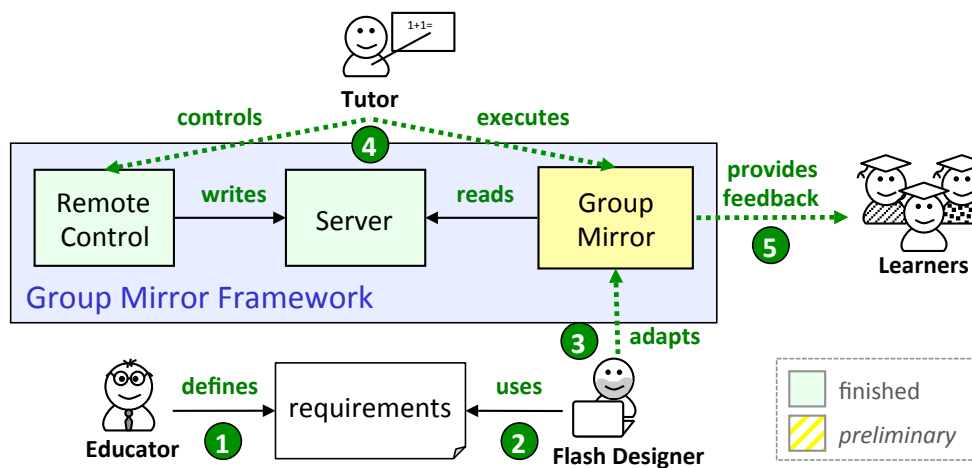
1. **Reading:** The *reader* reads the sentence out loud.
2. **Clarifying:** The '*consultant*' (i.e. person responsible for the dictionary) looks up ambiguous words and reads all possible meanings out loud.
3. **Translation:** Each learner (silently) translates the sentence.
4. **Integration:** Different translations are discussed and the best one is agreed on.
5. **Writing:** The *writer* writes down the translation.

The roles are switched regularly so that all students reach the same learning effects.

### Group Mirror Framework

Due to our approach of first analyzing the students' unsupported collaboration, the design of the group mirror was defined fairly late in the project. Until the educational experts of our team had defined the collaboration script, the programmers could not implement the user interface. In the meantime, a group mirror framework was implemented, which allows to flexibly configure certain group mirror functionalities instead of implementing the entire group mirror from scratch once the user interface is defined.

The architecture (see Figure 7.9) is an extension of the group mirror presented in section 7.2. It consists of a PHP server, which accepts data (e.g. a qualitative assessment of a learner's contributions) from the remote control. The group mirror reads the data and adapts the visualization accordingly. The clients (remote control and group mirror) are implemented with Adobe Flash and ActionScript.



**Figure 7.9:** Group mirror framework architecture.

The group mirror framework supports the automatic configuration of three group mirror components. First, avatars can be used to provide feedback to one specific learner (e.g. the qualitative feedback provided by the trees). Second, warnings can be sent to the entire group (e.g. a lightning that warns in case of interruptions). Third, a timer can be used to indicate how much time should be spent on a specific task (e.g. the sun moving from one side of the display to the other).

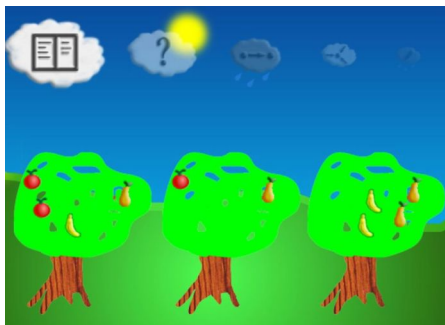
To flexibly configure these components, a startup-configuration screen is shown on the remote control, allowing the tutor to enter the number of group members and a time. Based on this information, the group mirror visualization adapts the number of avatars and distributes them within a defined area (e.g. on the lawn). Furthermore, the speed of the time-giving group mirror component is adapted.

Flash Designers can adapt the visualizations by exchanging or redesigning the according MovieClips. For instance, the avatar MovieClip has several FrameLabels, which define its different stages (e.g. five tree images from many to no leaves). By redesigning these five stages the

appearance of the avatars can be changed without rewriting the ActionScript code. Similarly, the background image as well as the timer and warning components can easily be adapted. If more functionality is required, the ActionScript code needs to be adapted as well.

### *Design Phase*

Once the collaboration script was defined, the goal was to create a group mirror visualization, which guides the learners through this script. Two design alternatives were created, both of which guide the group through the five phases and reflect how many times each learner performed each task. The first visualization was a scenery with trees and clouds similar to the previous guidance system (see Figure 7.11, left). However, the meaning of the trees and clouds changed. The five clouds represent the five consecutive phases. The cloud representing the current phase is highlighted. When the phase's task is performed very well the cloud is white, if not it is grey and rainy. For instance, if the group does not even consider looking up ambiguous words, the second cloud becomes dark. The trees reflect the role distribution. Each tree represents one learner. The trees carry fruits, each of which represents a role. For example, if a learner performs the task of the reader an apple is added to his tree, if he looks up words (consultant) a pear is added and if he takes the role of a writer a banana is added. Consequently, the learner's goal is to build a tree with different kinds of fruit.



**Figure 7.10:** Design alternative 1: Scenery with trees and clouds.



**Figure 7.11:** Design alternative 2: Cocktail bar.

A second design was created where the goal of adding different elements is more coherent. In the cocktail bar metaphor (Figure 7.11, right) the different roles are represented by different cocktail ingredients. Each learner is represented by a glass, which is filled up by performing the different role-specific tasks. The five phases are represented by five wall paintings.

The two design alternatives were presented and discussed with the students of the project seminar. All students favored the cocktail metaphor - not only because of the more consistent metaphor, but also because the cocktail bar scenario better fit their age.

### 7.3.3 Method

A user study was conducted to investigate the following research questions:

RQ 1: To what extent do the different feedback forms (tutor vs. guidance system) affect the acceptance of the collaboration script?

RQ 2: To what extent do the different feedback forms (tutor vs. guidance system) guide the learners through the collaboration script?

#### *Participants and Design*

Twelve students enrolled in the project seminar. However, only the two translation teams participated in the study. Each team consisted of four students (N=8). Five of them were 17, the other three 16 years old. Six participants were female.

The project seminar was weekly on Thursday mornings. The study took place in two consecutive sessions. In the first session feedback on the compliance to the collaboration script was provided by a tutor (*tutor condition*). In the second session feedback was provided by the group mirror using the cocktail bar representation (*group mirror condition*).

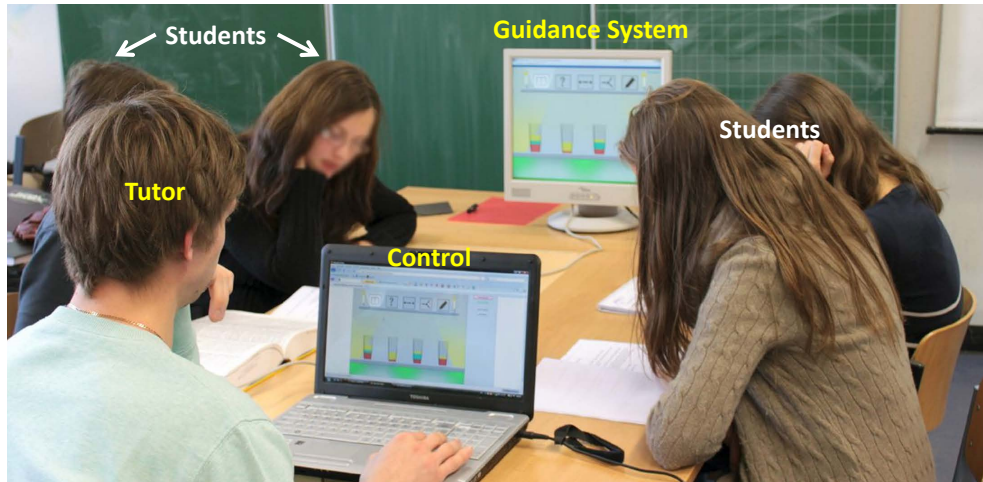
#### *Learning Task and Procedure*

The study took place in the regular classroom. For each group three desks were pushed together to build a large conference table. The students sat on opposite sides of the table (cf. Figure 7.12), a tutor sitting between them. The group mirror was located opposite of the tutor in the last condition.

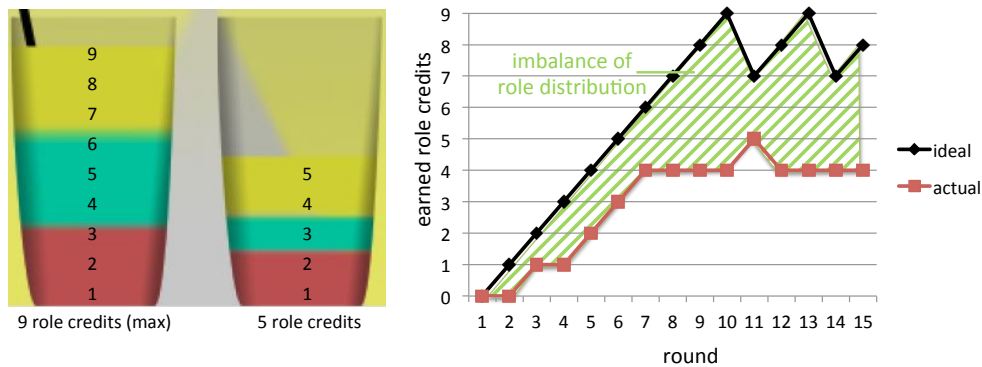
In the beginning of each session the collaboration script was explained and the students were asked to collaborate accordingly. Furthermore, a printout with symbols of the five consecutive phases was handed out in the tutor condition. This was done to eliminate the possibility that the only advantage of the group mirror is a representation of the task sequence. Afterwards the groups translated a part of the movie script. At the end of each session, questionnaires were handed out.

#### *Measures*

In terms of quantitative data the *imbalance of role distribution* was measured. It refers to the visual feedback provided by the cocktail bar representation. The maximum of cocktail levels that learners can reach is nine, three for each of the three roles. In the following these levels are called 'role credits'. The graph on the right of Figure 7.13 shows the ideal development of a learner's role credits over time (upper black line). The red line below represents an actual development in which the learner did not get as many role credits as possible. The area between these two lines is a negative indicator for the balance of role distribution. To measure the imbalance of role



**Figure 7.12:** Classroom setting in the group mirror condition.



**Figure 7.13:** Computation of imbalance of role distribution.

distribution, this area is computed per learner. Then the arithmetic mean is calculated to get the balance of role distribution for the whole group. If  $n$  is the number of group members and  $m$  the number of rounds over time, the formula for the imbalance of role distribution is:

$$\text{imbalance of role distribution} = \frac{1}{n} \sum_{k=1}^n \int_0^m (\text{credits}_{max} - \text{credits}_{actual}) \quad (7.1)$$

In addition questionnaires were handed out after each condition containing the same set of items. Each item consisted of a statement and a 5-point Likert scale from 1 (completely disagree) to 5 (completely agree). First, the students' acceptance of the collaboration script was inquired. The *acceptance of role changes* was measured with four items (e.g. 'Rotating roles lead to a more balanced collaboration'). The reliability of the scale was above Cronbach's  $\alpha=.9$  for each condition. The *acceptance of the task sequence* was measured with another set of four items

such as ‘The task sequence leads to a better collaboration’, (Cronbach’s  $\alpha > .8$ ). Second, the subjective adherence to the collaboration script was measured. For the subjective *adherence to the role changes* there were three items, for instance ‘I took each role equally often’ (Cronbach’s  $\alpha > .8$ ). The subjective *adherence to the task sequence* was measured with three items such as ‘I followed the proposed task sequence’ (Cronbach’s  $\alpha > .8$ ). Furthermore, the subjective *influence of the feedback on the role distribution* was measured with four items, for example ‘The feedback helped me to choose roles in a balanced way’ (Cronbach’s  $\alpha > .9$ ). The subjective *influence on the task sequence* was measured with four items such as ‘The feedback helped me to conform to the proposed task sequence’ (Cronbach’s  $\alpha > .9$ ). In addition, participants were asked to compare the feedback types in a final questionnaire. They were asked which feedback they preferred, which feedback they found less interruptive and which feedback they found more likely to improve their learning.

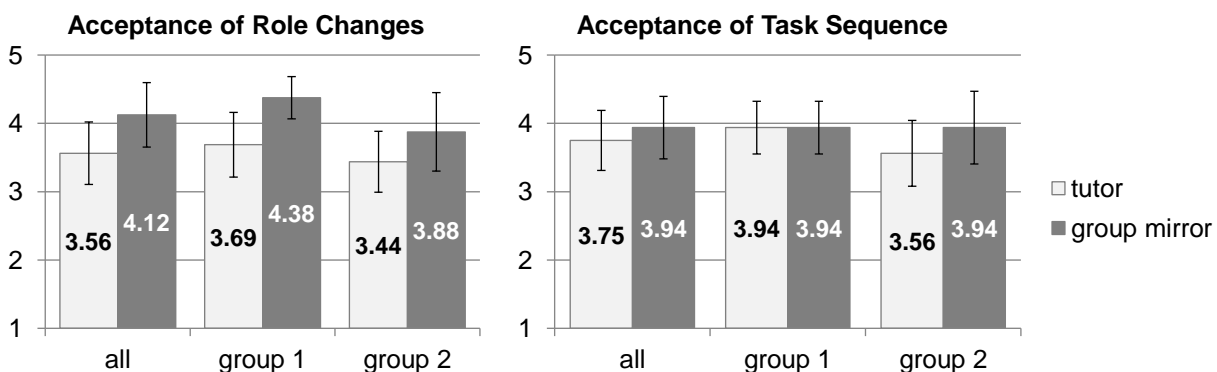
### Statistical Tests

Due to the small number of participants no statistical tests were applied. The main focus of the study was to get qualitative feedback.

## 7.3.4 Results

### Acceptance of Collaboration Script

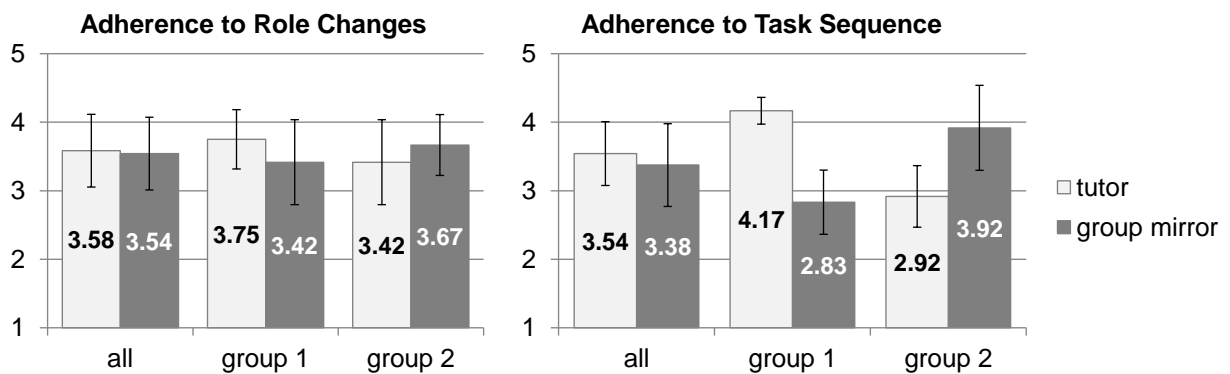
Figure 7.14 shows the acceptance of role changes and the task sequence. The *acceptance of role changes* tended to be higher in the group mirror condition ( $M=4.12$ ,  $SD=.94$ ) than in the tutor condition ( $M=3.56$ ,  $SD=.91$ ). This trend is visible in both groups. For the *acceptance of the task sequence* the trend was much smaller. The mean answer was  $M=3.94$  ( $SD=.91$ ) for the group mirror and  $M=3.75$  ( $SD=.88$ ) for the tutor. This tendency can be attributed to group 2. Group 1 rated both conditions equally.



**Figure 7.14:** Acceptance of collaboration script components on a 5-point Likert scale from 5 (high) to 1 (low).

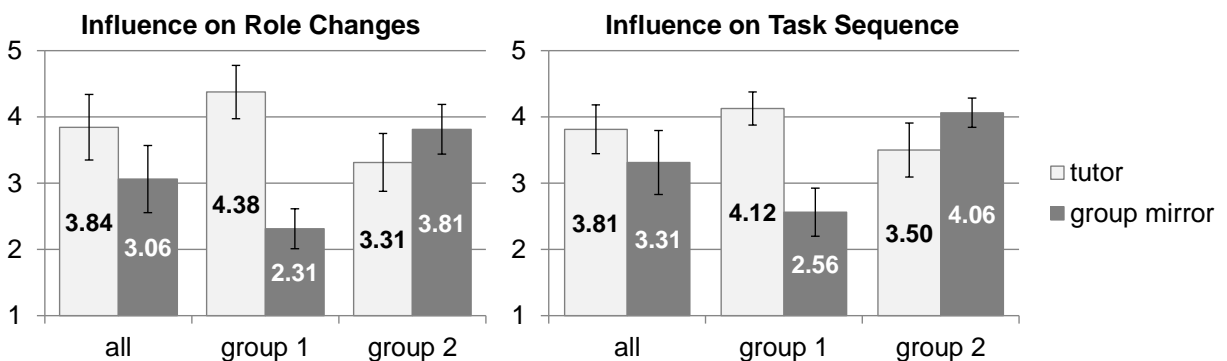
### Adherence to Collaboration Script

The subjective adherence to the collaboration script components are shown in Figure 7.15. The type of feedback did not seem to affect the subjective *adherence to the role changes*. Overall the mean answer was 3.58 (SD=1.06) for feedback from the tutor and 3.54 (SD=1.06) for the group mirror. Group 1 rated the tutor condition slightly higher (M=3.75, SD=.87) than the group mirror (M=3.42, SD=1.24). In group 2 there was the opposite tendency (M=3.42, SD=1.24 for the tutor and M=3.67, SD=.89 for the group mirror).



**Figure 7.15:** Subjective adherence to the collaboration script components on a 5-point Likert scale from 5 (high) to 1 (low).

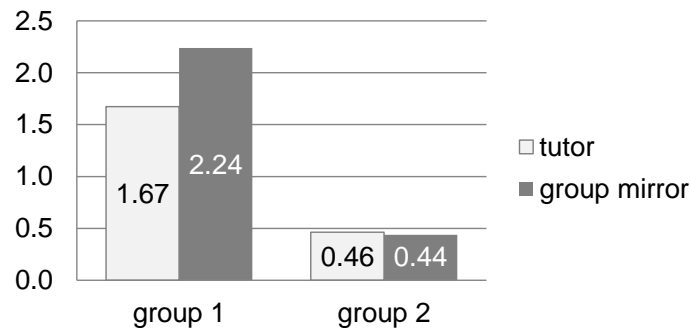
Regarding the subjective *adherence of the task sequence* the answers are fairly balanced across groups (M=3.54, SD=.93 for the tutor and M=3.38, SD=1.21 for the group mirror, see Figure 7.16). However, group 1 clearly thinks they better complied with the task sequence in the tutor condition (M=4.17, SD=.39) than in the group mirror condition (M=2.83, SD=.94), while group 2 thought the opposite (M=2.92, SD=.90 for the tutor and M=3.92, SD=1.24 for the group mirror).



**Figure 7.16:** Subjective influence of feedback on the adherence to the collaboration script components on a 5-point Likert scale from 5 (high) to 1 (low).

Another set of items measured the overall influence of the feedback (see Figure 7.16). Group 1 rated the influence of the tutor higher than the influence of the group mirror. This regards both the influence on role changes ( $M=4.38$ ,  $SD=.81$  for the tutor vs.  $M=2.31$ ,  $SD=.60$  for the group mirror) as well as on the task sequence ( $M=4.12$ ,  $SD=.50$  for the tutor vs.  $M=2.56$ ,  $SD=.73$  for the group mirror). In group 2 there was the opposite trend, although less strong. The influence on role changes was rated with  $M=3.31$  ( $SD=.87$ ) for the tutor and  $3.81$  ( $SD=.75$ ) for the group mirror. The influence on the task sequence was  $M=3.50$  ( $SD=.82$ ) for the tutor and  $4.06$  ( $SD=.44$ ) for the group mirror.

To get an objective assessment of the adherence to the role changes was measured. In group 1 the *imbalance of role distribution* tended to be higher in the group mirror condition (see Figure 7.17). Group 2 performed equally in both conditions.



**Figure 7.17:** Imbalance of role distribution.

### Popularity

In the final questionnaire the students were asked which type of feedback they preferred. Six students preferred the group mirror, two favored the tutor condition. Furthermore, the participants were asked which type of feedback was less interruptive. Seven students thought the group mirror was less interruptive, only one student chose the tutor. Furthermore, five students thought the group mirror is more likely to improve their learning, two chose the tutor, one abstained.

### 7.3.5 Summary and Discussion

Due to the small sample size no statistical analysis was done. First, the acceptance of the collaboration script tended to be higher in the group mirror condition. However, there was no trend, which indicated that the higher acceptance could lead to a better objective adherence to the collaboration script in the group mirror condition. Regarding the subjective adherence, one group considered it higher in the group mirror condition, the other one in the tutor condition.

There are several factors that could explain these inconclusive tendencies. First and foremost, the sample size was very small. Second, the acceptance of the collaboration script was problematic.



While students and teacher were involved in the iterative design process of the group mirror representation, their opinion was not obtained during the development of the collaboration script to the same degree. The collaboration script was developed on the basis of video recordings, which researchers of the learning sciences analyzed to identify deficits in the collaborative learning processes. The teacher and the students put both components of the collaboration script in question. Regarding the task sequence the overall opinion was that forcing all group members to translate the sentence individually is a waste of time and energy. The same goes for pre-emptively looking up words in the dictionary. These steps of the script were perceived as rather unnatural, even by the teacher. The second component was the role rotation. Students and teacher generally considered a balanced role distribution as fair and beneficial for learning. However, switching roles after each sentence was considered as too often. Consequently, the high-frequency role rotation was not very well accepted either. Therefore, a major lesson that we learned in the study, was that students and teacher not only need to be involved in the design of the group mirror but also in the development of the educational goals and methods. It is not advisable to place theory-based scripts into classrooms without prior validation as they may need adaptations for specific educational situations. Finally, classroom situations are much more complex and harder to control than laboratories. There are many side factors that interfere, for example the relationship among the students, between students and teacher or the students' individual goals (e.g. grades). Consequently, it is in general difficult to run controlled experiments in classrooms.

To sum up, the study could not show benefits of the group mirror. Despite the lack of quantitative results, qualitative feedback was promising. The majority of the students preferred the group mirror over the tutor condition. Many students mentioned that they found it less interruptive and thought it was more likely to have a positive effect on the learning process. Also, the teacher liked the system very much (even though she did not fully approve of the underlying collaboration script) and showed interest in using a similar system in her eighth grade as well. This general feedback indicates that group mirrors could be suitable tools to support small group collaboration in classrooms.

## 7.4 Lessons Learned

Group mirrors can have positive effects on collaborative processes. Previous work has demonstrated various ways in which group mirrors affect collaboration. This chapter specifically investigates how group mirrors can be used to guide learners through collaboration scripts. Two group mirrors were presented, which provided ambient feedback to co-located groups during *scripted* collaborative processes. A key finding of our studies is the potential of metaphoric group mirror representations. In the first (laboratory) study learners improved their contributions significantly faster when the feedback was shown in the form of a metaphoric representation compared to a diagram. In the project seminar the teacher also preferred a metaphoric representation from the beginning on.

Furthermore, in the second study, the group mirror was compared to tutors. While the group mirror provides continuous feedback without interrupting the collaborative process, the tutor

provided feedback rarely but with more emphasis. Consequently, both ways have different qualities. Which one is more appropriate obviously depends on the type of task and feedback. In addition, our experience in the class showed that different learners reacted differently to the two feedback types. For instance, one group rated their subjective adherence to the proposed script higher in the tutor condition, the other group thought they better conformed to the group mirror feedback. This suggests that the appropriateness of the feedback type also depends on the individual learners.

Finally, the experience in the classroom provided some interesting insights on the acceptance of collaboration scripts outside of laboratory settings. In the various laboratory studies that were presented in the course of this thesis participants rarely expressed concerns or doubt with regard to the (scripted) learning tasks. However, in the classroom, teachers *and* students questioned certain aspects of the collaboration script. For example, the phase in which learners had to clarify the possible meanings of all words before translating the sentence was considered artificial and unnecessary. There are two possible explanations why the participants were more critical in the classroom than in the laboratory settings. First, the classroom is the students' familiar environment in which they are (a) more self-confident and (b) better able to judge the usefulness of a script than in unknown laboratory settings. Another explanation could be that a *self-made* collaboration script was used, which was specifically developed for this class and not evaluated prior to its use. On the contrary, all collaboration scripts used in the other laboratory studies were well-established and their effectiveness had been proven in previous studies. In all likelihood it is a combination of both factors. Consequently, there are two more lessons that we learned in the classroom study. First, collaboration scripts need to be evaluated *before* their computational support is evaluated. Second, findings from laboratory studies need to be reverified in classroom settings.



## CONCLUSION AND FUTURE WORK



# Chapter 8

---

## Conclusion and Future Work

Collaboration scripts are great instruments to study the effects of user interfaces on computer-supported collaborative learning. There is a variety of collaboration scripts which verifiably further learning outcomes. In this thesis collaboration scripts were used as definitions of successful learning scenarios, which were then studied from a user interface perspective. Effects of different user interface technologies on co-located group learning processes were empirically explored in several studies. Taking an exploratory approach these studies cover a selection of learning scenarios (argumentation, knowledge sharing, reciprocal teaching, ...), different display configurations (tabletop, ambient wall displays, multi-display environments, ...) and different styles of guidance (implicit/explicit as well as varying levels of strictness).

### 8.1 Summary of Findings

At the beginning of the thesis two overall research questions were raised: (RQ I) Which types of displays or workspaces are suitable in which scripted collaborative learning scenarios and (RQ II) how should user interfaces be designed to guide learners through collaboration scripts. In the following the main findings on both questions are summarized.

#### *The Role of Private and Shared Workspaces*

Shareable user interfaces such as interactive wall or tabletop displays are helpful tools for face-to-face collaboration as they provide a joint workspace for presenting, exchanging and arranging (learning) content. In particular the benefits of tabletop displays are widely discussed in related work. However, several studies that were conducted in the scope of this thesis showed that it is worthwhile to integrate private workspaces in co-located collaborative learning applications. An example is our experience with the *ArgueTable*, an application that supports face-to-face argumentation based on collaboration scripts. While the initial decision was to use a tabletop display, the iterative design process led to an alternative display environment consisting of laptops and

a wall display. This combination of private and shared workspaces had a positive effect on the number of sequenced arguments, the increase of which was a major goal during the design process. In addition, learners were less reluctant to modify their arguments when they were provided with private workspaces (i.e. laptops) compared to shared tabletop displays. Furthermore, learners appreciated private workspaces, in particular reticent learners and learners who did not know their partners beforehand. This desire for an own private workspace may be explained by the competitive task in the ArgueTable/ArgueWall scripts. In a different learning scenario (Structured Academic Controversies), where the two learners acted in concert, the majority of learners did not wish for a private workspace.

Another project that demonstrates the value of private workspaces is *Note&Share*, an application that facilitates face-to-face knowledge sharing processes. It consists of a combination of laptops and a shared wall display. In addition, a multi-display reaching technique allows learners to share information by transferring it from their laptop to the wall display with simple drag-and-drop gestures. This combination of private and shared workspaces facilitated a process with two phases: In the individual preparation phase learners summarize the knowledge they can contribute without being influenced by others. Using their laptops they create keywords that represent their knowledge. During the second phase (the knowledge sharing process) the same keywords are sent to the wall display where a moderator arranges them in a mindmap. During this phase the keywords remind each individual about their knowledge. That way negative effects such as the omission of unshared knowledge can be mitigated. Two studies confirmed that learners shared knowledge more effectively when using Note&Share compared to using pen-and-paper or a whiteboard, which are traditionally used to support knowledge sharing processes. Also, the participants were more confident that no misunderstandings occurred when using Note&Share. It seems that the combination of private and shared workspaces affords a behavior pattern that positively affects knowledge sharing processes. In other words, the display environment helped to implicitly guide the learners' activities.

The examples of ArgueWall and Note&Share showed clear advantages of integrating private workspaces in collaborative learning applications. Other studies showed that the combination of laptops and wall displays can be *as effective* as tabletop displays. For instance, Ludo is a tabletop application, which assigns roles and activities among group members. That way learners are guided through collaboration scripts to enhance the learning process. A comparable setup with laptops and a beamer projection was shown to be as effective in terms of role and task distribution. Furthermore, the awareness of the learning partners' tasks and progress was not significantly lower. In addition, common problems with tabletops such as text input, ergonomic difficulties and text orientation in around-the-table seating arrangements can be avoided with combinations of laptops and wall displays. The significance of these tabletop issues has been underestimated in several other projects as well, for instance in the ArgueTable and SACTable projects. A key observation in these studies is that the participation on both the interactive wall display and the tabletop displays was equally balanced in character. This finding is contrary to the results of several previous studies (e.g. Rogers and Lindley, 2004 and Everitt et al., 2006) in which the participation was more balanced on tabletop displays than on interactive wall displays. The most

likely explanation is the establishing of the individual preparation phase which equipped the learners for a more balanced participation in the collaborative phase.

These main findings can be boiled down to two lessons that we learned throughout the projects. First, while shareable user interfaces are without doubt indispensable in most collaborative scenarios, there are many reasons for integrating private workspaces as well. For instance, they can help to shape the learners' activities, they are often appreciated by learners and they may encourage learners to interact in a less constrained way. The second key finding is that benefits of tabletop displays that are frequently mentioned in the literature were shown to be less important than expected. At the same time we underestimated issues on tabletop displays such as text input and orientation. Consequently, against our expectations tabletop displays did not seem to be the superior display setting in several projects. Based on these findings we argue that the use of tablespots for collaborative learning needs to be handled with care, at least when the learning involves considerable amounts of text.

### *Designing the User Interface to Guide Learners*

With regard to the question of how to guide learners, several types of guidance were explored in the thesis. First, there is implicit guidance through the choice of display environment, as shown above using the example of Note&Share. Another form of implicit guidance can be accomplished by providing each learner with a wizard-style application on a personal device: The graphical user interface of each personal device is designed as a sequence of screens, one for each task. On the first screen the learner only sees the information that is required to fulfil the first task. Once the task is finished, the next screen is shown for the subsequent task and so on. That way learners are guided through a task sequence. This form of guidance is applied by the applications developed with the XSS Framework, for instance the M.U.R.D.E.R. application. It is considered as particularly strict because learners are left with little or no freedom regarding the fulfillment of tasks.

In addition to these implicit guidance mechanisms, there are explicit ways of guidance, for example through *Guiding Widgets*, which assign roles and tasks to learners. Two projects that explored the use of such Guiding Widgets are Task Tokens and Ludo. Task Tokens are graphical representations of learning tasks that are moved across the table similar to gaming pieces in a board game. That way learners are guided through a sequence of effective learning activities in a playful manner. In a user study different levels of guidance were compared. The results show that automatically passing tokens to learners caused a significantly better adherence to both the task sequence and to the assignment of tasks to learners. Moreover, learners preferred the automatic assignment to less strict forms of guidance which leave the assignment of tasks to the learners. A follow-up project was Ludo, which extends the functionality of the Task Tokens by distributing roles: For each learning activity (e.g. questioning), different roles are assigned to the group members (e.g. teacher, examinee and recorder). Again, learners preferred the strict, automatic assignment of roles and tasks. While it is not surprising that the strict guidance leads to a better adherence to the collaboration script, it is interesting that automatic role and task distribution

by the learning application is not considered to be too restrictive and is actually appreciated by learners.

Another way of explicit guidance is dedicated mirroring or guidance systems. These systems monitor and reflect the group's behavior and/or give advice. In one study, two different visualizations were compared in a Wizard of Oz experiment: a diagram and a metaphorical representation of the same information. The latter consisted of a pictorial scene in which the weather and the condition of trees represented the quality of contributions. The results show that (1) the group reacted to the qualitative feedback and (2) the group's reaction to the metaphor was significantly faster than to the diagram. A similar system was designed, implemented and used in a classroom in a seminar at a German highschool over a period of several months.

Many of the presented guidance mechanisms are not mutually exclusive - they can complement each other. For instance, Guiding Widgets could be used to assign roles and tasks to different learners while a group mirror provides feedback on how well these tasks are performed. In the same way group mirrors can be combined with basically any form of guidance. Another possibility is the combination of a wizard-style application (implicit) and Guiding Widgets (explicit): The wizard-style application guides each learner through a task sequence but there is no representation of the learning partners' tasks or roles. By enhancing a wizard-style application with Guiding Widgets on a shared (wall or tabletop) display, an awareness of the partners' roles could be established.

## 8.2 Discussion and Future Studies

The exploratory approach, which was taken to pursue the research question, allowed us to experiment with a fairly wide range of possible solutions. At the same time, the approach has serious limitations due to (1) the design of the conducted studies, (2) the chosen display types / user interfaces and (3) the collaboration scripts. These limitations as well as their implications for future studies are discussed in the following.

### *Study designs*

Most studies were conducted in laboratory settings. An exception is the last study, which took place in a highschool seminar. This study indicates that teachers who implement scripted collaborative learning technologies in their classrooms run into difficulties that cannot be taken into account in laboratory studies. On the other hand it is also possible that additional positive effects evolve in the classroom. For instance, in the Note&Share laboratory study participants were eager to be well prepared for the discussion across all conditions due to the unfamiliar laboratory setting. In real classrooms Note&Share could increase student's efforts during the individual preparation because they are expected to prepare keywords, which they later share with the group. Future work needs to verify whether our findings can be transferred into classrooms and whether there are positive or negative side effects. Despite this limitation we consider the conducted lab-



oratory studies valuable for identifying, preselecting and experimenting with different tools and user interfaces, not all of which can be tested in time-consuming classroom studies.

A related issue is the selected target group. Most of our laboratory studies were conducted with a high ratio of college and university students. Exceptions were the Note&Share follow-up study (section 5.3), which was at least partly tested with highschool students, and the classroom study (section 7.3). Nonetheless, all participants were 16 or older. How other target groups react to the proposed display configurations and guidance mechanisms needs to be shown in further studies.

Yet another problem related to the exploratory approach is the small number of participants. While this approach allowed us to explore a rich set of ideas, it is difficult to identify all effects and to draw cause and effect conclusion. This problem is aggravated by the fact that most studies conducted in the scope of this thesis were semi-experimental. For instance, several display environments that were compared varied in several aspects at the same time (e.g. text input method or size and orientation of the shared display). Such studies do not allow narrowing down the cause of observed effects, which makes it hard to produce generalizable results - independent of the number of participants. However, our attempt to identify a single display factor as the cause of an effect in several follow-up studies of the ArgueTable was not successful either. In these studies artificial prototypes were used to specifically examine one display factor at a time. Although the studies produced further insights, they did not lead to generalizable display recommendations. The resulting study design dilemma was discussed in detail in section 4.3.5. The bottom line of the discussion is that the attempt to isolate individual display characteristics as causes of effects on collaborative processes is not advisable. On the other hand, effects of *entire display environments* could be generalizable. For instance, Note&Share provides private and shared workspaces and a multi-display reaching technique, which were shown to facilitate a process that furthers knowledge sharing processes. It is likely that effects of Note&Share - as well as Guiding Widgets or group mirrors - apply to a wider range of cases. The generalizability of these effects needs to be verified in future studies with larger numbers of participants, different age groups as well as outside the laboratory environment.

### *Display Types / User Interfaces*

In addition to different study designs, further work is required on variations of the user interfaces. In the course of this thesis the same display types were used repeatedly. For instance, laptops were usually chosen as representative of private workspaces whereas tabletop or interactive wall displays were used as shared workspace. However, there are other types of displays that could not be investigated in the scope of this thesis. To name just one example: tablet computers could be used instead of laptops. As tablets are getting more and more popular it would be interesting to examine whether, for example, Note&Share works equally well if the laptops are replaced by tablet computers. There are several things that would have to be considered. Depending on the seating arrangement tablet computers can either be private or personal workspaces. Since it is difficult to control whether participants place the tablet computer on their lap or on the table as well as how they hold the tablet computer (cf. Figure 4.12), they are less suitable for controlled experiments on privacy and awareness aspects. Furthermore, the different input and

output technologies need to be kept in mind. For instance, whether a physical or an on-screen keyboard is used for text input can have dramatic consequences on text-based learning tasks (cf. section 4.1). For these reasons it is necessary to verify whether it is possible to successfully move concepts to different display types in each individual case.

### *Collaboration Scripts*

Using collaboration scripts as the educational foundation of this work provides a valuable instrument for our research purposes. The key benefit of our approach is that the scripts are detailed descriptions of collaborative learning scenarios, which have been evaluated in experiments and have shown to positively affect learning outcomes. However, there are two downsides to the script-based approach. First, the applicability of collaboration scripts in the classroom is not always unproblematic. This issue has been reported in previous research (cf. Dillenbourg, 2002). Difficulties in applying collaboration scripts in the classroom have also been experienced in the scope of this work (cf. section 7.3). Second, while there are some common principles such as peer questioning and peer reviewing, other script-based learning scenarios are very specific (e.g. the construction of single arguments, cf. section 4.1). The more specific the learning scenario, the more questionable it is whether or not findings also apply to other learning scenarios. This issue seems particularly important as our own findings suggest that even basic aspects, such as the desire for one's own private workspace, depend on the learning task (cf. section 4.5). Further studies need to examine whether our results are task-specific or generalizable.

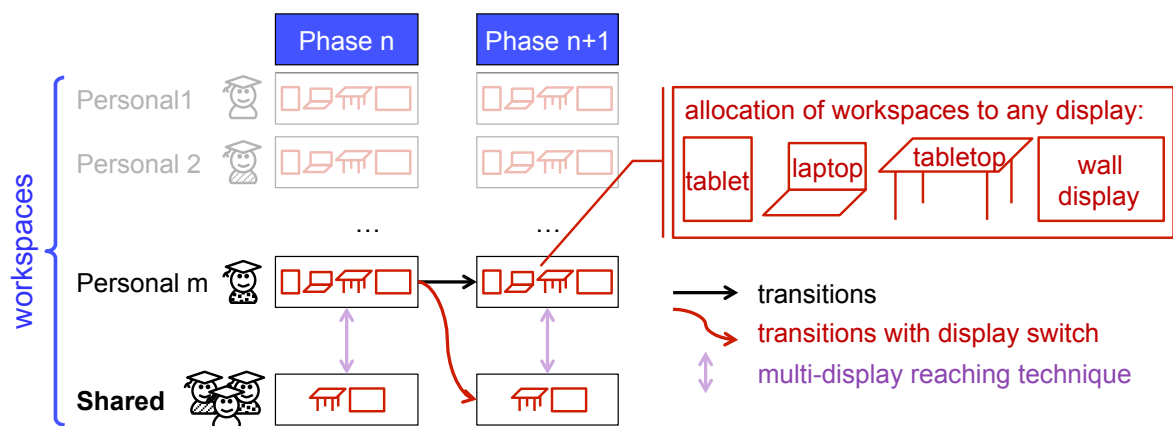
Besides repeating experiments with other learning scenarios and collaboration scripts, there is also room for future work regarding the measurement of learning outcomes. In our studies we measured to what degree learners stuck to collaboration scripts which have been shown to have positive effects on learning outcomes. Consequently, we did not directly measure learning outcomes. This approach saved us a lot of effort, which was invested in the exploration of a wider range of solutions. However, the technologies we used to guide learners in accordance with collaboration scripts could interfere with their effectiveness on learning outcomes in various ways. Therefore, in the long run it is necessary to directly measure the effects of scripted collaborative learning applications on learning outcomes.

To sum up, we consider the exploratory approach to be a good starting point for the investigation of our research question but at the same time it entails a multitude of follow-up studies that need to be done in the future.

## 8.3 Future Work on the XSS Framework

To support the development of scripted collaborative learning applications, the XSS Framework was built and presented in chapter 3. The framework supports automatic code generation from the information specified in a collaboration script. Over the course of this thesis different extensions of the XSS Framework have been proposed. At this point these ideas as well as other findings of our studies will be integrated into a comprehensive vision of a future framework. The current

version of the framework is restricted to the use of personal devices. An extension for shareable user interfaces was already presented in section 3.5.2. However, our studies indicate that the *combination* of shared and private workspaces is useful in many scenarios. Consequently, we need a framework that allows user interface designers to flexibly allocate workspaces to different kinds of displays.



**Figure 8.1:** Extension of the XSS Framework towards flexible integration and management of private and shared workspaces.

Figure 8.1 (a modification of Figure 3.9) illustrates the three additional functionalities that are needed to flexibly integrate and manage combinations of private and shared workspaces. First, in addition to personal or private workspaces there are *shared* workspaces (bottom row in Figure 8.1). For instance, an additional shared wall or tabletop display is provided as in Note&Share or the Task Token Table. Second, it should be possible to flexibly allocate workspaces to different display types. In the *current* framework phase transitions always happen on the same device: When a learner has finished one phase there is a transition to a new screen on the same display for the next phase. (In Figure 8.1 this is illustrated by the straight horizontal arrow, which leads from the left-hand column (phase n) to the right-hand column (phase n+1). These arrows already existed in Figure 3.9.) An *extended* framework should allow phase transitions with display switches. For example, after an individual phase that is done on laptops (personal workspaces), learners switch to a tabletop display for the subsequent phase (as in the ArgueWall or SACWall applications). In Figure 8.1 this is illustrated by the curved arrow, which leads from one row (personal workspace) to another row (shared workspace). Finally, multi-display reaching techniques should be integrated to allow learners to send content from one workspace to another within a phase, usually from private workspaces to the shared workspace and vice versa. In Figure 8.1 this corresponds to the vertical arrows.

The implementation of these features is not a trivial endeavor. In particular the allocation of workspaces to different displays requires careful consideration. For instance, different display types have different input and output technologies. If direct touch is used instead of an indirect pointing device, problems of reach, occlusion or text input have to be considered. Such issues have been addressed in section 3.5.2 in more detail. Furthermore, it may be useful to allocate sev-

eral personal workspaces *and* one shared workspace to tabletop displays. If the tabletop display is intended for face-to-face seating arrangements, it is unclear how the layout of the workspace composition should look. The simplest solution (which was suggested in section 3.5.2) is to directly transfer the private workspaces to personal workspaces which are placed along the edges of the tabletop display (cf. Figure 3.18). This would require minor or no changes to the workspace layout. Even in this supposedly simple solution there are possible problems to be considered, for example overlapping personal and shared workspaces or mismatching display resolutions. A more advanced solution would optimize the personal workspace layout by adapting it to the characteristics of the tabletop display. That way the above-mentioned problems (reach, occlusion and the like) could be accounted for. In addition, the tabletop layout could be designed according to the territories which are naturally formed on tabletops (cf. Scott et al., 2004). In this case the personal workspaces might be implicitly defined by their proximity to the learner, similar as in the DiamondSpin (Shen et al., 2004), making visual workspace borders unnecessary.

In a radically advanced version this optimization could be done automatically by the framework: User interface designers and educators could define the workspace functionality and the framework would adjust the workspace layout to the display characteristics. This would even allow teachers and tutors the flexible allocation of workspaces to different display types during runtime, which would simplify classroom orchestration considerably.

In addition to an extension for private and shared workspaces, different guidance mechanisms could be integrated into the framework. This suggestion was already addressed in section 6.3. As the sequence of tasks and roles is defined in the collaboration script, it is possible to automatically generate Guiding Widgets that guide learners through a sequence of tasks or assign roles to different learners. Such widgets could be offered as modules to user interface designers who can add them to a workspace when needed. Again, a radically advanced framework would allow teachers to add such guiding widgets during runtime when needed and allocate them according to the display type that is available.

## 8.4 Final Remarks

For many years, a learning environment was designed as a single piece of technology to be used in schools (e.g. desktops, laptops, mobile phones, ...). Now, we have entered times where a learning environment refers to an ecosystem of digital devices. This induces an evolution of research in which the questions are no longer restricted to one piece of technology but need to address the interplay of multiple artifacts of a different nature (laptops, tabletops, wall display,...). In our work, we even tried to integrate non-digital elements that populate classroom ecosystems (e.g. paper sheets). Therefore, we go beyond the question of how to use private, personal and shared workspaces and how implicit and explicit guidance mechanisms can be applied in these spaces. It is one of the first theses that address the ecosystem at the systemic level, i.e. the relation between multiple types of digital artifacts in the classroom environment. We hope that our exploratory approach provides a good foundation for future research and inspires others to continue the investigation of these important questions.

# **Appendices**



## A Example of an IMS-LD Document

```

<?xml version="1.0" encoding="UTF-8"?>
<imscp:manifest xmlns:imscp="..."
  <imscp:organizations>
    <imsld:learning-design identifier="ExampleScript" level="A" uri="" version="1.0">
      <imsld:title>ExampleScript</imsld:title>
      <imsld:components>
        <imsld:roles>
          <imsld:learner identifier="RO-reviewer" min-persons="1" max-persons="1"/>
          <imsld:learner identifier="RO-listener" min-persons="1" max-persons="1"/>
        </imsld:roles>

        <imsld:activities>
          <imsld:learning-activity identifier="LA-read" parameters="0">
            <imsld:activity-description>
              <imsld:title>Read</imsld:title>
              <imsld:item identifierref="RES-text01" parameters="">
                <imsld:title>Read the text carefully.</imsld:title>
              </imsld:item>
            </imsld:activity-description>
          </imsld:learning-activity>
          <imsld:learning-activity identifier="LA-choose-keywords">
            <imsld:activity-description>
              <imsld:title>Choose keywords</imsld:title>
              <imsld:item identifierref="RES-LA-read">
                <imsld:title>Mark the five most important keywords in the text.</imsld:title>
              </imsld:item>
            </imsld:activity-description>
          </imsld:learning-activity>
          <imsld:learning-activity identifier="LA-summarize">
            <imsld:activity-description>
              <imsld:title>Summarize</imsld:title>
              <imsld:item>
                <imsld:title>Summarize the content. [...]</imsld:title>
              </imsld:item>
            </imsld:activity-description>
          </imsld:learning-activity>
          <imsld:learning-activity identifier="LA-listen">
            <imsld:activity-description>
              <imsld:title>Listen</imsld:title>
              <imsld:item>
                <imsld:title>Listen to your partner's summary.</imsld:title>
              </imsld:item>
            </imsld:activity-description>
          </imsld:learning-activity>
        </imsld:activities>
      </imsld:components>
    </imsld:learning-design>
  </imscp:organizations>
</imscp:manifest>

```

(continued on next page)

```

<imsld:learning-activity identifier="LA-listen">
  <imsld:activity-description>
    <imsld:title>Listen</imsld:title>
    <imsld:item>
      <imsld:title>Listen to your partner's summary.</imsld:title>
    </imsld:item>
  </imsld:activity-description>
</imsld:learning-activity>
...

<imsld:activity-structure identifier="AS-reviewer" structure-type="sequence">
  <imsld:title/>
  <imsld:learning-activity-ref ref="LA-read"/>
  <imsld:learning-activity-ref ref="LA-choose-keywords"/>
  <imsld:learning-activity-ref ref="LA-summarize"/>
  ...
</imsld:activity-structure>
<imsld:activity-structure identifier="AS-listener" structure-type="sequence">
  <imsld:title/>
  <imsld:learning-activity-ref ref="LA-read"/>
  <imsld:learning-activity-ref ref="LA-choose-keywords"/>
  <imsld:learning-activity-ref ref="LA-listen"/>
  ...
</imsld:activity-structure>
</imsld:activities>
</imsld:components>

<imsld:method>
  <imsld:play identifier="PLAY-ExampleScript">
    <imsld:act identifier="ACT-ExampleScript">
      <imsld:role-part>
        <imsld:role-ref ref="RO-reviewer"/>
        <imsld:activity-structure-ref ref="AS-reviewer"/>
      </imsld:role-part>
      <imsld:role-part>
        <imsld:role-ref ref="RO-listener"/>
        <imsld:activity-structure-ref ref="AS-listener"/>
      </imsld:role-part>
      <imsld:complete-act>
        <imsld:when-role-part-completed ref="LA-closing"/>
      </imsld:complete-act>
    </imsld:act>
  </imsld:play>
</imsld:method>
</imsld:learning-design>
</imsdp:organizations>

<imsdp:resources>
  <imsdp:resource identifier="RES-text01" type="webcontent" href="resources/text01.rtf"/>
  <imsdp:resource identifier="RES-closing" type="webcontent" href="resources/schlusswort.rtf"/>
  <imsdp:resource identifier="RES-LA-read" type="response"/>
</imsdp:resources>

</imsdp:manifest>

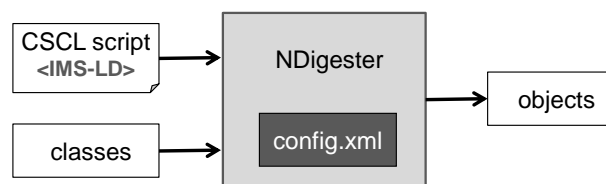
```



## B XSS Framework Implementation

The XSS Framework is implemented in C#. It consists of two solutions: applications and libraries. The solution *libraries* contains basic classes and functionalities, which are compiled into .dll library files. There are three projects:

1. **Tools** offers classes for the client-server communication such as Client, Server and Message
2. **CSCL** contains basic classes for scripted collaborative learning processes such as Roles, Sequences, Activities, Groups, etc. Figure B.3 shows a simplified UML diagram of the core classes.
3. **NDigester**<sup>1</sup> is responsible for the XML parsing. NDigester is an open-source tool, which provides full control over XML documents processing. Instead of hard coding the analyses and object generation processes, NDigester deserializes XML documents and creates objects from pre-defined classes based on a configuration file (see Figure B.2).

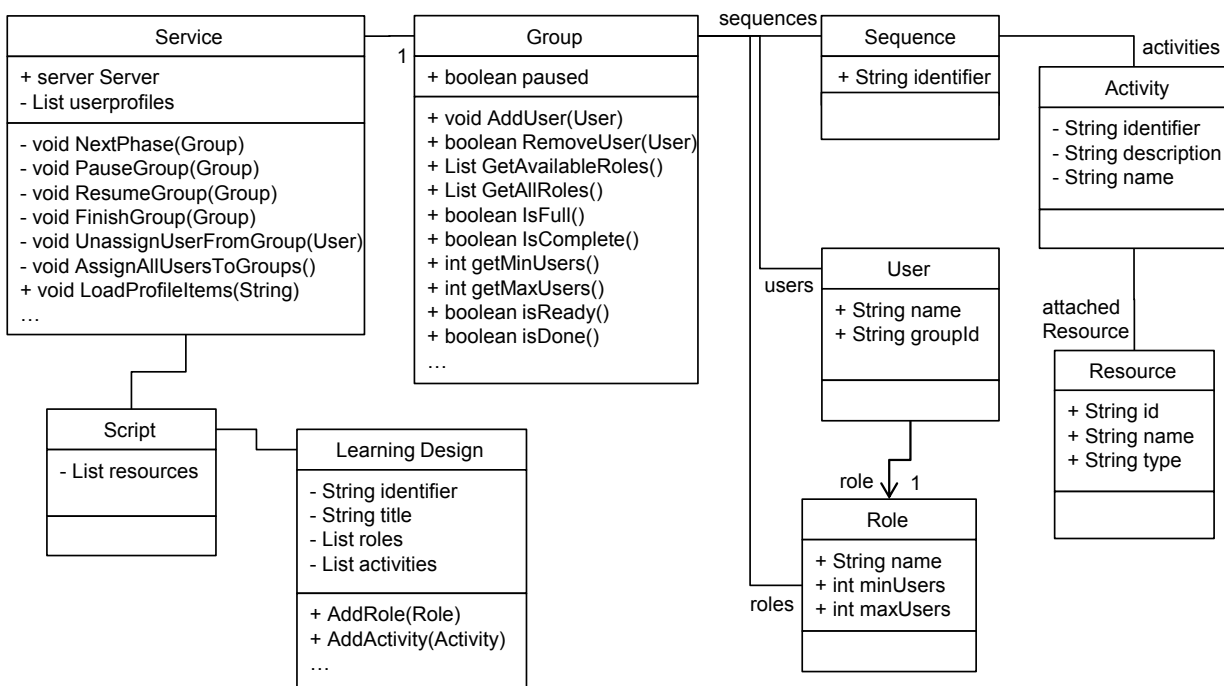


**Figure B.2:** XML Parsing with the NDigester.

The solution *applications* contains the five projects, which are compiled into executable files (e.g. ‘CSCL Service.exe’). The five projects are:

1. **CSCL Service** provides a graphical user interface, which allows to import a collaboration script (IMS-LD file) and start/stop the server. It uses the above-mentioned libraries.
2. **CSCL Client** provides the client user interface that is used by the learners. It also uses the above-mentioned libraries.
3. **CSCL Tutor** provides the graphical user interface for the tutor (cf. ‘Group Administration Tool’ in section 3.3.4). It also uses the above-mentioned libraries.
4. **CSCL MultiClient** corresponds to the ‘Multi Client’ Tool presented in section 3.3.4). It allows to run multiple clients in one window for simulation purposes.
5. **CSCL Simplestarter** is a tool, which guides users through the steps that are necessary to run applications (cf. ‘Simple Starter’ in section 3.3.4).

<sup>1</sup> <http://sourceforge.net/projects/ndigester/>



**Figure B.3:** Simplified UML Diagram containing the core classes of the project CSCL.

## C Parsing IMS-LD Documents with the NDigester

```
namespace CSCL {
    public class Service {
        public Server server = new Server();

        // list of groups
        private List<Group> groups = new List<Group>();

        // list of users
        private List<User> users = new List<User>();

        // collaboration script
        public CSCL.CSCLObject.Script script;

        ...

        // constructor: generate a new service for a given IMS-LD file
        public Service(String imsldPath)
        {
            // create a new digester
            Digester digester = new Digester();

            // load the configuration file and configure the digester
            String configPath = Environment.CurrentDirectory + "\\XMLConfig\\config.xml";
            digester.Configure(configPath);
            ...
            XmlNode configNode = Data.GetDocumentRootNode(imsldPath);
            ...
            script = (CSCL.CSCLObject.Script)digester.Parse(configNode);
            ...

            // create a new list ...
            List<Role> roleList = new List<Role>();

            // ... run through all roles defined in the script
            foreach (CSCLObject.Role role in script.LearningDesign.Roles)
            {
                // create a new role object and add it to the list
                roleList.Add(new Role(role));
            }

            ...
        }
    }
}
```

**Figure C.4:** Code listing: Parsing IMS-LD Documents with the NDigester.

## D Finalization of the Client

```
namespace CSCL_Client {
    public partial class Main : Form {
        private Client client;

        public Viewer viewer;
        public UserInfoForm userInfoForm;
        public String username;
        private List<UserProfileItem> profileItems;

        public Main() {
            ...
            viewer = new Viewer();
            ...
        }

        // connect to server
        public void Connect(Object sender, EventArgs arguments) { ... }

        // disconnect from server
        public void Disconnect() { ... }

        ...

        // enter new phase and adapt the viewer
        private void NextPhase(Activity activity) {
            // set title and task description
            viewer.taskTitle.Text = activity.name;
            viewer.taskDescription.Text = activity.description;

            // show input resource
            viewer.editor.Rtf = activity.attachedResource.rtfText;

            ...

            // enable ready button
            viewer.readyButton.Enabled = true;
        }

        // end current phase and send output resource to the server
        private void Ready(Object sender, EventArgs arguments) {
            // disable ready button
            viewer.readyButton.Enabled = false;

            // create output resource and send it to the server
            Resource data = new Resource("RES-"+currentId,"response",viewer.editor.Text);
            client.Send(data);
        }

        ...
    }
}
```

**Figure D.5:** Code listing: Finalization of the Client.

## E Transcription of Note&Share Discussions

In the following we outline the discussion of two groups. Both groups had the same order of conditions, starting with Note&Share, then pen-and-paper and finally the whiteboard. That way the groups were not only comparable, it was also ruled out that a better performance in the Note&Share condition was caused by learning effects. Group 1 performed fairly well across all conditions. The number of shared arguments was 7 for pen-and-paper, 8 for the whiteboard and 10 for Note&Share. Group 2 performed only well in the Note&Share condition (10 shared arguments), fairly well with pen-and-paper (7 arguments) and but only shared two arguments with whiteboard.

In Group 1 the moderator starts the discussion in the Note&Share condition by repeating the topic:

Moderator: *“Our topic is ‘The pensionable age should be raised.’ Ben?”*

Ben: *“There is a new pension bill to protect our social state [...]. The number of employees, who pay for the pensioners, is decreasing. Also increasing health and ability suggest that people can work longer and might even get bored if they retire early.”*

Moderator: *“Ok, what is your position, Katie?”*

Katie: *“I am against the new law, because even nowadays many people cannot hang on until the age of 55 because companies do not create basic conditions regarding health, qualification and potential. Furthermore, young employees are preferred while older people have difficulties in finding new jobs when losing their old ones. As mentioned before, it would be wiser if the number of employees would rise, which would fill pension funds and pensioners would benefit from this. Also, if pensionable age rises there are too many employers and the employment market is overburdened.”*

Moderator: *“And you, Susan?”* [...]

In the pen-and-paper condition the course of events is very similar. The moderator only steers the conversation by asking all analysts for their opinions, one after another: *“What is your opinion on this subject”, “Do you agree with that”, “What is your opinion”*. In the whiteboard condition the moderator starts with the question *“Our last topic is whether to tighten the smoking ban in Bavaria. Who supports this?”* instead of asking a specific person, but again only asks for opinions.

In Group 2 the moderator behaves the same way when using Note&Share. The moderator reads the task description:

Moderator: *“Can I hear arguments against the nuclear power phase-out from Mary, please?”*

Mary: *“Experts predicted an increasing power consumption, which cannot be covered without nuclear power. Therefore, one would have to buy expensive power. [Sends a keyword] There is less emission of carbon dioxide. [Sends a keyword] And a nuclear power phase-out would cause a huge energy supply shortfall.”*

Moderator: *“Ok, now we will hear Steve’s opinion.”*

Steve: *“I am for and against power plants. On the one hand there is no permanent disposal of radioactive waste and building new plants is very expensive. Also, it is important that the current plants exist, otherwise there would be energy supply shortfalls. Furthermore, buying energy from neighbors is very expensive and alternatives (solar and wind) energy are not always available.”*

Moderator: *“That was it?”* [...]

In the pen-and-paper condition, the moderator does more than steering the conversation.

Moderator: *“We discuss the smoking ban, which might be tightened in Bavaria. Each analyst has an opinion, which we want to discuss and maybe find a compromise in the end. Okay. Steve? Pro or con?”*

Steve: *“Pro. In my opinion the smoking ban should be tightened because [...] children, adolescents and all other non-smokers are exposed to a danger and are also encouraged to smoke. Studies showed that in countries with strict smoke bans there are 40% less smokers.”*

Moderator: *“So you are in favor of the smoke ban because non-smokers are polluted by passive smoking, and that the number of smokers is increased because smokers set an example, and that children do not have the freedom to decide whether they want to smoke or not are still polluted.”*

Steve: *“And also employees in the gastronomy are polluted at their workplace. [...]”*

Moderator: *“Okay, Mary?”* [...]

In the whiteboard condition:

Moderator: *“The topic is whether the pensionable age should be raised. Analyst 1, are you pro or con?”*

Mary: *“I am pro.”*

Moderator: *“Okay, Mary is in favor of raising the pensionable age.”*

Mary: *“Correct. I think if the pensionable age is raised, this means a higher work load for the employees but it is important to protect the social state.”*

The moderator suggests keywords and writes them down on the whiteboard.

Moderator: *“higher work load, protection of social state”*

Mary: *“Also, I think by raising the pensionable age, the health and potential of the general public is increased and that nobody will be harmed.”*

The moderator suggest a keyword and writes it down.

Moderator: *“positive for people’s health. Another argument for it?”*

Mary: *“No, not at the moment.”*

Moderator: *“Okay. Analyst Steve? You are con?”* [...]

## BIBLIOGRAPHY

- Abowd, G. D., 1999. Classroom 2000: An Experiment with the Instrumentation of a Living Educational Environment. *IBM Systems Journal* 38 (4), 508–530.
- Ackermann, F., Eden, C., 1994. Issues in Computer and Non-Computer Supported GDSSs. *Decision Support Systems* 12 (4-5), 381–390.
- Adamic, L. A., Zhang, J., Bakshy, E., Ackerman, M. S., 2008. Knowledge Sharing and Yahoo Answers: Everyone Knows Something. In: *Proceeding of the 17th international conference on World Wide Web (WWW '08)*. ACM, New York, NY, USA, pp. 665–674.
- Alavi, M., Tiwana, A., October 2002. Knowledge Integration in Virtual Teams: The Potential Role of KMS. *J. Am. Soc. Inf. Sci. Technol.* 53, 1029–1037.
- Alj, Y., 2009. An Ambient Display as Group Mirror. Project Thesis at the Ludwig-Maximilians-Universität München.
- Andriessen, J., Baker, M., Suthers, D. (Eds.), 2003. *Arguing to Learn: Confronting Cognitions in Computer-Supported Collaborative Learning Environments*. Kluwer Academic Publishers. Amsterdam. Netherlands.
- Andriessen, J., Coirier, P., 1999. Foundations of Argumentative Text Processing. In: Rijlaarsdam, G., Esperet, E. (Eds.), *Studies in Writing*. Amsterdam University Press, Amsterdam, NL.
- Aronson, E., Blaney, N., Sikes, J., Stephan, G., Snapp, M., 1978. *The Jigsaw Classroom*. Sage Publication.
- Ayala, G., 2007. Scripting Collaborative Learning in Agent-Based Systems. In: Fischer, F., Ingo, K., Mandl, H., Haake, J. M. (Eds.), *Scripting computer-supported collaborative learning: cognitive, computational and educational perspectives*. Springer, New York, pp. 101–115.
- Bachour, K., Kaplan, F., Dillenbourg, P., 2008. Reflect: An Interactive Table for Regulating Face-to-Face Collaborative Learning. In: *Times of Convergence. Technologies Across Learning Contexts, 2008*. Springer, Berlin, Germany, pp. 39–48.
- Bachour, K., Kaplan, F., Dillenbourg, P., July 2010. An Interactive Table for Supporting Participation Balance in Face-to-Face Collaborative Learning. *IEEE Trans. Learn. Technol.* 3, 203–213.

- Baghaei, N., Mitrovic, A., Irwin, W., 2007. Supporting Collaborative Learning and Problem-Solving in a Constraint-Based CSCL Environment for UML Class Diagrams. *The International Journal of Computer-Supported Collaborative Learning (CSCL '11)* 2 (2-3), 159–190.
- Baker, M., 2003. Computer-Mediated Argumentative Interactions for the Co-Elaboration of Scientific Notions. In: Andriessen, J., Baker, M., Suthers, D. (Eds.), *Arguing to Learn: Confronting Cognitions in Computer-Supported Collaborative Learning environments*. Kluwer Academic Publishers. Amsterdam. Netherlands, pp. 47–78.
- Baker, M., Lund, K., 1997. Promoting Reflective Interactions in a CSCL Environment. *Journal of Computer Assisted Learning* 13 (3), 175–193.
- Barron, M., Moher, T., Maharry, J., 2006. RoomBugs: Simulating Insect Infestations in Elementary Classrooms. In: *CHI '06 extended abstracts on Human factors in computing systems*. CHI EA '06. ACM, New York, NY, USA, pp. 315–320.
- Berger, A., Moretti, R., Chastonay, P., Dillenbourg, P., Bchir, A., Baddoura, R., Bengondo, C., Scherly, D., Ndumbe, P., Farah, P., Kayser, B., 2001. Teaching Community Health by Exploiting International Socio-Cultural and Economical Differences. In: Dillenbourg, P., Eurelings, A., Hakkarainen, K. (Eds.), *Proceedings of the first European Conference on Computer Supported Collaborative Learning*. Maastricht, NL, pp. 97–105.
- Bergstrom, T., Karahalios, K., 2007a. Conversation Clock: Visualizing Audio Patterns in Co-Located Groups. In: *Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS '07)*. IEEE Computer Society, Washington, DC, USA, p. 78.
- Bergstrom, T., Karahalios, K., 2007b. Seeing More: Visualizing Audio Cues. In: *Proceedings of the 11th IFIP TC 13 international conference on Human-computer interaction - Volume Part II. INTERACT'07*. Springer-Verlag, Berlin, Heidelberg, pp. 29–42.
- Bergstrom, T., Karahalios, K., 2007c. Visualizing Colocated Conversation Feedback. In: *Proceedings of the IEEE Tabletop '07*.
- Biehl, J. T., Baker, W. T., Bailey, B. P., Tan, D. S., Inkpen, K. M., Czerwinski, M., 2008. IMPROMPTU: A New Interaction Framework for Supporting Collaboration in Multiple Display Environments and Its Field Evaluation for Co-Located Software Development. In: *Proceeding of Human factors in computing systems (CHI '08)*. ACM, New York, NY, USA, pp. 939–948.
- Böhm, S., 2009. A Knowledge Sharing Tool for Collaborative Group Sessions. Project Thesis at the Ludwig-Maximilians-Universität München.
- Böhm, S., 2010. Evaluating Different User Interface Concepts and Guidance Mechanisms for the ArgueTable. Diploma Thesis at the Ludwig-Maximilians-Universität München.
- Borchers, J. O., 2001. *A Pattern Approach to Interaction Design*. Wiley, Chichester, UK.



- Brodbeck, F. C., Kerschreiter, R., Mojzisch, A., Schulz-Hardt, S., 2007. Group Decision Making Under Conditions of Distributed Knowledge: The Information Asymmetries Model. *The Academy of Management Review* 32 (2), 459–479.
- Brown, B., MacColl, I., Chalmers, M., Galani, A., Randell, C., Steed, A., 2003. Lessons From the Lighthouse: Collaboration in a Shared Mixed Reality System. In: Proceedings of the SIGCHI conference on Human factors in computing systems. CHI '03. ACM, New York, NY, USA, pp. 577–584.
- Brown, S. C., 2003. Interactive Whiteboards in Education. TechLearn for Joint Information Systems Committee.
- Buisine, S., Besacier, G., Aoussat, A., Vernier, F., Jan. 2012. How Do Interactive Tabletop Systems Influence Collaboration? *Comput. Hum. Behav.* 28, 49–59.
- Burkhard, R. A., 2004. Learning from Architects: The Difference Between Knowledge Visualization and Information Visualization. *Information Visualisation, International Conference on* 0, 519–524.
- Calori, I. C., Divitini, M., 2009. Reflections on the Role of Technology in City-Wide Collaborative Learning. *International Journal of Interactive Mobile Technologies (IJIM)* 3 (2), 33–39.
- Cañas, A. J., Hill, G., Carff, R., Suri, N., Lott, J., Gómez, G., Eskridge, T. C., Arroyo, M., Carvajal, R., 2004. CmapTools: A Knowledge Modeling and Sharing Environment. In: Universidad Pública de Navarra. pp. 125–133.
- Cappelletti, A., Gelmini, G., Pianesi, F., Rossi, F., Zancanaro, M., 2004. Enforcing Cooperative Storytelling: First Studies. In: Proceedings of the IEEE International Conference on Advanced Learning Technologies. IEEE Computer Society, Washington, DC, USA, pp. 281–285.
- Carell, A., Herrmann, T., Kienle, A., Menold, N., 2005. Improving the Coordination of Collaborative Learning With Process Models. In: Proceedings of the 2005 conference on Computer support for collaborative learning (CSCL '05). International Society of the Learning Sciences, pp. 18–27.
- Clark, D., Sampson, V., Stegmann, K., Marttunen, M., Kollar, I., Janssen, J., Weinberger, A., 2010. Online Learning Environments, Scientific Argumentation, and 21st Century Skills. In: B. Ertl (Ed.), *E-Collaborative Knowledge Construction: Learning from Computer-Supported and Virtual Environments*. IGI Global: Hershey, pp. 1–39.
- Coffey, J., Hoffman, R., Cañas, A., Ford, K., 2002. A Concept Map-Based Knowledge Modeling Approach to Expert Knowledge Sharing. In: Proceedings of the IASTED International Conference on Information and Knowledge Sharing. pp. 212–217.
- Colley, J., Stead, G., 2004. Mobile Learning = Collaboration. In: *Mobile Learning anytime everywhere - A book of papers from MLEARN 2004*. pp. 57–58.

- Cross, R., Parker, A., Prusak, L., Borgatti, S. P., November 2001. Knowing What We Know: Supporting Knowledge Creation and Sharing in Social Networks. *Organizational Dynamics* 30 (2), 100–120.
- Dalziel, J., 2003. Implementing Learning Design: The Learning Activity Management System (LAMS). In: *Proceedings of the 20th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education (ASCILITE)*. pp. 593–596.
- Dansereau, D. F., Collins, K. W., McDonald, B. A., Holley, C. D., Garland, J. C., Diekhoff, G., 1979. Development and Evaluation of a Learning Strategy Training Program. *Journal of Educational Psychology* 71 (1), 64–73.
- de Velde, W. V., Häkkinen, P., Järvelä, S., Stegmann, K., 2004. Examples of CSCL Scripts Within an Organisational Framework. Tech. Rep. Deliverable 23.3.1., Ecole Polytechnique Fédérale de Lausanne.  
URL <http://craftwww.epfl.ch/mosil/deliverables/D23-3-1-AVEC.pdf>
- Demetriadis, S., Egerter, T., Hanisch, F., Fischer, F., 2011. Peer Review-Based Scripted Collaboration to Support Domain-Specific and Domain-General Knowledge Acquisition in Computer Science. *Computer Science Education* 21 (1), 29–56.
- Diehl, M., Stroebe, W., 1987. Productivity Loss in Brainstorming Groups: Toward the Solution of a Riddle. *Journal of Personality and Social Psychology* 53 (3), 497–509.
- Dietz, P., Leigh, D., 2001. DiamondTouch: A Multi-User Touch Technology. In: *Proceedings of the 14th annual ACM symposium on User interface software and technology*. UIST '01. ACM, New York, NY, USA, pp. 219–226.
- Dillenbourg, P., 2002. Over-Scripting CSCL: The Risk of Blending Collaborative Learning With Instructional Design. In: Kirschner, P. A. (Ed.), *Three worlds of CSCL: Can we support CSCL?* Heerlen: Open University of the Netherlands, pp. 61–91.
- Dillenbourg, P., 2005. Designing Biases that Augment Socio-Cognitive Interactions. In: Bromme, R., Hesse, F. W., Spada, H. (Eds.), *Barriers and Biases in Computer-Mediated Knowledge Communication*. Springer, US, pp. 497–501.
- Dillenbourg, P., Baker, M., Blaye, A., O'Malley, C., 1996. The Evolution of Research on Collaborative Learning. In: Spada, E., Reiman, P. (Eds.), *Learning in Humans and Machines: Towards an Interdisciplinary Learning Science*. Elsevier, pp. 189–211.
- Dillenbourg, P., Hong, F., March 2008. The Mechanics of CSCL Macro Scripts. *International Journal of Computer-Supported Collaborative Learning* 3 (1), 5–23.
- Dillenbourg, P., Hong, F., Brahm, T., for Innovations in Learning, U. S. G. S. C., 2009. *The Many Scripts Pedagogical Handbook: How to Build Scripts for Collaborative Learning?* Swiss Centre for Innovations in Learning (SCIL), University of St. Gallen.  
URL <http://books.google.de/books?id=1lPmZwEACAAJ>

- Dillenbourg, P., Jermann, P., 2007. Designing Integrative Scripts. In: F. Fischer, K. Ingo, H. M. . J. M. H. (Ed.), *Scripting computer-supported collaborative learning: Cognitive, computational and educational perspectives*. Springer, New York, pp. 276–301.
- DiMicco, J., Bender, W., 2007. Group reactions to visual feedback tools. In: de Kort, Y., IJsselsteijn, W., Midden, C., Eggen, B., Fogg, B. (Eds.), *Persuasive Technology*. Vol. 4744 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, pp. 132–143.
- DiMicco, J. M., Pandolfo, A., Bender, W., 2004. Influencing Group Participation With a Shared Display. In: *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*. ACM, New York, NY, USA, pp. 614–623.
- Dishner, E. K., Readence, J. E., 1977. A Systematic Procedure for Teaching Main Idea. *Reading World* 16 (4), 292–298.
- Dix, A., Finlay, J., Abowd, G., Beale, R., 2004. *Human-Computer Interaction*, 3rd Edition. Prentice-Hall.
- Doise, W., Mugny, G., 1984. *The Social Development of the Intellect*. Pergamon Press.
- Druin, A., Stewart, J., Proft, D., Bederson, B., Hollan, J., 1997. KidPad: A Design Collaboration Between Children, Technologists, and Educators. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. CHI '97. ACM, New York, NY, USA, pp. 463–470.
- Dufresne, R. J., Gerace, W. J., Leonard, W. J., Mestre, J. P., Wen, L., 1996. Classtalk: A Classroom Communication System for Active Learning. *Journal of Computing in Higher Education* 7 (2), 3–47.
- Echtler, F., Klinker, G., Butz, A., 2010. Towards a Unified Gesture Description Language. In: *Proceedings of the 13th International Conference on Humans and Computers*. HC '10. University of Aizu Press, Fukushima-ken, Japan, Japan, pp. 177–182.
- Edelson, D. C., Pea, R. D., Gomez, L. M., April 1996. The Collaboratory Notebook. *Commun. ACM* 39, 32–33.
- Espinosa, A., Cadiz, J., Rico-Gutierrez, L., Kraut, R., Scherlis, W., Lautenbacher, G., 2000. Coming to the Wrong Decision Quickly: Why Awareness Tools Must Be Matched With Appropriate Tasks. In: *CHI '00: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, New York, NY, USA, pp. 392–399.
- Everitt, K., Shen, C., Ryall, K., Forlines, C., 2006. MultiSpace: Enabling Electronic Document Micro-mobility in Table-Centric, Multi-Device Environments. In: *TABLETOP '06: Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems*. IEEE Computer Society, Washington, DC, USA, pp. 27–34.

- Falcão, T. P., Price, S., 2009. What Have You Done! The Role of 'Interference' in Tangible Environments for Supporting Collaborative Learning. In: Proceedings of the 9th international conference on Computer supported collaborative learning - Volume 1. CSCL'09. International Society of the Learning Sciences, pp. 325–334.
- Fischer, F., Kollar, I., Haake, J. M., Mandl, H., 2007. Perspectives on Collaboration Scripts. In: F. Fischer, K. Ingo, H. M. . J. M. H. (Ed.), Scripting computer-supported collaborative learning: Cognitive, computational and educational perspectives. Springer, New York, pp. 14–37.
- Fischer, F., Kollar, I., Stegmann, K., Wecker, C., in press. Towards a Script Theory of Guidance in Computer-Supported Collaborative Learning. Tech. rep.
- Gallupe, R. B., Bastianutti, L. M., Cooper, W. H., 1991. Unblocking Brainstorms. *Journal of Applied Psychology* 76 (1), 137–142.
- Gellersen, H., Fischer, C., Guinard, D., Gostner, R., Kortuem, G., Kray, C., Rukzio, E., Streng, S., 2009. Supporting Device Discovery and Spontaneous Interaction with Spatial References. *Personal and Ubiquitous Computing* 13, 255–264.
- Gibson, J., 1977. The Theory of Affordances. In: Shaw, R., Brandsford, J. (Eds.), *Perceiving, Acting, and Knowing*. Lawrence Erlbaum Associates, Hillsdale, New Jersey.
- Greenberg, S., Boyle, M., Laberge, J., Mar. 1999. PDAs and Shared Public Displays: Making Personal Information Public, and Public Information Personal. *Personal and Ubiquitous Computing* 3 (1), 54–64.
- Greenberg, S., Marquardt, N., Ballendat, T., Diaz-Marino, R., Wang, M., January 2011. Proxemic Interactions: The New Ubicomp? *interactions* 18, 42–50.
- Greenberg, S., Rounding, M., 2001. The Notification Collage: Posting Information to Public and Personal Displays. In: CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, New York, NY, USA, pp. 514–521.
- Griffiths, D., Beauvoir, P., Liber, O., Barrett-Baxendale, M., 2009. From Reload to ReCourse: Learning from IMS Learning Design Implementations. *Distance Education* 30 (2), 201–222.
- Guinard, D., Streng, S., Gellersen, H., May 2007. Extending Mobile Devices with Spatially Arranged Gateways to Pervasive Services. In: Workshop on Pervasive Mobile Interaction Devices (PERMID) in conjunction with Pervasive 2007, Toronto, Canada, May 2007.
- Guzdial, M., Turns, J., 2000. Effective Discussion through a Computer-Mediated Anchored Forum. *The Journal of the Learning Sciences* 9 (4), pp. 437–469.
- Ha, V., Inkpen, K. M., Whalen, T., Mandryk, R. L., 2006. Direct Intentions: The Effects of Input Devices on Collaboration around a Tabletop Display. In: Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems. IEEE Computer Society, Washington, DC, USA, pp. 177–184.

- Halbinger, M., 2009. ArgueTable: An Instrumented Environment for Argumentative Learning. Project Thesis at the Ludwig-Maximilians-Universität München.
- Hall, E. T., 1966. *The Hidden Dimension*, 1st Edition. Doubleday, Garden City, N.Y.
- Harrer, A., Malzahn, N., 2006. Bridging the Gap - Towards a Graphical Modeling Language for Learning Designs and Collaboration Scripts of Various Granularities. In: Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies. ICALT '06. IEEE Computer Society, Washington, DC, USA, pp. 296–300.
- Hascoët, M., 2003. Throwing Models for Large Displays. In: Group, B. H. (Ed.), *Human Computer Interaction, HCI'2003*. Bath, UK, pp. 77–108.
- Hattie, J., Timperley, H., 2007. The Power of Feedback. *Review of Educational Research* 77 (1), 81–112.
- Haué, J.-B., Dillenbourg, P., 2009. Do Fewer Laptops Make a Better Team? In: Pierre Dillenbourg, J. H., Cherubini, M. (Eds.), *Interactive Artifacts and Furniture Supporting Collaborative Work and Learning*. Springer, New York, pp. 15–34.
- Hernández-Leo, D., Asensio-Pérez, J. I., Dimitriadis, Y., Bote-Lorenzo, M. L., Jorrín-Abellán, I. M., Villasclaras-Fernández, E. D., 2005. Reusing IMS-LD Formalized Best Practices in Collaborative Learning Structuring. *Advanced Technology for Learning* 2 (3), 223–232.
- Hernández-Leo, D., Burgos, D., Tattersall, C., Koper, R., 2006. Representing CSCL Macro-Scripts Using IMS LD: Lessons Learned. Tech. rep., extended version of Hernández-Leo, D., Burgos, D., Tattersall, C., Koper, R. Representing Computer-Supported Collaborative Learning macro-scripts using IMS Learning Design Proceedings of the Second European Conference on Technology Enhanced Learning, CEUR Workshop Proceedings, EC-TEL'07, Crete, Greece, September 2007.  
URL [http://dspace.ou.nl/bitstream/1820/784/7/hernandezleo\\_etal\\_creation-scripts.pdf](http://dspace.ou.nl/bitstream/1820/784/7/hernandezleo_etal_creation-scripts.pdf)
- Hilliges, O., Terrenghi, L., Boring, S., Kim, D., Richter, H., Butz, A., 2007. Designing for Collaborative Creative Problem Solving. In: *C&C '07: Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition*. ACM, New York, NY, USA, pp. 137–146.
- Hinrichs, U., Hancock, M., Carpendale, S., Collins, C., 2007. Examination of Text-Entry Methods for Tabletop Displays. In: *Horizontal Interactive Human-Computer Systems (TABLETOP) 2007*. IEEE Computer Society, Los Alamitos, CA, USA, pp. 105–112.
- Hinrichs, U., Schmidt, H., Carpendale, S., 2008. EMDialog: Bringing Information Visualization into the Museum. *IEEE Transactions on Visualization and Computer Graphics* 14 (6), 1181–1188.

- Horn, M. S., Solovey, E. T., Crouser, R. J., Jacob, R. J., 2009. Comparing the Use of Tangible and Graphical Programming Languages for Informal Science Education. In: Proceedings of the 27th international conference on Human factors in computing systems. CHI '09. ACM, New York, NY, USA, pp. 975–984.
- Hornecker, E., Marshall, P., Dalton, N. S., Rogers, Y., 2008. Collaboration and Interference: Awareness with Mice or Touch Input. In: Proceedings of the 2008 ACM conference on Computer supported cooperative work. CSCW '08. ACM, New York, NY, USA, pp. 167–176.
- Hutchings, H. M., Pierce, J. S., 2006. Understanding the Whethers, Hows, and Whys of Divisible Interfaces. In: AVI '06: Proceedings of the working conference on Advanced visual interfaces. ACM, New York, NY, USA, pp. 274–277.
- Huysman, M., Wulf, V., 2006. IT to Support Knowledge Sharing in Communities, Towards a Social Capital Analysis. *Journal of Information Technology* 21, 40–51(12).
- Hymes, C. M., Olson, G. M., 1992. Unblocking Brainstorming Through the Use of a Simple Group Editor. In: CSCW '92: Proceedings of the 1992 ACM conference on Computer-supported cooperative work. ACM, New York, NY, USA, pp. 99–106.
- IMS Global Learning Consortium, 2003. IMS Learning Design Specification. Version 1.0. Final Specification. Retrieved on October 1st 2009 from <http://www.msglobal.org/specificationdownload.cfm>.
- Inkpen, K., Hawkey, K., Kellar, M., Mandryk, R., Parker, J., Reilly, D., Scott, S., Whalen, T., July 2005. Exploring Display Factors that Influence Co-located Collaboration: Angle, Size, Number, and User Arrangement. In: Proceedings of HCI International 2005. Lawrence Erlbaum Associates, pp. 22–27.
- Isenberg, P., Carpendale, S., November 2007. Interactive Tree Comparison for Co-located Collaborative Information Visualization. *IEEE Transactions on Visualization and Computer Graphics* 13, 1232–1239.
- Izadi, S., Brignull, H., Rodden, T., Rogers, Y., Underwood, M., 2003. Dynamo: A Public Interactive Surface Supporting the Cooperative Sharing and Exchange of Media. In: UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology. ACM, New York, NY, USA, pp. 159–168.
- Jain, M., Birnholtz, J., Cutrell, E., Balakrishnan, R., 2011. Exploring Display Techniques for Mobile Collaborative Learning in Developing Regions. In: Proceedings of the MobileHCI 2011. ACM, pp. 81–90.
- Jermann, P., Dillenbourg, P., 2003. Elaborating New Arguments Through a CSCL Script. In: Andriessen, J., Baker, M., Suthers, D. (Eds.), *Arguing to Learn: Confronting Cognitions in Computer-Supported Collaborative Learning environments*. Kluwer Academic Publishers. Amsterdam. Netherlands, pp. 205–226.

- Jermann, P., Soller, A., Lesgold, A., 2004. Computer Software Support for Collaborative Learning. In: Strijbos, J.-W., Kirschner, P., Martens, R. (Eds.), *What We Know About CSCL in Higher Education*. Kluwer, Amsterdam, NL, pp. 141–166.
- Jermann, P., Soller, A., Muehlenbrock, M., 2001. From Mirroring to Guiding: A Review of the State of the Art Technology for Supporting Collaborative Learning. In: *In Proceedings of the European conference on Computer Supported Collaborative Learning, EuroCSCL-2001*. IOS Press, pp. 324–331.
- Johansen, R., 1988. *GroupWare: Computer Support for Business Teams*. The Free Press, New York, NY, USA.
- Johanson, B., Fox, A., Winograd, T., 2002. The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms. *IEEE Pervasive Computing* 1 (2), 67–74.
- Johnson, D. W., Johnson, R. T., 1994. Constructive Conflict in the Schools. *Journal of Social Issues* 50 (1), 117–137.
- Kaplan, F., Lenh, S., Bachour, K., Kao, G. Y., Gault, C., Dillenbourg, P., 2009. Interpersonal Computers for Higher Education. In: Pierre Dillenbourg, J. H., Cherubini, M. (Eds.), *Interactive Artifacts and Furniture Supporting Collaborative Work and Learning*. Springer, New York, pp. 129–145.
- Kauko, J., Häkkinä, J., 2010. Shared-Screen Social Gaming with Portable Devices. In: *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services. MobileHCI '10*. ACM, New York, NY, USA, pp. 317–326.
- Kelley, J. F., 1983. An Empirical Methodology for Writing User-Friendly Natural Language Computer Applications. In: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems. CHI '83*. ACM, New York, NY, USA, pp. 193–196.
- Kim, T., Chang, A., Holland, L., Pentland, A. S., 2008. Meeting Mediator: Enhancing Group Collaboration Using Sociometric Feedback. In: *Proceedings of the 2008 ACM conference on Computer supported cooperative work. CSCW '08*. ACM, New York, NY, USA, pp. 457–466.
- King, A., 1997. Educational psychologist. ASK to THINK-TEL WHY: A Model of Transactive Peer Tutoring for Scaffolding Higher Level Complex Learning 32, 221–235.
- King, A., 2007. Scripting Collaborative Learning Processes: A Cognitive Perspective. In: F. Fischer, K. Ingo, H. M. . J. M. H. (Ed.), *Scripting computer-supported collaborative learning : cognitive, computational and educational perspectives*. Springer, New York, pp. 14–37.
- Kirschner, F., Paas, F., Kirschner, P. A., March 2009. Individual and Group-Based Learning from Complex Cognitive Tasks: Effects on Retention and Transfer Efficiency. *Computers in Human Behavior* 25, 306–314.

- Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., Häkkinen, P., Fischer, F., 2007. Specifying Computer-Supported Collaboration Scripts. *International Journal of Computer-Supported Collaborative Learning* 2 (2-3), 211–224.
- Koch, J., 2009. Computerunterstützung für das MURDER-Skript auf Basis eines generischen Frameworks. Diploma Thesis at the Ludwig-Maximilians-Universität München.
- Kollar, I., Fischer, F., Hesse, F., 2006. Collaboration Scripts - A Conceptual Analysis. *Educational Psychology Review* 18 (2), 159–185.
- Kollar, I., Fischer, F., Slotta, J. D., Dec. 2007. Internal and External Scripts in Computer-Supported Collaborative Inquiry Learning. *Learning and Instruction* 17 (6), 708–721.
- Koper, R., 2001. Modeling Units of Study from a Pedagogical Perspective: The Pedagogical Meta-Model Behind EML.  
URL <http://eml.ou.nl/introduction/docs/ped-metamodel.pdf>
- Koschmann, T., 2003. CSCL, Argumentation, and Deweyan Inquiry: Argumentation is Learning. In: Andriessen, J., Baker, M., Suthers, D. (Eds.), *Arguing to Learn: Confronting Cognitions in Computer-Supported Collaborative Learning environments*. Kluwer Academic Publishers. Amsterdam, Netherlands, pp. 1–25.
- Kruger, R., Carpendale, S., Scott, S. D., Greenberg, S., 2003. How People Use Orientation on Tables: Comprehension, Coordination and Communication. In: *Proceedings of the 2003 international ACM SIGGROUP conference on Supporting group work. GROUP '03*. ACM, New York, NY, USA, pp. 369–378.
- Kugler, T., 2011. Ein Framework für Feedback-Systeme zur Unterstützung von Team-Sitzungen in der gymnasialen Oberstufe. Bachelor Thesis at the Ludwig-Maximilians-Universität München.
- Kukulska-Hulme, A., May 2009. Will Mobile Learning Change Language Learning? *ReCALL* 21, 157–165.
- Leitão, S., 2000. The Potential of Argument in Knowledge Building. *Human Development* 43, 332–360.
- Liang, J., Wang, H., Huang, C., Chang, S., Chan, T., 2001. Highly Interactive Instruction Environment for Classroom - Integration of Wireless Testing System and Learning Information Management System. In: *Learning across the Ages - Looking Back and Looking Forwards*. p. 311.
- Liu, T.-C., Wang, H.-Y., Liang, J.-K., Chan, T.-W., Yang, J.-C., 2002. Applying Wireless Technologies to Build a Highly Interactive Learning Environment. In: *WMTE '02: Proceedings IEEE International Workshop on Wireless and Mobile Technologies in Education*. IEEE Computer Society, Washington, DC, USA, pp. 63–70.



- Looi, C.-K., Lin, C.-P., Liu, K.-P., July 2008. Group Scribbles to Support Knowledge Building in Jigsaw Method. *IEEE Transactions of Learning Technologies* 1 (3), 157–164.
- Lund, K., Molinari, G., Sèjourné, A., Baker, M., 2007. How Do Argumentation Diagrams Compare When Student Pairs Use Them as a Means for Debate or as a Tool for Representing Debate? *International Journal of Computer-Supported Collaborative Learning* 2, 273–295.
- Malzahn, N., Pokrandt, M., Hoppe, H. U., 2008. Extending a Learning Design Editor with a Monitoring Component. In: Chan, T.-W., Biswas, G., Chen, F.-C., Chen, S., Chou, C., Jacobson, M., Kinshuk, Keltt, F., Looi, C.-K., Mitrovic, T., Mizoguchi, R., Naakabayashi, K., Reimann, P., Suthers, D., Yang, S., Yang, J.-C. (Eds.), *Proceedings of the 16th International Conference on Computers in Education (ICCE 2008)*. Taipei, Taiwan, pp. 499–504.
- Mandryk, R. L., Scott, S. D., Inkpen, K. M., 2002. Display Factors Influencing Co-located Collaboration. *Work* 2, 137–138.
- Mankoff, J., Dey, A. K., Hsieh, G., Kientz, J., Lederer, S., Ames, M., 2003. Heuristic Evaluation of Ambient Displays. In: *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, New York, NY, USA, pp. 169–176.
- Marshall, P., 2007. Do Tangible Interfaces Enhance Learning? In: *Proceedings of the 1st international conference on Tangible and embedded interaction*. TEI '07. ACM, New York, NY, USA, pp. 163–170.
- Matzke, W., 2010. *TokenTable: Rollenbasiertes Gruppenlernen am interaktiven Tisch*. Project Thesis at the Ludwig-Maximilians-Universität München.
- McInnerney, J., 2004. Collaborative or Cooperative Learning? In: Roberts, T. (Ed.), *Online collaborative learning: Theory and practice*. Information Science Publishing, pp. 203–214.
- Metzner, M., 2010. *Die Rolle persönlicher Arbeitsbereiche in einer Lernumgebung für “strukturierte akademische Diskussionen”*. Project Thesis at the Ludwig-Maximilians-Universität München.
- Miao, Y., Hoeksema, K., Hoppe, H. U., Harrer, A., 2005. CSCL Scripts: Modelling Features and Potential Use. In: *CSCL '05: Proceedings of th 2005 conference on Computer support for collaborative learning*. International Society of the Learning Sciences, pp. 423–432.
- Mock, K., December 2004. Teaching with Tablet PC's. *J. Comput. Small Coll.* 20, 17–27.
- Moher, T., 2006. Embedded Phenomena: Supporting Science Learning with Classroom-Sized Distributed Simulations. In: *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, New York, NY, USA, pp. 691–700.
- Morris, M., Piper, A., Cassanego, A., Winograd, T., 2005. *Supporting Cooperative Language Learning: Issues in Interface Design for an Interactive Table*. Tech. rep., Stanford University Technical Report.

- Morris, M. R., Brush, A. J. B., Meyers, B. R., 2008. A Field Study of Knowledge Workers' Use of Interactive Horizontal Displays. In: Proceedings of the 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems (TABLETOP 2008). pp. 105–112.
- Morris, M. R., Cassanego, A., Paepcke, A., Winograd, T., Piper, A. M., Huang, A., September 2006a. Mediating Group Dynamics through Tabletop Interface Design. *IEEE Comput. Graph. Appl.* 26, 65–73.
- Morris, M. R., Paepcke, A., Winograd, T., Stamberger, J., 2006b. TeamTag: Exploring Centralized Versus Replicated Controls for Co-Located Tabletop Groupware. In: CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems. ACM, New York, NY, USA, pp. 1273–1282.
- Morris, M. R., Ryall, K., Shen, C., Forlines, C., Vernier, F., 2004. Beyond "Social Protocols": Multi-User Coordination Policies for Co-Located Groupware. In: Proceedings of the 2004 ACM conference on Computer supported cooperative work. CSCW '04. ACM, New York, NY, USA, pp. 262–265.
- Motlik, S., 2008. Mobile Learning in Developing Nations. *International Review of Research in Open and Distance Learning* 9 (2), 1–7.
- Mühlbauer, M., 2011. SACTable: Eine Tabletop-basierte Lernumgebung für “strukturierte akademische Diskussionen”. Bachelor Thesis at the Ludwig-Maximilians-Universität München.
- Müller-Tomfelde, C., 2010. Tabletops - Horizontal Interactive Displays, 1st Edition. Springer Publishing Company, Incorporated.
- Nacenta, M. A., Aliakseyeu, D., Subramanian, S., Gutwin, C., 2005. A Comparison of Techniques for Multi-Display Reaching. In: CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, New York, NY, USA, pp. 371–380.
- Nacenta, M. A., Pinelle, D., Stuckel, D., Gutwin, C., 2007. The Effects of Interaction Technique on Coordination in Tabletop Groupware. In: Proceedings of Graphics Interface 2007. GI '07. ACM, New York, NY, USA, pp. 191–198.
- Nacenta, M. A., Sallam, S., Champoux, B., Subramanian, S., Gutwin, C., 2006. Perspective Cursor: Perspective-Based Interaction for Multi-Display Environments. In: Proceedings of the SIGCHI conference on Human Factors in computing systems. CHI '06. ACM, New York, NY, USA, pp. 289–298.
- Naismith, L., Lonsdale, P., Vavoula, G., Sharples, M., 2004. Literature Review in Mobile Technologies and Learning. *Technology* 11, 48.
- Nakahara, J., Hisamatsu, S., Yaegashi, K., Yamauchi, Y., 2005. iTREE: Does the Mobile Phone Encourage Learners to be More Involved in Collaborative Learning? In: Proceedings of the 2005 conference on Computer support for collaborative learning: learning 2005: the next 10 years! CSCL '05. International Society of the Learning Sciences, pp. 470–478.

- Nardi, B., Whittaker, S., 2002. The Place of Face-to-Face Communication in Distributed Work. In: Hinds, P., Kiesler, S. (Eds.), *Distributed Work*. MIT Press, Cambridge, MA, pp. 83–110.
- Nastasi, B. K., Clements, D. H., 1992. Social-Cognitive Behaviors and Higher-Order Thinking in Educational Computer Environments. *Learning and Instruction* 2 (3), 215 – 238.
- Nielsen, J., November 1993. Iterative User-Interface Design. *Computer* 26, 32–41.
- Norman, D. A., September 2002. *The Design of Everyday Things*. Basic Books.
- Obexer, G., 2009. Folgestudie: “Note&Share - Untersuchung eines Werkzeuges zur Förderung von Wissensaustausch”. Project Thesis at the Ludwig-Maximilians-Universität München.
- O’Donnell, A. M., Dansereau, D. F., Hall, R. H., Rocklin, T. R., 1987. Cognitive, Social/Affective, and Metacognitive Outcomes of Scripted Cooperative Learning. *Journal of Educational Psychology* 79 (4), 431 – 437.
- Okada, A., Shum, S. J. B., Sherborne, T., 2008. *Knowledge Cartography: Software Tools and Mapping Techniques*, 1st Edition. Springer Publishing Company, Incorporated.
- Palincsar, A. S., Brown, A. L., 1984. Reciprocal Teaching of Comprehension-Fostering and Comprehension-Monitoring Activities. *Cognition and Instruction* 1 (2), 117–175.
- Pfister, H., Oehl, M., 2009. The Impact of Goal Focus, Task Type and Group Size on Synchronous Net-Based Collaborative Learning Discourses. *Journal of Computer Assisted Learning* 25, 161–176.
- Pfister, H.-R., Mühlpfordt, M., 2002. Supporting Discourse in a Synchronous Learning Environment: The Learning Protocol Approach. In: *Proceedings of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL Community. CSCL ’02*. International Society of the Learning Sciences, pp. 581–582.
- Piper, A. M., O’Brien, E., Morris, M. R., Winograd, T., 2006. SIDES: A Cooperative Tabletop Computer Game for Social Skills Development. In: *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work. CSCW ’06*. ACM, New York, NY, USA, pp. 1–10.
- Pleuss, A., Vitzthum, A., Hussmann, H., 2007. Integrating Heterogeneous Tools into Model-Centric Development of Interactive Applications. In: Engels, G., Opdyke, B., Schmidt, D., Weil, F. (Eds.), *Model Driven Engineering Languages and Systems*. Vol. 4735 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, pp. 241–255.
- Prante, T., Magerkurth, C., Streitz, N., 2002. Developing CSCW Tools for Idea Finding: Empirical Results and Implications for Design. In: *Proceedings of the 2002 ACM conference on Computer supported cooperative work (CSCW ’02)*. ACM, New York, NY, USA, pp. 106–115.
- Qin, Z., Johnson, D. W., Johnson, R. T., 1995. Cooperative Versus Competitive Efforts and Problem Solving. *Review of Educational Research* 65, 129–143.

- Rekimoto, J., 1997. Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments. In: Proceedings of the 10th annual ACM symposium on User interface software and technology (UIST '97). ACM, New York, NY, USA, pp. 31–39.
- Rekimoto, J., 1998. Multiple-Computer User Interfaces: A Cooperative Environment Consisting of Multiple Digital Devices. In: Proceedings of the First International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture. CoBuild '98. Springer-Verlag, London, UK, pp. 33–40.
- Rekimoto, J., Saitoh, M., 1999. Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments. In: Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '99). ACM, New York, NY, USA, pp. 378–385.
- Renkl, A., Stark, R., Gruber, H., Mandl, H., 1998. Learning from Worked-Out Examples: The Effect of Variability and Elicited Self-Explanations. *Contemporary Educational Psychology* 23, 90–107.
- Rick, J., 2009. Towards a Classroom Ecology of Devices: Interfaces for Collaborative Scripts. In: Workshop Proceedings of 8th International Conference on Computer Supported Collaborative Learning (CSCL2009): "Scripted vs. Free CS collaboration: alternatives and paths for adaptable and flexible CS scripted collaboration", 8-13 June 2009, Rhodes, Greece. pp. 73–80.
- Rick, J., 2010. Quadratic: Manipulating Algebraic Expressions on an Interactive Tabletop. In: Proceedings of the 9th International Conference on Interaction Design and Children (IDC 2010). Barcelona, Spain, pp. 304–307.
- Rick, J., Rogers, Y., 2008. From DigiQuilt to DigiTile: Adapting Educational Technology to a Multi-Touch Table. In: Proceedings of 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems (TABLETOP '08). IEEE Computer Society, pp. 73–80.
- Rogers, Y., Lim, Y., Hazlewood, W. R., Marshall, P., 2009. Equal Opportunities: Do Shareable Interfaces Promote More Group Participation Than Single User Displays? *Human-Computer Interaction* 24, 79–116.
- Rogers, Y., Lindley, S., 2004. Collaborating Around Vertical and Horizontal Large Interactive Displays: Which Way is Best? *Interacting with Computers* 16 (6), 1133 – 1152.
- Rogers, Y., Rodden, T., 2003. Configuring Spaces and Surfaces to Support Collaborative Interactions. In: Kenton O'Hara, Mark Perry, E. C., Russell, D. (Eds.), *Public and Situated Displays: Social and Interactional Aspects of Shared Display Technologies*. Kluwer, Dordrecht, Netherlands, pp. 45–79.
- Ronen, M., Kohen-Vacs, D., Raz-Fogel, N., 2006. Adopt & Adapt: Structuring, Sharing and Reusing Asynchronous Collaborative Pedagogy. In: Proceedings of the 7th international conference on Learning sciences. ICLS '06. International Society of the Learning Sciences, pp. 599–605.

- Roschelle, J., Pea, R., 2002. A Walk on the WILD Side: How Wireless Handhelds May Change CSCL. In: Proceedings of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL Community. CSCL '02. International Society of the Learning Sciences, pp. 51–60.
- Rosé, C., Wang, Y.-C., Cui, Y., Arguello, J., Stegmann, K., Weinberger, A., Fischer, F., 2008. Analyzing Collaborative Learning Processes Automatically: Exploiting the Advances of Computational Linguistics in Computer-Supported Collaborative Learning. *International Journal of Computer-Supported Collaborative Learning* 3 (3), 237–271.
- Rudd, J., Stern, K., Isensee, S., 1996. Low vs. High-Fidelity Prototyping Debate. *Interactions* 3 (1), 76–85.
- Ryall, K., Forlines, C., Shen, C., Morris, M. R., 2004. Exploring the Effects of Group Size and Table Size on Interactions with Tabletop Shared-Display Groupware. In: Proceedings of the 2004 ACM conference on Computer supported cooperative work. CSCW '04. ACM, New York, NY, USA, pp. 284–293.
- Scarlatos, L. L., 2002. TICLE: Using Multimedia Multimodal Guidance to Enhance Learning. *Inf. Sci. Inf. Comput. Sci.* 140 (1), 85–103.
- Schank, R. C., Abelson, R. P., 1977. *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*. L. Erlbaum, Hillsdale, NJ.
- Scheuer, O., Loll, F., Pinkwart, N., McLaren, B., 2010. Computer-Supported Argumentation: A Review of the State of the Art. *International Journal of Computer-Supported Collaborative Learning* 5, 43–102.
- Scott, S. D., Sheelagh, M., Carpendale, T., Inkpen, K. M., 2004. Territoriality in Collaborative Tabletop Workspaces. In: CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work. ACM, New York, NY, USA, pp. 294–303.
- Sharples, M., April 2000. The Design of Personal Mobile Technologies for Lifelong Learning. *Computers and Education* 34, 177–193.
- Shell, J. S., Vertegaal, R., Cheng, D., Skaburskis, A. W., Sohn, C., Stewart, A. J., Aoudeh, O., Dickie, C., 2004. ECSGlasses and EyePliances: Using Attention to Open Sociable Windows of Interaction. In: ETRA '04: Proceedings of the 2004 symposium on Eye tracking research & applications. ACM, New York, NY, USA, pp. 93–100.
- Shen, C., Everitt, K., Ryall, K., 2003. UbiTable: Impromptu Face-to-Face Collaboration on Horizontal Interactive Surfaces. In: Proceedings of UbiComp 2003. pp. 281–288.
- Shen, C., Ryall, K., Forlines, C., Esenther, A., Vernier, F. D., Everitt, K., Wu, M., Wigdor, D., Morris, M. R., Hancock, M., Tse, E., 2009. Collaborative Tabletop Research and Evaluation - Interfaces and Interactions for Direct-Touch Horizontal Surfaces. In: Pierre Dillenbourg, J. H., Cherubini, M. (Eds.), *Interactive Artifacts and Furniture Supporting Collaborative Work and Learning*. Springer, New York, pp. 111–127.

- Shen, C., Vernier, F. D., Forlines, C., Ringel, M., 2004. DiamondSpin: An Extensible Toolkit for Around-the-Table Interaction. In: CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, New York, NY, USA, pp. 167–174.
- Shih, P. C., Nguyen, D. H., Hirano, S. H., Redmiles, D. F., Hayes, G. R., 2009. GroupMind: Supporting Idea Generation Through A Collaborative Mind-Mapping Tool. In: GROUP '09: Proceedings of the ACM 2009 international conference on Supporting group work. ACM, New York, NY, USA, pp. 139–148.
- Simonson, M., Smaldino, S., Albright, M., Zvacek, S., 2000. Teaching and Learning at a Distance: Foundations of Distance Education. Prentice Hall, Upper Saddle River, NJ, US.
- Slavin, R. E., 1995. Cooperative Learning: Theory, Research and Practice. Allyn and Bacon, Boston, MA, USA.
- Sluis, R. J. W., Weevers, I., van Schijndel, C. H. G. J., Kolos-Mazuryk, L., Fitrianie, S., Martens, J. B. O. S., 2004. Read-It: Five-to-Seven-Year-Old Children Learn to Read in a Tabletop Environment. In: IDC '04: Proceedings of the 2004 conference on Interaction design and children. ACM, New York, NY, USA, pp. 73–80.
- Soller, A., Martínez, A., Jermann, P., Muehlenbrock, M., December 2005. From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. *International Journal of Artificial Intelligence in Education* 15, 261–290.
- Sommer, R., 1967. Small Group Ecology. *Psychological Bulletin* 67 (2), 145–152.
- Staatsinstitut für Schulqualität und Bildungsforschung, last accessed August 8th 2011. Berufs- und Studienorientierung im P-Seminar der gymnasialen Oberstufe.
- Stasser, G., Stewart, D., 1992. Discovery of Hidden Profiles by Decision-Making Groups: Solving a Problem Versus Making a Judgment. *Journal of Personality and Social Psychology* 63, 426–434.
- Stegmann, K., Christoph Wecker, A. W., Fischer, F., 2007a. Collaborative Argumentation and Cognitive Processing - An Empirical Study in a Computer-Supported Collaborative Learning Environment. In: Proceedings of the Computer-Supported Collaborative Learning 2007. pp. 661–670.
- Stegmann, K., Wecker, C., Harrer, A., Ronen, M., Kohen-Vacs, D., Dimitriadis, Y., Hernández-Leo, D., Fernandez, E., Asensio-Pérez, J., Fischer, F., 2011. How Can Current Approaches to the Transfer of Technology-Based Collaboration Scripts for Research and Practice Be Integrated? In: Spada, H., Stahl, G., Miyake, N., Law, N. (Eds.), CSCL 2011 Community Events Proceedings. ISLS.
- Stegmann, K., Weinberger, A., Fischer, F., 2007b. Facilitating Argumentative Knowledge Construction with Computer-Supported Collaboration Scripts. In: Proceedings of the Computer-Supported Collaborative Learning 2007. pp. 421–447.

- Streitz, N. A., Geissler, J., Haake, J. M., Hol, J., 1994. DOLPHIN: Integrated Meeting Support Across Local and Remote Desktop Environments and LiveBoards. In: CSCW '94: Proceedings of the 1994 ACM conference on Computer supported cooperative work. ACM, New York, NY, USA, pp. 345–358.
- Streng, S., Stegmann, K., Boring, S., Böhm, S., Fischer, F., Hussmann, H., Oct. 2010. Measuring Effects of Private and Shared Displays in Small-Group Knowledge Sharing Processes. In: Proceedings of the 6th Nordic Conference on Human-Computer Interaction (NordiCHI), Reykjavik, Iceland, Oct. 2010. ACM.
- Streng, S., Stegmann, K., Fischer, F., Hussmann, H., Jul. 2011a. From Graphical Learning Designs to Computer-Supported Collaboration Scripts: A Rapid Development Process. In: Proceedings of the International Conference on Technology-Enhanced Learning (ICTEL 2011).
- Streng, S., Stegmann, K., Hussmann, H., Fischer, F., Nov. 2009a. Metaphor or Diagram? Comparing Different Representations for Group Mirrors. In: Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group (CHISIG) of the Human Factors and Ergonomics Society of Australia (HFESA), Melbourne, Australia, Nov. 2009. ACM.
- Streng, S., Stegmann, K., Koch, J., Hussmann, H., Fischer, F., 2009b. Computer Support for Practicing Text Understanding in Pairs. In: Vilas, A. M., Martin, A. S., González, J. M., González, J. M. (Eds.), Research, Reflections and Innovations in Integrating ICT in Education (Vol. 1). FORMATEX, Badajoz, Spain, pp. 497–501.
- Streng, S., Stegmann, K., Wagner, C., Böhm, S., Hussmann, H., Fischer, F., 2011b. Supporting Argumentative Knowledge Construction in Face-to-Face Settings: From ArgueTable to ArgueWall. In: Proceedings of the conference on Computer-Supported Collaborative Learning (CSCL) 2011. International Society of the Learning Sciences (ISLS).
- Stuffer, M., 2011. Assigning Tasks and Roles Using an Interactive Tabletop Display in Collaborative Learning Sessions. Bachelor Thesis at the Ludwig-Maximilians-Universität München.
- Sturm, J., Iqbal, R., Kulyk, O., Wang, J., Terken, J., 2005. Peripheral Feedback on Participation Level to Support Meetings and Lectures. In: Proceedings of Designing Pleasurable Products Interfaces (DPPI). Eindhoven Technical University Press.
- Sugimoto, M., 2009. Design of Systems for Supporting Collaborative Learning with Physical Artefacts. In: Pierre Dillenbourg, J. H., Cherubini, M. (Eds.), Interactive Artifacts and Furniture Supporting Collaborative Work and Learning. Springer, New York, pp. 15–34.
- Suthers, D. D., Vatrappu, R., Medina, R., Joseph, S., Dwyer, N., 2008. Beyond Threaded Discussion: Representational Guidance in Asynchronous Collaborative Learning Environments. *Computers & Education* 50 (4), 1103–1127.
- Suzuki, H., Kato, H., 1994. AlgoBlock: A Tangible Programming Language for Collaborative Learning. In: Proceedings of the ED-Media '94.

- Swaminathan, K., Sato, S., 1997. Interaction Design for Large Displays. *Interactions* 4 (1), 15–24.
- Tandler, P., Prante, T., Müller-Tomfelde, C., Streitz, N., Steinmetz, R., 2001. *ConnecTables: Dynamic Coupling of Displays for the Flexible Creation of Shared Workspaces*. In: *UIST '01: Proceedings of the 14th annual ACM symposium on User interface software and technology*. ACM, New York, NY, USA, pp. 11–20.
- Tausch, S., 2010. *Untersuchung des Einflusses von Displaykonfigurationen auf phasenorientiertes Gruppenlernen*. Diploma Thesis at the Ludwig-Maximilians-Universität München.
- Torma, S., 2011. *Evaluating Different Display Settings for Task and Role Assignment in Collaborative Learning Sessions*. Bachelor Thesis at the Ludwig-Maximilians-Universität München.
- Toulmin, S. E., 1958. *The Uses of Argument*. Cambridge University Press.
- Trahasch, S., October 2004. *From Peer Assessment Towards Collaborative Learning*. *Frontiers of Education, FIE 2004, 34th Annual 2004* (2).
- Ullmer, B., Ishii, H., Jul. 2000. *Emerging Frameworks for Tangible User Interfaces*. *IBM Systems Journal* 39 (3-4), 915–931.
- Underkoffler, J., Ishii, H., 1998. *Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface*. In: *Proceedings of the SIGCHI conference on Human factors in computing systems. CHI '98*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, pp. 542–549.
- Villasclaras-Fernández, E. D., Hernández-Leo, D., Asensio-Pérez, J. I., Dimitriadis, Y., September 2009a. *Incorporating Assessment in a Pattern-Based Design Process for CSCL Scripts*. *Comput. Hum. Behav.* 25, 1028–1039.
- Villasclaras-Fernández, E. D., Isotani, S., Hayashi, Y., Mizoguchi, R., 2009b. *Looking Into Collaborative Learning: Design from Macro-and Micro-Script Perspectives*. In: *Proceedings of the 2009 conference on Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling*. IOS Press, Amsterdam, The Netherlands, The Netherlands, pp. 231–238.
- Wagner, C., 2010. *ArgueTable or ArgueWall? Comparing Different Display Environments for Collaborative Learning Through Argumentation*. Bachelor Thesis at the Ludwig-Maximilians-Universität München.
- Wagner, C., 2011. *Evaluating the Effects of Personal and Shared Workspaces on Argumentative Knowledge Construction*. PWAL Thesis at the Ludwig-Maximilians-Universität München.
- Warr, A., O'Neill, E., 2006. *Public Social Private Design (PSPD)*. In: *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*. ACM, New York, NY, USA, pp. 1499–1504.



- Wecker, C., Stegmann, K., Bernstein, F., Huber, M., Kalus, G., Kollar, I., Rathmayer, S., Fischer, F., 2010. S-COL: A Copernican Turn for the Development of Flexibly Reusable Collaboration Scripts. *International Journal of Computer-Supported Collaborative Learning* 5, 321–343.
- Weinberger, A., 2003. Scripts for Computer-Supported Collaborative Learning Effects of Social and Epistemic Cooperation Scripts on Collaborative Knowledge Construction. Ph.D. thesis, Ludwig-Maximilians-Universität, München, Germany.  
URL [http://hal.archives-ouvertes.fr/docs/00/19/01/71/PDF/Weinberger\\_2003.pdf](http://hal.archives-ouvertes.fr/docs/00/19/01/71/PDF/Weinberger_2003.pdf)
- Weinberger, A., Ertl, B., Fischer, F., Mandl, H., 2005a. Epistemic and Social Scripts in Computer-Supported Collaborative Learning. *Instructional Science*.
- Weinberger, A., Fischer, F., Stegmann, K., 2005b. Computer-Supported Collaborative Learning in Higher Education: Scripts for Argumentative Knowledge Construction in Distributed Groups. In: CSCL '05: Proceedings of the 2005 conference on Computer support for collaborative learning. International Society of the Learning Sciences, pp. 717–726.
- Weinberger, A., Ronen, M., Tchounikine, P., Harrer, A., Dillenbourg, P., Haake, J., Kali, Y., Fischer, F., Kohen-Vacs, D., 2007a. Languages and Platforms for CSCL Scripts. In: Chinn, C., Erkens, G., Puntambekar, S. (Eds.), *Proceedings of the Computer Supported Collaborative Learning (CSCL) Conference '07*. Vol. 8. International Society of the Learning Sciences, pp. 831–832.
- Weinberger, A., Stegmann, K., Fischer, F., Aug. 2007b. Knowledge Convergence in Collaborative Learning: Concepts and Assessment. *Learning and Instruction* 17 (4), 416–426.
- Weinberger, A., Stegmann, K., Fischer, F., July 2010. Learning to Argue Online: Scripted Groups Surpass Individuals (Unscripted Groups Do Not). *Computers in Human Behavior* 26, 506–515.
- Weinberger, A., Stegmann, K., Fischer, F., Mandl, H., 2007c. Scripting Argumentative Knowledge Construction in Computer-Supported Learning Environments. In: F. Fischer, K. Ingo, H. M. . J. M. H. (Ed.), *Scripting computer-supported collaborative learning : cognitive, computational and educational perspectives*. Springer, New York, pp. 191–211.
- Weisband, S. P., 1992. Group Discussion and First Advocacy Effects in Computer-Mediated and Face-to-Face Decision Making Groups. *Organizational Behavior and Human Decision Processes* 53 (3), 352 – 380.
- Wellner, P., July 1993. Interacting With Paper on the DigitalDesk. *Commun. ACM* 36, 87–96.
- Wessner, M., Pfister, H.-R., Miao, Y., 1999. Using Learning Protocols to Structure Computer-Supported Cooperative Learning. In: *Proceedings of the ED-MEDIA 1999 - World Conference on Educational Multimedia, Hypermedia & Telecommunications*, Seattle, Washington. Association for the Advancement of Computing in Education, Charlottesville, pp. 471–476.

- Wigdor, D., Jiang, H., Forlines, C., Borkin, M., Shen, C., 2009. WeSpace: The Design Development and Deployment of a Walk-Up and Share Multi-Surface Visual Collaboration System. In: CHI '09: Proceedings of the 27th international conference on Human factors in computing systems. ACM, New York, NY, USA, pp. 1237–1246.
- Wigdor, D., Shen, C., Forlines, C., Balakrishnan, R., 2007. Perception of Elementary Graphical Elements in Tabletop and Multi-Surface Environments. In: Proceedings of the SIGCHI conference on Human factors in computing systems. CHI '07. ACM, New York, NY, USA, pp. 473–482.
- Winter, F., 2011. Erweiterung und Evaluation eines Frameworks für Feedback-Systeme zur Unterstützung von Team-Sitzungen in der gymnasialen Oberstufe. Diploma Thesis at the Ludwig-Maximilians-Universität München.
- Zafeiriou, G., Nunes, J. M. B., Ford, N., 2001. Using Students' Perceptions of Participation in Collaborative Learning Activities in the Design of Online Learning Environments. *Education for Information* 19 (2), 83–106.
- Zarraonandia, T., Diaz, P., Guerra, E., Vargas, M. R. R., Aedo, I., 2009. A Framework for the Multi-disciplinary Design of Web-Based Educational Systems. In: *Advanced Learning Technologies*. pp. 45–49.
- Zhang, W., 2010. XML-basierte Integration graphischer Lernmodelle in ein Framework zur Entwicklung kollaborativer Lernanwendungen. Diploma Thesis at the Ludwig-Maximilians-Universität München.
- Zirker, S., 2011. Untersuchung von Einflussfaktoren beim Vergleich verschiedener Displayumgebungen für kollaboratives Lernen. Bachelor Thesis at the Ludwig-Maximilians-Universität München.
- Zufferey, G., Jermann, P., Lucchi, A., Dillenbourg, P., 2009. TinkerSheets: Using Paper Forms to Control and Visualize Tangible Simulations. In: *Tangible and Embedded Interaction'09*. pp. 377–384.