Context-Sensitive Interaction Support during Augmented Lectures

Detlef Bosau and Cora Burger and Martin Pfletschinger and Arno Wacker
University of Stuttgart
{firstname.lastname}@informatik.uni-stuttgart.de

October 23, 2002

Abstract

With mobile devices and wireless networks, new possibilities evolve to improve the interaction between teacher and students in class. The paper introduces the idea of DONUT, a Distributed annOtation tool for Notebook University Teachware, which is going to be developed within the NUSS (Notebook University Stuttgart) project[1] and integrated into an existing framework for application sharing. DONUT is based on defining and making use of semantic relationships between presented teaching material and contributions of students. Hence, automatic categorization of contributions becomes available in advance. This supports the teacher's decision, whether to integrate contributions into the lecture or postpone them. Especially, a lot of questions regarding the same topic can indicate problems in understanding on the side of students quite well.

1 Introduction

At all times, education has been important for communities. Among other factors, its success depends on the degree of interaction between people over the whole learning process. We focus on supporting one essential part of this process, namely a lecture in a large auditorium with a great many, e.g. more than 200, participants. In small classes, with less than 20 participants, students are asking questions more or less frequently. Furthermore, a teacher is able to observe their faces and to draw conclusions about the understanding and attentiveness of students. In a large audience, contributions of students are rare and a lecturer catches the faces of a few participants only. This situation becomes even worse with students participating from remote, an aspect that is beyond the scope of this paper.

Nowadays, electronic presentations via projector on one hand and mobile devices connected by wireless networks on the other become more and more wide spread. This provides an opportunity to increase the interaction between lecturer and students, even in front of a large audience. The following work has already been performed to study the applicability of these techniques to teaching:

- The Pebbles project [1] enables students to use a *tele-pointer* to explain and underline their questions or statements. In addition, the Pebbles project provides a set of tools like chat, whiteboard etc.
- The UCE-Architecture introduced by Mauve, Scheele and Geyer [4] provides a feedback tool, which allows the student to rate the teachers audibility, speed and clarity of the presentation. Support for telelectures has been introduced as well.
- Griffioen et al. have found encouraging results when sharing the view on slides among teacher and students via an electronic whiteboard and allowing public as well as private *annotation* [3].
- With the focus on tele-lectures, the JETS system introduced in [6] exhibits the possibility of electronic *hand-raising* to inform the lecturer about potential contributions of students.

These systems demonstrate the increasing interest in supportive tools for interaction support, like tele-pointing and annotating. One reason for the aforementioned tools becoming more and more popular is, that interactivity can be expanded from small classes even to large audiences.

[1]The NUSS project is funded by the German Ministry for Education and Research.

For instance, compared to real hand-raising in a crowded room, it may be easier to overcome one's inhibition by just clicking on a button and typing some question or murmuring it into the microphone of a mobile device instead of talking aloud. However, the concept of electronic hand-raising without further support would overstrain the lecturer. Confronted with a lot of questions and remarks from all over the auditorium, it is difficult for the lecturer to decide whom to call in which sequence. The reason behind, lies in the fact that there is no hint to estimate the value of contributions in advance.

Feedback tools, as introduced in [4], aim to provide the lecturer with information about clarity of his presentation or the attentiveness of the students by ratings given by the students. However, this approach bears some shortcomings. To provide valuable input to lecturers, students should permanently reflect the current teaching situation and their feelings, whether they like it, are annoyed or stay behind and so forth. This means a lot of effort and distraction for students, while the outcome of such reasoning is rather questionable. Neither freshness nor correctness of feedback data can be guaranteed, and thus do not help the teacher.

The work presented in this paper aims to overcome these disadvantages. The key idea is to add taxonomic and content specific information to all kinds of contributions like speak requests, annotations, pointing activities etc. For this purpose, the taxonomies given by Pfister in [5] and Mauve in [4] are combined and extended, as discussed later on in this paper. From an abstract point of view, all the above mentioned activities like pointing, annotating, hand-raising and feedback constitute a set of interaction types. Following the model given in [5], we enrich the scheme of unqualified interaction types by a content-based dimension, the interaction topic. While this model comprises many aspects, it is far too complex to be used directly by students. Hence, a facility is introduced allowing a lightweight categorization of contributions. This is performed by letting students mark the area in question, e. g. on a presentation slide that is shared between all participants. As a consequence, by making use of context-sensitive information being bound to this area, the system can classify contributions on its own. Students are not bothered with classification criteria but can concentrate on the material itself. Nevertheless, the lecturer's awareness of the class is increased. She will be presented contributions that are pre-ordered according to the above described categories. At first sight, the amount of each category will indicate the corresponding importance. Calling and answering one representative of a large group of questions for instance, potentially handles all of them. Depending on time, more or less contribution categories can be taken into account. By storing the remaining ones, the opportunity arises to treat them after the lecture, e. g. by electronic mail.

The functionality as briefly sketched above was named DONUT, a Distributed annOtation tool for Notebook University Teachware. In the following, it is outlined in more detail. Section 2 describes our extensions to the taxonomy of [4]. The realization takes place in the context of the SASCIA project as described in [2]. After sketching the overall architecture in Section 3, an example is given in Section 4. Finally, Section 5 summarizes the paper and depicts ongoing as well as future work.

2 Extended taxonomy for feedback

As laid out in the introduction, the taxonomy of [4] contains unspecified hand-raising and meta-information to classify feedback of students in terms of audibility, speed of the lecture and so forth. To enable pre-ordering of students' contributions in a proper way, a more comprising taxonomy is needed.

Actually, the existing classification scheme already refers to two distinct dimensions. From a more general point of view, these dimensions are the following:

- Interaction type
- Topic of the interaction

Regarding the interaction type, a suitable model has been proposed by [5]. They use the concept of speech acts to define templates for protocols among teacher and students. In the context of a lecture, the following categories for speech acts originating from students are known :

- Question, which normally is the most frequent type of contribution as used by students.
- Remark
- Answer to a teacher's question

Each of these interaction types can be uttered merely verbally or be supplied by some pointing or annotating instrument.

The second dimension is closely related to the lecture, i. e. organizational matters like schedule, meta-information as used by [4] and the teaching material. Examples for the latter, the mere content-based aspects are the following:

- The lecturer's talk and explanations
- Writings or drawings on black- or whiteboard
- Presented slides
- Physical or chemical experiments
- Simulation
- Videos

It can be seen, that some of these types of teaching material are electronically available whereas others are not. For material from outside the electronic world, a categorization can be performed by a list of keywords or free style interactions only. In contrast to this, for electronic applications like presentations or simulations, a more comfortable classification can be thought of. Depending on the interface and structure of an application, it can deliver context information about content items. For instance, a presentation is composed of several slides. Each one can contain various regions, e. g. text areas, graphics or tables. A specific question or remark of a student can refer to some region, a whole slide or some cross-reference among slides. By selecting the appropriate item, e. g. by marking one region on a specific slide, defines the topic of an interaction in an intuitive way.

Several components are needed to realize electronic input by students, pre-ordering and presentation of the interactions types described above to a lecturer. The one to handle the topic, i. e. either the definition of keywords or context-sensitive information as delivered by teaching applications, was named DONUT (cf. above).

3 Integrating DONUT into the SASCIA Framework

A facility like DONUT is not a stand-alone tool, but has to be integrated into an overall architecture for application sharing.

3.1 The SASCIA framework

The SASCIA system [2] provides middleware support to interaction tools and arbitrary applications. All communication within the SASCIA system is event-based, thus new components are easily added. In an early version, SASCIA provides the following features: recording of application data for subsequent replay, floor control for shared applications, user authentication and two applications (shared whiteboard, chat).

To enable the interaction support as described above, further components have to be integrated into SASCIA:

- DONUT to communicate with students on one hand and with applications on the other.
- Interaction Tracking System (ITS) to manage contributions as delivered by DONUT, present them to the lecturer and maybe pass them to the shared whiteboard.

Figure 1 shows the components of the SASCIA architecture as well as the additional components proposed in this paper. DONUT resides at the top of this figure, whereas the aforementioned shared whiteboard, which can be used as an overlay tool for distinct applications, is depicted as *Application Overlay*.

3.2 DONUT (Distributed annOtation tool for Notebook University Teachware)

The DONUT application is an enhancement for existing presentation applications used in lectures, e.g a slide show. DONUT allows to mark up regions of interest on the actual presentations.

The DONUT application presented in this paper simply appears as an overlay to the presentation seen by the student. The student is offered tools like a pen, a brush, a rubber as they are known

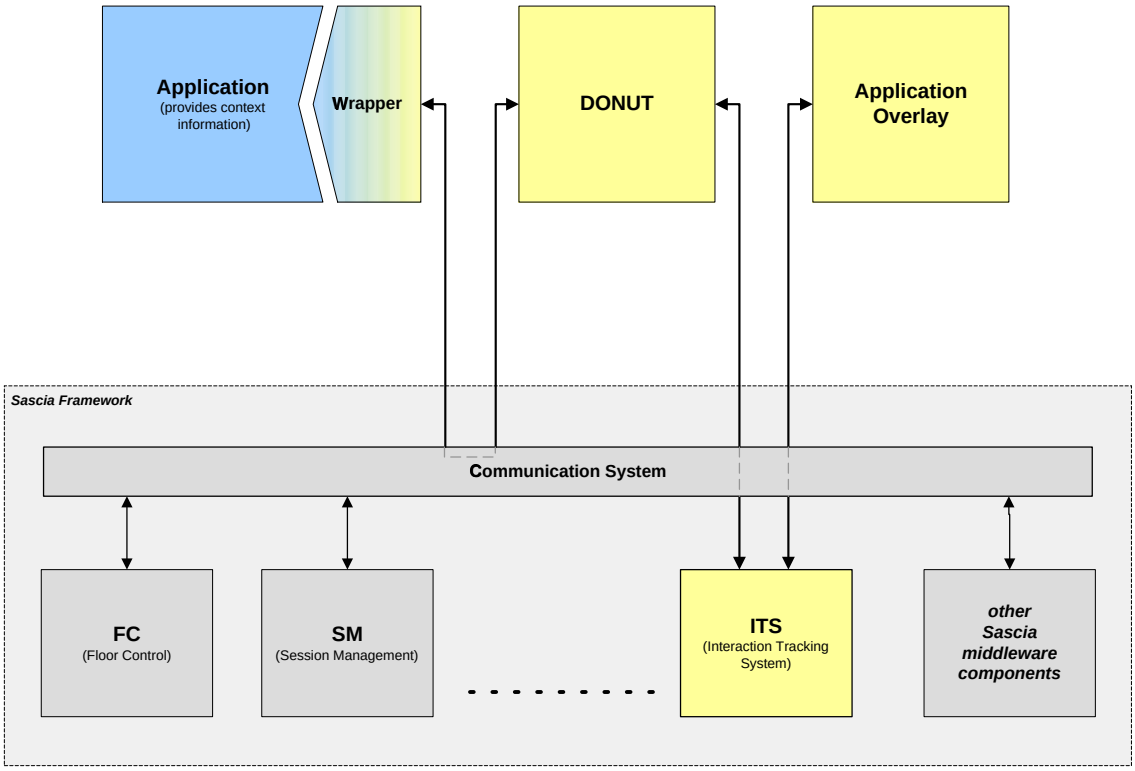

Figure 1: Architecture

from electronic whiteboards, with the only difference that the student does not write or draw on a whiteboard but annotates the presentation.

A student who wants to contribute to the lecture, activates the DONUT facility by marking an area of interest on his presentation. This can be done by drawing a circle around a certain area. As a result DONUT offers a pop up menu, where the student can choose whether to send a question without further comments to the teacher, to attach further annotations to his question or to cancel his request. Cancellation can be necessary if it was done by mistake or the question got answered in the meanwhile. Thus, in its simplest way, a contribution is created by just drawing a circle around an area of interest and commit by clicking an "OK" button.

In a more complicated variant the student can attach further annotations. This results in sending an interaction context request to the application in question. Depending on the application, this request can carry additional information, e. g. the marked area on the slide. In its answer, the application returns available context information like a list of predefined keywords which are related to the additional information. This enables the student to concretize his request in a more specific way. Moreover, it provides an opportunity to present background information on the topic. In some cases, this can even make obsolete a student's question by already answering it. In any case the student can append free-style text to his contribution.

The interaction tool may support different interaction types, as defined in Section 2. This exceeds the functionality of a simple hand-raising tool only supporting the interaction type *question*, as well as that of a shared whiteboard or overlay tool supplying the interaction type *remark*. The student is required to choose the appropriate interaction type prior to the submission. In the simplest way this can be performed by offering a distinct color for each interaction type. Moreover, he can specify whether his identification has to be hidden. All these parameters, like topic, interaction type etc. are then passed to the lecturer via the ITS, which is described in the next subsection.

In order to provide context information to DONUT, applications only expose an event based interface, which provides other components with application-specific context information. Section 4 gives an example of how this can be achieved for the widely used presentation program PowerPoint.

The SASCIA architecture defines appropriate wrappers for different platforms (JAVA applications, Windows, Linux etc.) which encapsulate applications and expose the aforementioned interface. In short: Any context information request is handled directly between DONUT and the application.

3.3 The Interaction Tracking System (ITS)

The most important extension to the original SASCIA system is the Interaction Tracking System (ITS), the purpose of which is to mediate contributions between students on one side and applications as well as the lecturer on the other. To this end, the ITS will

- Log the interaction for future reference.
- If the lecturer allows the student to make his contribution public, the latter is sent to the Overlay tool (cf. box on the right side in Figure 1). The lecturer is free to either allow or forbid public contributions in general or require them to be granted individually by her.
- Display pending interaction requests to the lecturer.

In each case, contributions have to pass the ITS. This is to ensure that each visible or audible question or remark etc. is controlled by the ITS, depending on the definition by the lecturer (general permission, inhibition or permission on an individual basis for each contribution). Only those contributions being permitted by the lecturer, are transferred to the Overlay Component via a suitable event. Hence, any perceivable change or command to the application and any event that has changing effects not only to the interaction tool itself is filtered by the ITS.

In general, the originator of a contribution is indicated by the Overlay tool. If a student wants to stay anonymous and has marked his contribution accordingly via DONUT, this reference is omitted. Nevertheless, the student's identity must be revealed to the lecturer in order to prevent misuse of the system.

Besides queuing and logging messages, the ITS offers means to display, filter and sort messages on the lecturers display. According to the taxonomy as introduced in Section 2, interaction requests are divided into questions, remarks and answers with or without annotation. In a second step, interactions are sorted with respect to the assigned keywords or application context. Questions and remarks not fitting any keyword are covered by a generic category "Free-Style Contributions". The ITS continuously monitors the number of pending interactions referring to a certain keyword. ITS is thus able to estimate the current interest in a certain topic and sort the keywords and interaction requests accordingly.

4 Powerpoint - A Sample Application

As a widely used presentation program, PowerPoint is well suited for a validation of the proposed concepts. PowerPoint provides access to documents and context data. Examples are the number of the slide, which is currently being presented, or the presentation notes, which are included within the PowerPoint document. We intend to specify context-specific information in XML format and to put this information into the presentation notes of the PowerPoint document. Hence, such information can be sent to a requesting interaction tool like DONUT, which in turn can provide the user with context specific information.

5 Conclusion and future work

Concepts of the new facility DONUT were introduced. It couples questions, remarks and maybe further contributions of students on one hand and its topic on the other in the most intuitive and comfortable way. Hence, instead of flooding lecturers with students' contributions, a neatly ordered list of categories is presented to them and stored for later usage.

Currently, DONUT is going to be implemented by integrating it into the existing SASCIA framework [2] for shared usage of teaching applications. With PowerPoint as an application, the software will be evaluated in lectures during summer 2003. Besides the performance of the system, acceptance by students and the definition of keywords resp. context-sensitive information are subject to examination.

References

[1] B. A. Myers and H. Stiel and R. Gargiulo. Collaboration using multiple PDAs connected to a PC. In *Proceedings of CSCW '98*, pages 285–294, Seattle (WA), November 1998.

[2] Cora Burger, Stella Papakosta, and Kurt Rothermel. Application sharing in teaching context with wireless networks. In *Proceedings World Congress NETWORKED LEARNING IN A GLOBAL ENVIRONMENT - Challenges and Solutions for Virtual Education NAISO (Natural and Artificial Intelligence Systems Organization)*, The Netherlands, May 2002.

[3] James Griffioen, W. Brent Seales, and James E. Lumpp Jr. Teaching in realtime wireless classroom. In *Proceedings of the 1998 Frontiers in Education Conference*, November 1998.

[4] Martin Mauve, Nicolai Scheele, and Werner Geyer. Enhancing synchronous distance education with pervasive devices. In *GI Jahrestagung (2)*, pages 1117–1122, 2001.

[5] Hans-Ruediger Pfister, Martin Wessner, Jennifer Beck-Wilson, Yongwu Miao, and Ralf Steinmetz. Rooms, protocols, and nets: Metaphors for computer supported cooperative learning of distributed groups.

[6] Shervin Shirmohammadi and Nicolas D. Georganas. JETS: A java-enabled telecollaboration system. In *International Conference on Multimedia Computing and Systems*, pages 541–547, 1997.