Template-driven Teacher Modelling Approach

A thesis submitted in partial fulfilment of the requirements for the degree of

Master of Science

in Information Science

at Massey University, Palmerston North

Department of Information Systems
College of Business

Yanmin Shi

2004

ACKNOWLEDGEMENTS

I am grateful to Dr. Alexei Tretiakov and Kinshuk for their collaboration in the preparation of our article on template server architecture which was presented at the 3rd IEEE International Conference on Advanced Learning Technologies and published in its proceedings. Also, I am grateful to Roland Kaschek for inviting me to make a presentation on the subject of this thesis at the e-Commerce seminar. I also would like to thank many lecturers, friends, and supporters who have made this happen. My loving wife, Jie Wang, assisted me in my research. Barry Jackson, Peter Blakey, Madre Chrystall, and the other members of academic staff were of invaluable help. Finally, thanks go to the HOD Chris Freyberg, the Secretary Julie, Lyons, Rachael Carruthers, and other supporting staff for providing the adequate, and well-equipped environment in which to work.

ABSTRACT

This thesis describes the Template-driven Teacher Modeling Approach, the initial implementation of the template server and the formative evaluation on the prototype. The initiative of Template-driven teacher modeling is to integrate the template server and intelligent teacher models in Web-based education systems for course authoring. There are a number of key components in the proposed system: user interface, template server and content repository. The Template-Driven Teacher Modeling (TDTM) architecture supports the course authoring by providing higher degree of control over the generation of presentation. The collection of accumulated templates in the template repository for a teacher or a group of teachers are selected as the inputs for the inference mechanism in teacher's model to calculate the best representation of the teaching strategy, and then predict teacher intention when he or she interacts with the system. Moreover, the presentation templates are kept to support the re-use of the on-line content at the level of individual screens with the help of Template Server.

TABLE OF CONTENT

TABLE OF FIGURES

TABLES

REFERENCE

APPENDIX

Chapter1 Introduction

1.1 Background

The broadening acceptance of the World Wide Web as a medium for access to commercial information and services has been accompanied by a growth in the spectrum of pedagogical practice, such as online teaching or distance learning. Education Service Providers (ESPs) are becoming aware of the needs to offer services that satisfy their users, i.e. both teachers and students. Ultimately, the effectiveness and efficiency of the teaching and learning processes become the main concerns in developing and deploying the Web-based educational systems [Specht, 1998]. The Intelligent Tutoring Systems (ITS) indicate the improvements to the effectiveness of the computer-mediated educational systems by introducing the student model [Wenger, 1987]. Based on students' knowledge levels and preferences, the systems cope with issues like personalization and adaptive presentation of contents. Such systems possess a degree of intelligence and adaptive nature. It means that the amount of the content and the navigation to the content could be dynamically adjusted according to student's academic level, preferences and learning ability. The motivation of modelling students in these systems is to overcome the students' tendency to cognitive overloading and enhance the effectiveness of the learning process. However, these systems have a number of problems and limitations when they are used in educational practice.

The first problem is the over-emphasis of on students and insufficient attention to the needs of the teacher. To date, most research efforts in adaptive systems have focused on the students' tutoring and the learning processes, i.e. the recipients of the content. Yet, the contributors of the content, teachers have their own needs when they create the course online. Their demands, which are different from students' requirements, haven't received much attention yet. Currently, most successful courseware authoring products, such as WebCT and BlackBoard, display some degree of adaptive natures and customization when students have access to the course contents. But none of them provides intelligent facilities to help the teachers to create the online education site [Bayne & Cook, 2002], and none of them help the teachers to select the right teaching

strategy in Web environment. In this case, the user model or teacher's model will play a very important role in helping the teacher to create the online courses.

The second problem is the excessive focus on the content sequencing, indexing and selecting in the current adaptive hypermedia system and less attention has been paid to the presentation of the content [Brusilovsky, 1998a]. In these systems, the course developers and teachers either define the navigation styles and choice of document formats for each of the learning units [De Bra, 1999], or otherwise rely on the system to set up the presentation automatically in terms of a few pre-defined templates which are common to all course content units [Carro, et al. 1999]. The teachers of Web-based educational systems find it hard to modify the presentation unless they have enough knowledge on content selection, the architecture of the adaptive systems and underlying structure of student models.

Another problem is the exclusion of the teacher's role in the learning process. In terms of traditional teaching theories, such as "Objectivism", teachers are responsible for selecting teaching methods, setting scenes, providing content, managing the curriculum, and overseeing the learning progress. It is said that the teacher's teaching style and methods are more important than student's learning style [Kinshuk, Patel. Oppermann, Russell, 2001]. In most Web-based educational systems, there is no face-to-face, one-to-one direct interaction between teacher and students. The interactions between teacher and students normally happen in indirect forms that are mediated by the systems, such as email exchanges, or online chat. Teachers usually stay behind the scene. As a result, the interaction between students and content outweighs the interaction between students and teacher. When students interact with the content over the system, the teacher model which represents the human teacher will oversee the student's learning process to provide adequate guidance when necessary.

The lack of a uniform format for data interchange is a common deficiency in the commercial courseware, document authoring and intelligent tutoring systems. These systems normally consist of several modules which exchange data to implement a task. Take the example of the intelligent tutoring system, the content data (document

fragments) and student model data are stored in a database. This data is extracted from the database (using structured query language), calculated in the control program (vendor-specific), and presented in the user interface with a predefined document format. All of these data formats and document format are vendor-specified. It presents some difficulties in integrating one system with another, or exchanging data with another. If the extensible mark-up language XML is adopted, it could make the data exchange much easier [SUN, 2002], as it offers a common, platform–independent, standards–supported basis for formatting data. If standards for educational content formatting based on XML technology, such as IMS [IMS, 2003], are adopted, these standards will definitely have a strong impact on commercial courseware, document authoring and intelligent tutoring systems.

In order to design the robust Web-based educational systems, we need to overcome the shortcomings of the intelligent tutoring system, commercial courseware authoring tools. Templates have been widely accepted in multimedia authoring and commercial document authoring (MS Word, PowerPoint). When templates are described in XML rather than vender-specific formats, they provide a more lasting embodiment of reusable content, because their useful lifetime is no longer bound to the lifetime of the applications used to create them.

In this thesis, we consider the application of templates to the educational content creation. In Web-based educational content creation, templates provide a high-level description of the presentation structure and relations of domain knowledge in the absence of content. Template can also be broken down into template fragments. Template can carry the template fragments attributes such as the name of the template fragments, the incidence of the fragment, and the sequence of the template fragments. As long as we can find out the relationships between the templates, template fragments and teachers' behaviour when they interact with the system, templates and usage patterns can be used to define and train the teacher model. In addition, it also provides a mechanism for sharing and distributing the templates, thus sharing and reusing presentation design ideas.

Here, we suggest a Template-Driven Teacher Modelling (TDTM) architecture which supports the course authoring by providing a higher degree of control over the creation of presentation. The collection of accumulated templates for a teacher or a group of teachers is selected as the inputs for the inference mechanism in the teacher's model to figure out the best representation of the teaching strategy and then to predict teacher's intention when he or she interacts with the system. Moreover, the presentation templates are kept to support the re-use of the on-line content at the level of individual screens with the help of Template Server. With Template Server Architecture, teachers will be able to view the course content in the designated template and document format. Thus, we are filling in what appears to be a gap in the current practice and research in the area of courseware authoring [Tretiakov & Shi, 2003].

1.2 The objective of the research

As has been stated above, the main objective of the research is to enrich the Web-based teaching theories by introducing the template-driven teacher modelling (TDTM) architecture. Based on this architecture and the Template Server Approach, two prototype systems will be partially built to test the new approach. More specifically, these prototype systems will be used to prove the feasibility of the architecture, and collect user's attitudes towards the architecture and template server approaches. Before the formal usability test is initiated on these prototype systems, we will carry out an evaluation on these prototypes. The feedback from this evaluation will help enhance the architecture, supplement the template server approach, and consolidate the online teaching theories.

As long as the teacher model is embedded into a course authoring system, it will greatly reduce a teacher's effort in authoring course content which can be easily adapted to the teacher's teaching style or method. It also allows the teacher to make selections among a variety of templates and multimedia presentation formats (hypertext, graphics, animation, audio, video) or build their own templates in terms of the natures of the discipline and course content, which implicitly represents teaching methods (lecturing, collaboration-oriented, simulation-based, exploration-based etc.).

1.3 Improvement of Adaptability

In order to cater for the teacher's needs in a Web-based education course, the system must possess adaptivity (the system automatically adjusts to the individual teachers) and reusability. It is widely accepted that the intelligent tutoring systems, present two levels of adaptation: content level, and link or navigation level adaptations [Brusilovsky, et al. 1996]. In the first level of adaptation, the system will decide what content will be presented to the user in terms of the user model and task. In the second level of adaptation, the links or navigation to other content documents will be put into certain sequence, be annotated with different colours, icons or dimming, or be disabled in terms of user model (knowledge level goal and task).

All of these adaptation techniques can be adopted in the proposed Template-Driven Teacher-Modelling (TDTM) architecture for the course authoring systems. Instead of manipulating the course content, TDTM focuses on dynamically presenting templates in terms of the teacher model. The course content unit in the intelligent tutoring system contains content fragments. Similarly, template in TDTM consists of template fragments. Various relationships, association rules, clustering can be mined from the usage of templates and template fragments. These rules will be used in the teacher model to implement the adaptation. Like the Intelligent Tutoring Systems, TDTM also possesses two levels of adaptation. In the first level of adaptation, TDTM will decide how the template looks and which group of template fragments will be integrated into the template in terms of teacher's model. In the second level of adaptation, the system will define in which way templates can be found. For example, putting them into certain sequence, or annotating them with different colours, icons or dimming, or disabling them in terms of the teacher model.

1.4 The modelling methods and techniques

Teacher modelling starts with data acquisition (mainly from observing requests to the template repository) regarding to teacher's behaviour when he or she interacts with the system. Then based on the data, a set of data mining techniques will be used to discover association, classification or clustering rules. These rules or algorithms will be used to predict the teacher's actions. Data mining is a process of supervised or unsupervised discovery of interesting, useful, and previously unknown information from this set of data [Kumar, 2002]. The traditional methods and techniques for data mining include classification, discovery of association rules, and clustering. Classification relates to grouping interaction sessions, association rules relate to predicting the usage patterns within a session, and clustering refers to the dynamic approaches to classification, which do not use explicit information about the user.

One of the techniques to facilitate the teacher model is to classify teachers based on teacher's attributes and usage history to the templates and template fragments. Then the system can make predictions about the teacher's intention when he or she interacts with the system. The data of templates and the navigation to (or the selection of) the templates will depend on the settings in these attributes. Some of the potential teacher's attributes are: native language, familiarity to the multimedia authoring tools, teaching experience and teaching history (the number of years in teaching, tertiary or secondary sector). Moreover, a number of environmental attributes, such as organizational attributes, geographical attributes and orientation attributes (space, temporal, belief), could also be associated with teachers and have their impacts on the system by influencing the teacher's behaviour [Iivonen & Sonnenwald, 1998].

However, some of the teacher's attributes are very difficult to describe and quantify in an explicit way, such as different teaching styles (theory first or practice first, procedural or declarative), teaching philosophy (Instructivist or constructivist), social and cultural backgrounds. There are wide variations from individual to individual even though they all belong to the same user group. Furthermore, some attributes show certain dynamic natures. It means that they are changing from time to time when

teachers change their goals and tasks. Modelling implicit aspects of this teacher model presents a real challenge to researchers and developers. The template server approach adopted in this research will take into account the possibility to retrieve implicit information from teacher behaviour as observed by the system by using a decision engine such as Bayesian networks.

We create an architecture (Template-driven Teacher Modelling) which incorporates the template server and teacher model. As the changes of teacher's mind and preferences will be reflected in the usage of the templates, the templates become the key components of template-driven teacher modelling architecture.

1.5 Description of the research

Generally put, a system based on the template-driven teacher modelling architecture will assist teachers in creating or re-constructing the online courses using templates that reflect basic preferences of the teacher, and represent different teaching styles and tasks. The teacher model will play its role reflecting the teacher's preferences and making adequate predictions. The templates will be adapted to represent both the explicit teacher's preferences, and the implicit teacher's teaching style.

In terms of the implementation of teacher model, both implicit and explicit aspects of the teacher model could be stored in the same logical and physical layer on the architecture. It could be stored in a relational database, or embedded in an XML file. We opted for XML because it does not bind us to a particular database management system and it offers a format which can be readily consumed as a data source by a variety of systems without raising the portability issues.

The proposed system will work in the following way. When a teacher logs into the system and initiates a task, several templates appropriate for the task will be offered for selection according to the default data stored in the teacher's model. After one template has been selected and filled with course content, the course content will be stored in a database or an XML file. The selections of templates become part of the teaching

activities. The teaching strategy is a particular combination of teaching activities put together in a sequence [Dee Fink, 2003]. Therefore any major changes in the use of templates may suggest the changes of teacher's teaching strategy. Consequently, the teacher's model will be updated to reflect these changes. The teacher model is maintained in teaching strategy level. The rationale is that teachers could change the teaching strategy in terms of the nature of the content, the academic level and cultural backgrounds of the students [Aikenhead, G. S., 1997]. Considering the system at the highest level of generality, we do not distinguish the teacher model and the data behind it, as we envisage that the teacher model can be modified not only by changing the parameter values, but also by replacing the components and the implementation algorithms. This is the spirit of the Template Method and similar design patterns [Gamma, et al. 1995].

Additionally, the course contents in the database or XML file are globally accessible and reusable. They can be rendered into different document formats (MS Word, PowerPoint, HTML). Moreover, as we are using semantic mark-up to store the course content, any new formats that may emerge in the future can be easily accommodated.

1.6 Schedule of the research

This research project consists of a number of clearly defined stages. In the planning and investigative stage, current literature on Web-based teaching systems and intelligent or adaptive tutoring systems are reviewed. The main purpose of this stage is to be familiar with the research backgrounds, and to initiate the research plan. In the following requirement analysis stage, we analyze and identify the strengths, weaknesses of current Web-based educational systems, and formulate the requirements for the proposed template-driven teacher modelling architecture. Then the framework of the proposed template-driven teacher modelling architecture and the template server are discussed. In the design stage, we seek to answer the question "How will the architecture and the template server approach address the weakness of most Web-based course authoring systems?" The key component of the prototypes is a template server, which serves the template on demand. We also describe the requirements analysis for client applications

that play the vital role in Template-driven teacher modelling architecture. In the implementation stage, software development tools and programming languages, such as Java, JavaScript, XML, XSLT, HTML, CSS, XMLDOM, are used to build the prototype systems that facilitates the teacher model and the template server which serves the templates.

1.7 Expected outcomes

The ultimate outcome of this research is to construct the Template-driven Teacher Modelling Architecture, and to evaluate the template server approach. The architecture is used as the guidelines to build prototype systems (not complete systems) that demonstrate how templates could be served adaptively. Because the templates usage patterns are being used as an input to the teacher model, the template server becomes the key component to implement in this research project. The template server approach is evaluated with the end-users to identify its usefulness and weakness to suggest further improvements. To ensure feasibility, the prototype will address a limited implementation of the Template-driven Teacher Modelling Architecture, as far as the scope of support to teachers is concerned. On the other hand, the architectural discussions will be conducted at a high level of generality.

1.8 The organization of this research thesis

This thesis is organized in the following order:

- In Chapter 1, we outlined the research questions and objectives, schedule, and expected outcomes.
- In Chapter 2 "Education Theories", we take a close look at the teaching and learning processes in context of Web-based education. We also discuss the debate over teaching paradigms such as "Instructivism", "Constructivism" and "Social-cultural theories" within the education community. We focus on their impacts on education practice and their implications for Web-based educational system.
- In Chapter 3 "Web-based educational systems", we discuss the categorizations of Web-based educational systems, such as course authoring and tutoring

systems. Special attention is given to the common features of commercial courseware packages and concept-based research projects. We also address the latest intelligent and adaptive techniques in Web-based educational systems.

- In Chapter 4 "Modelling techniques for Web-based educational systems", the basic user groups of the Web-based educational systems are identified, teachers and students. Various student and user modelling techniques will be discussed extensively. At the end of the chapter, we outline our approach to teacher modelling by suggesting the Template Driven Teacher Modelling architecture.
- In Chapter 5 "Template Server Architecture", Template Server is described in detail. Some of the terms appear frequently in this chapter, such as Template Server, Template Engine, Template Repository and Content Server. The UML (Uniformed Modelling Language) will be used to describe each component in the architecture and the interactions between them.
- In Chapter 6 "Technical Implementation", we discuss the technical issues related to the implementation of the prototype. The role of XML, deployment of components of Template Server, and installation of software packages are described in details.
- In Chapter 7, we design and conduct a user evaluation which is to check the viability of the Template Server Approach and we analyze the results of the evaluation.
- In Chapter 8 "Conclusion, Discussion and Future work", we conclude the research and highlight the main contributions. Then we explore the possible improvement to the proposed Template Server Architecture.