Chapter 1

FACILITATING LEARNING THROUGH DYNAMIC STUDENT MODELLING OF LEARNING STYLES –
An Architecture and its Application for Providing Adaptivity

SABINE GRAF, KINSHUK, QINGSHENG ZHANG, PAUL MAGUIRE AND VICTORIA SHTERN
School of Computing and Information Systems, Athabasca University, Canada

Abstract:
Technology enhanced learning environments usually track a variety of data about students' behaviour while students are learning. These data can be used to infer valuable information about how students learn as well as about their characteristics. This chapter focuses on the consideration of learning styles in technology enhanced learning. Considering students' learning styles in technology enhanced learning can have many benefits for students such as providing them with personalized recommendations and advice based on their learning styles. In this chapter, we introduce an architecture that aims at dynamically identifying students' learning styles from their behaviour in a learning system by frequently checking students' behaviour and updating their learning styles based on their recent behaviour. Such a dynamic student modelling approach enables systems to incrementally learn students' learning styles, identify and consider exceptional behaviour of students, and respond to changes in students' learning styles by updating the student model respectively. The proposed architecture has been developed with only few dependencies to the learning system, making it possible to easily adjust and use the architecture for different learning systems. In this chapter, the integration of the proposed architecture into a particular learning system is demonstrated. Furthermore, an adaptivity module has been developed to show the benefits of the proposed architecture. This adaptivity module accesses the information about students' learning styles to provide students with adaptive feedback about their learning styles as well as about how to improve their learning processes considering their learning styles and their courses.

Keywords:
Dynamic Student Modelling, Static Student Modelling, Learning Styles, Adaptivity and Personalization in Learning Systems

1. INTRODUCTION

Technology enhanced learning environments provide many new ways of facilitating learning as well as teaching. One of the possibilities that are available in such learning environments is to get information about how students learn and use online courses by tracking their learning paths and activities in the system. Such information can be very valuable in many ways and can be used, for example, for identifying when students have difficulties in learning and to get feedback about the course materials such as whether particular types of learning objects/activities (e.g., videos, exercises, etc.) are actually used by students as well as which learning objects/activities seem to be difficult for students, indicating the need for improvement of the respective learning materials. Furthermore, information from students' behaviour in an online course can be used to identify students' characteristics such as their learning styles (e.g., García, Amandi, Schiaffino, & Campo, 2007; Graf, Kinshuk, & Liu, 2009; Özpolat & Akar, 2009), cognitive abilities (e.g., Kinshuk & Lin, 2004; Lin, 2007), and affective states (e.g., Khan, Graf, Weippl, & Tjoa, 2010).

In this chapter, we focus on the consideration of students' learning styles in technology enhanced learning environments as well as on the dynamic identification of learning styles from students' behaviour in an online course. Knowing students' learning styles and considering this information in the learning process can lead to many benefits for students. First, students can be made aware of their learning styles as well as the implications of their learning styles for learning, including general strengths and weaknesses of students in the learning process. Such information can help students to understand why learning is sometimes difficult for them and builds the basis for developing their weaknesses. Second, the information about students' learning styles can be used to match the teaching style with the students' learning styles. Providing students with learning material/activities and personalized recommendations that fit their preferred ways of learning can make learning easier for them. This matching hypothesis is supported by educational theories. Moreover, studies such as those by Bajraktarevic et al. (2003), Graf and Kinshuk (2007), and Popescu (2010) demonstrated supportive results and showed that students can learn easier and faster if their courses are adapted to their learning styles.

To consider learning styles in education, the students' learning styles need to be known first. Brusilovsky (1996) distinguished between two different ways of student modelling: *collaborative* and *automatic*. In the collaborative approach, the students provide explicit feedback which can be used to build and update a student model, such as filling out a learning style questionnaire. In the automatic approach, the process of building and updating the student model is done automatically based on the behaviour and actions of students while they are using the system for learning. The automatic approach is direct and free from the problem of inaccurate self-conceptions of students. Moreover, it allows students to focus only on learning rather than additionally providing explicit feedback about their preferences. In contrast to learning style questionnaires, an automatic approach can also be more accurate and less error-prone since it analyses data from a time span rather than data which are gathered at one specific point of time.

Additionally, student modelling can be classified as *static* or *dynamic*. Static student modelling refers to an approach where the student model is initialised only once (mostly when students register in the system). In contrast, a dynamic student modelling approach frequently updates the information in the student model and therefore allows responding to changes of the investigated student characteristic. A dynamic approach has two advantages over a static one in the context of identifying learning styles. First, dynamic student modelling can consider exceptional behaviour of students and can extend static student modelling by incrementally improving and fine-tuning the information in the student model in real-time, learning students' learning styles until the learning styles have been identified reliably. Therefore, dynamic student modelling can contribute to identify students' learning styles with higher accuracy, considering new data whenever students use the system for learning. Second, since many of the major learning style models argue that learning styles can change over time, dynamic student modelling allows monitoring students' behaviour, identifying changes in their learning styles, and updating the learning styles once they changed.

When looking at the student modelling approaches that are used by adaptive learning systems that aim at providing adaptivity based on learning styles, it can be seen that a lot of these systems use questionnaires (a static and collaborative approach). Examples of such systems are CS383 (Carver, Howard, & Lane, 1999), IDEAL (Shang, Shi, & Chen, 2001), and LSAS (Bajraktarevic et al., 2003). Recently, more and more research has been done on developing automatic student modelling approaches by considering students' behaviour in a course. However, these approaches typically use a predefined amount of behaviour data for identifying students' learning styles at one point of time and are therefore automatic but still static approaches (Cha et al., 2006; García et al., 2007; Graf et al., 2009; Özpolat & Akar, 2009). Very little research has been conducted so far on developing approaches which aim at dynamic and automatic student modelling of learning style, where the system monitors a students' behaviour and uses this behaviour data to frequently update learning styles of students.

In this chapter, an architecture is introduced which integrates dynamic student modelling into existing learning systems, enabling them to monitor students' behaviour, analyse these data for detecting and frequently updating students' learning styles, and storing the information about students' learning styles in a student model which can be accessed by the system in order to provide adaptive and personalized support for students. The introduced architecture has been integrated in a learning system and a module for providing students with adaptive support has been developed in order to demonstrate the benefits of dynamic student modelling of learning styles and the introduced architecture.

This research is based on the Felder-Silverman learning style model (FSLSM) (Felder & Silverman, 1988). FSLSM is a learning style model that describes learning styles in detail and is therefore highly appropriate for providing adaptivity in learning systems. Furthermore, the FSLSM is based on the concept of tendencies, allowing handling of exceptional behaviour by considering learning styles as a main tendency rather than as an obligatory type. FSLSM assumes that these tendencies are "flexibly stable", meaning that they are more or less stable but can change over time, for example, if a student is training his/her weak learning preferences. Moreover, FSLSM is used very often in technology enhanced learning and some researchers even argue that it is the most appropriate learning style model for the use in adaptive learning systems (Carver et al., 1999; Kuljis & Liu, 2005). According to FSLSM, each learner has a preference for each of its four dimensions (active/reflective, sensing/intuitive, visual/verbal, sequential/global). *Active* learners prefer to learn by trying things out and working with others, whereas *reflective* learners prefer to learn by thinking things through and working alone. *Sensing* learners like to learn from concrete material like examples and tend to be more practical and more careful with details, whereas *intuitive* learners prefer to learn abstract material, tend to be more innovative, and like challenges. *Visual* learners remember best what they have seen, whereas *verbal* learners get more out of words, regardless of whether those words are spoken or written. *Sequential* learners learn in linear steps and prefer to be guided through the learning process, whereas *global* learners learn in large leaps and prefer more freedom in their learning process.

In the next section, the architecture for dynamic student modelling is described. Section 3 deals with the integration of the architecture into a learning system and describes the adaptivity module used for providing students with personalized recommendations based on their learning styles. Section 4 concludes the chapter and provides some directions for future work.

2. ARCHITECTURE FOR DYNAMIC STUDENT MODELLING

In this section, an architecture is presented that aims at enabling existing learning systems to build and frequently update a cognitive profile of their students, which is stored within the student model and includes information about students' learning styles based on the FSLSM. The architecture is illustrated in Figure 1. In order to integrate the architecture into a learning system, a *Notification Mechanism* needs to be added to the learning system, which notifies the *Dynamic Student Modelling Module* about students' actions. After a student has performed a predefined number of actions, the *Dynamic Student Modelling Module* requests the *Learning Style Calculation Module* to recalculate the learning styles of the particular student based on his/her recent behaviour in the system. The *Learning Style Calculation Module* requests the relevant data for calculating learning styles from the *Data Extraction Module* which connects to the data sources of the learning system, extracts the requested data and delivers them to the *Learning Style Calculation Module* where the current learning styles of the student are calculated and then stored in the *Student Model* as intermediate result. Once this process has been completed, the *Learning Style Calculation Module* reports back to the *Dynamic Student Modelling Module* which requests the *Dynamic Analysis Module* to check whether the results of the *Learning Style Calculation Module* differ significantly from the currently stored learning styles of the student in his/her cognitive profile and update the learning styles if required.

In addition to dynamic student modelling, static student modelling is integrated in the architecture for providing an option to initialise the cognitive profile in the *Student Model* from the data of a questionnaire. The *Static Student Modelling Module*, which administers the learning style questionnaire, is called directly from the learning system, for example, when students register the first time, and stores the results of the questionnaire in the cognitive profile of the *Student Model*. These results are then used as the student's learning styles until information from the student's behaviour is received for fine-tuning and updating the learning styles.

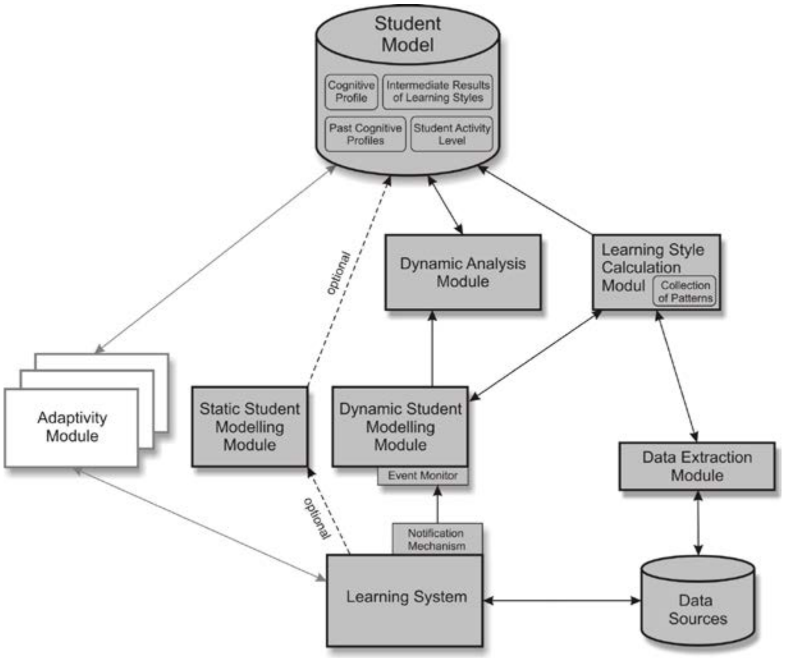

Figure 1 – Architecture for Dynamic Student Modelling

The modules of the architecture are designed to be as independent as possible with respect to the learning system, so that they can be integrated in different systems. Only two modules are system-dependent: the *Data Extraction Module* aims at extracting relevant data from the data sources of the learning system (e.g., a database that includes logs of students' behaviour) and therefore has to consider the different structures of data sources in different learning systems. Furthermore, the *Notification Mechanism* has to be directly integrated in the learning system for sending notifications whenever a student performs an action in the learning system. In addition, a link to the *Static Student Modelling Module* has to be provided within the learning system for integrating static student modelling.

In order to use the information about learning styles identified through the proposed architecture for dynamic student modelling, adaptivity modules can be added to the learning system. These adaptivity modules can access the cognitive profile in the *Student Model* and use the information about students' learning styles, for example, for providing students with personalized recommendations and/or adaptive courses based on students' learning styles. The implementation of such an adaptivity module is described in Section 3.

In the following subsections, the modules of the architecture are described in more detail.

2.1 Static Student Modelling Module

The *Static Student Modelling Module* aims at providing an option for initialising the cognitive profile through the use of a questionnaire. Such a questionnaire enables a system to quickly gather information about students' learning styles, which can then be refined and updated through dynamic student modelling once students use the system for learning. Therefore, by combining static and dynamic student modelling, adaptivity can be provided right after a student filled out the questionnaire rather than having to wait until enough information from student's behaviour is available for calculating his/her learning styles.

The *Static Student Modelling Module* is called through a link that can be added in the learning system, for example, when students register in the system or for their first course. Once students click on this link, the *Static Student Modelling Module* presents them with the Index of Learning Styles questionnaire (Felder & Soloman, 1997), a 44-item questionnaire that has been developed by Felder and Soloman in order to identify learning styles based on FSLSM. Once students filled out this questionnaire, the results, four values between +11 and -11 indicating the preference on each of the four learning style dimensions of FSLSM, are normalised to values between 0 and 1 and stored in the cognitive profile of the *Student Model*. These values are used as currently identified learning styles which are later refined and updated through dynamic student modelling. Students can choose not to fill out the questionnaire. In that case, no information is stored in the cognitive profile and dynamic student modelling is used to build and update information about students' learning styles from students' behaviour in the learning system.

2.2 Notification Mechanism

The *Notification Mechanism* is a system-dependent component which is integrated in the learning system and can be seen as the interface between the learning system and the *Dynamic Student Modelling Module*. The *Notification Mechanism* is responsible for notifying the *Dynamic Student Modelling Module* when a student is performing an action in the learning system. Actions are defined as visits of learning objects/activities so that when a student is going through the course, each learning object/activity that he/she is visiting is considered as an action. Whenever such an action occurs, the *Notification Mechanism* sends a message with the student ID and the course ID to the *Dynamic Student Modelling Module*.

2.3 Dynamic Student Modelling Module

The *Dynamic Student Modelling Module* is responsible for managing the dynamic student modelling process. This includes two activities. First, the *Dynamic Student Modelling Module* monitors students' activity levels, in terms of how many actions they perform, based on the messages received from the *Notification Mechanism*. Second, the *Dynamic Student Modelling Module* requests recalculations of students' learning styles once a student performed a predefined number of actions since the last recalculation of his/her learning styles. Such recalculations aim at considering students' recent behaviour in the calculation process of their learning styles and checking whether their behaviour still reflects the students' learning styles as stored in the cognitive profile of the *Student Model* or whether updates in the cognitive profile are required. Therefore, the *Dynamic Student Modelling Module* first requests a recalculation from the *Learning Style Calculation Module* and subsequently requests the *Dynamic Analysis Module* to check whether the newly calculated learning styles from the *Learning Style Calculation Module* differ significantly from the learning styles that are currently stored in the cognitive profile, considering cases of exceptional behaviour in students' actions, and update the learning styles in the cognitive profile if required.

2.4 Learning Style Calculation Module

The *Learning Style Calculation Module* aims at calculating students' learning styles from their behaviour in a learning system. This calculation is based on certain patterns of behaviour, which provide indications of students' learning styles. Such patterns can be, for example, the number of visits of particular types of learning objects/activities such as content, exercises, and quizzes, and the time spent on such types of learning objects/activities.

The *Learning Style Calculation Module* includes a collection of patterns where each pattern provides indications for identifying learning styles based on a particular learning style dimension of the FSLSM. Furthermore, information for each pattern is included that states how the pattern affects a certain learning style dimension (e.g., whether a high number of visits of exercises is an indication for an active or reflective learning style). Since different learning systems support different types of learning objects/activities, not all patterns can be used for each learning system and only the patterns that deal with types of learning objects/activities that are considered in the respective learning system are used in the calculation process. The collection of patterns can be extended in the case that a learning system considers types of learning objects/activities that are not included so far.

In order to get data about a student's behaviour with respect to each pattern that is considered in the learning system, the *Learning Style Calculation Module* sends a request to the *Data Extraction Module*, which then returns raw data for each available pattern. These raw data (e.g., the amount of times a student visited a certain type of learning object or the number of minutes a student spent on average on certain types of learning objects) are then transformed to ordered data based on predefined thresholds from literature. Ordered data are used to indicate whether the occurrence of a particular behaviour pattern is high, medium or low (or information about this pattern is not available). Subsequently, these ordered data are related to how the pattern affects a learning style of a student by using four values: 3 indicates that the student's behaviour gives a strong indication for the respective learning style, 2 indicates that the student's behaviour is average and therefore does not provide a specific hint, 1 indicates that the student's behaviour is in disagreement with the respective learning style, and 0 indicates that no information about the student's behaviour is available.

By summing up these values for each pattern that is relevant for a particular learning style dimension and dividing these values by the number of patterns that include available information for that dimension, a measure for the respective learning style dimension is calculated. This measure is then normalised on a range from 0 to 1, where 1 represents one pole of the dimension (e.g., active) and 0 represents the other pole of the dimension (e.g., reflective).

This calculation process is very similar to how learning styles are calculated in the learning style questionnaire but uses information from students' behaviour instead of asking students explicitly about their preferences. The approach has been introduced and successfully evaluated by Graf et al. (2009), using a predefined set of patterns for detecting learning styles in learning management systems.

The results of the *Learning Style Calculation Module* are four values between 0 and 1, each value representing the calculated learning style on each of the four dimension of the FSLSM. These results are stored in the Student Model as intermediate results, representing the students' current learning styles identified at one particular point of time from their behaviour while learning in the system. In the subsequent sections, these values will be referred as *ls_behaviour*. After the calculation and storage of these values is completed, the *Learning Style Calculation Module* notifies the *Dynamic Student Modelling Module*.

2.5 Data Extraction Module

The *Data Extraction Module* connects to the learning systems' database (or other sources of log data) and extracts data of available patterns in the learning system. Since the data extraction is dependent on the structure of the learning system's data sources and where particular data are located, this module is system-dependent and has to be adjusted to each learning system that integrates the introduced architecture.

Once the *Data Extraction Module* receives a request from the *Learning Style Calculation Module* to collect data from a particular student, it extracts and returns the data of the available patterns to the *Learning Style Calculation Module*.

2.6 Dynamic Analysis Module

This module is responsible for analysing how the learning styles calculated from students' recent behaviour by the *Learning Style Calculation Module* change over time and whether these changes should lead to a change in the learning styles stored in the students' cognitive profiles. For deciding whether such a revision should be done, two partially conflicting objectives have to be reached. On one hand, the currently stored information in the cognitive profile should reflect the current learning styles of students as good as possible and therefore should be updated as soon as a revision can be done. On the other hand, deviations of students' behaviour have to be considered and the student modelling approach should avoid situations where the learning styles of students are revised and then briefly afterwards this revision has to be reversed.

The *Dynamic Analysis Module* integrates an approach that has been introduced and evaluated by Graf and Kinshuk (2009). This approach uses the intermediate results (*ls_behaviour*) identified by the *Learning Style Calculation Module* as input data, representing the students' learning styles over time calculated based on their behaviour. In order to decide whether a learning style stored in the cognitive profile needs to be updated, three conditions have been formulated that have to be fulfilled. The first condition aims at updating students' learning styles as soon as a change in students' behaviour is noticed and therefore compares the currently stored learning style in the cognitive profile and the mean value of the last *A* identified learning styles (ls_behaviour), where an experiment by Graf and Kinshuk (2009) identified that 3 is a suitable number for *A*. The second and third conditions focus on detecting and considering deviations in terms of exceptional behaviour of students. The second condition looks into the difference between the currently identified learning style ($ls\_behaviour_t$) and the previously identified learning style ($ls\_behaviour_{t-1}$) in order to detect exceptional behaviour. Once exceptional behaviour has been detected, the third condition investigates whether the next identified learning style goes significantly towards the learning style stored in the cognitive profile or shows again exceptional behaviour (which then can indicate a change in a student's learning style rather than exceptional behaviour). If all conditions point to a change in the student's learning style rather than exceptional behaviour, the stored learning style in the cognitive profile is updated by the mean value of the past *A* identified learning styles.

2.7 Student Model

The *Student Model* aims at storing several types of information about students. First, it stores the cognitive profile of students, which includes the four values of students' learning styles based on the four dimensions of FSLSM. This information can be used by adaptivity modules, which can access the students' cognitive profiles and use the information in these profiles, for example, to provide students' with adaptive and personalized recommendations as well as with courses, learning objects, and/or learning activities that match students' learning styles.

Besides the cognitive profile, the *Student Model* also stores data about students' activity level, past data from the cognitive profile, intermediate results from the *Static Student Modelling Module* including data from the questionnaire, and intermediate results from the *Learning Style Calculation Module* which represent the identified learning styles over time based on students' behaviour.

3. APPLICATION OF THE ARCHITECTURE IN A LEARNING SYSTEM

The above described architecture has been implemented for an online learning system. A *Notification Mechanism* has been integrated in the learning system and the *Data Extraction Module* has been adjusted to the learning system's data sources and the available patterns that can be extracted in the learning system. Furthermore, an *Adaptivity Module* has been developed that uses the information from the cognitive profile in order to make students aware of their learning styles as well as provide them with personalized suggestions on how to improve their learning based on the elements available in their courses.

In the following subsection, the types of courses and course elements of the learning system are described. The next subsection describes how the information in the cognitive profile is used to provide students with adaptive feedback.

3.1 Course Structure and Available Behaviour Patterns

The learning system includes two types of courses: courses that focus on assessment only and courses that focus on learning and assessment. The assessment-only courses consist of exercises, quizzes and a study guide. *Exercises* are mainly for practicing the learned material through theoretical or practical questions which are randomly composed and can therefore be used by the students as often as they want to practice. An exercise can consist of multiple parts where each part includes a question. After a student answered the question, his/her answer is automatically marked and feedback about the correct answer is provided. *Quizzes* are more comprehensive tasks which are intended to be solved by students at the end of a section or course. Students are usually given a certain time limit (e.g., 60 minutes) for solving all the theoretical and practical questions within the quiz. Once a student submitted the quiz, his/her answers are automatically marked and feedback about the correct answers is provided. The *study guide* is a page that includes information about a student's progress in the context of all the concepts of the course.

The courses for learning and assessment also include exercises, quizzes and a study guide but they additionally use elements for presenting learning material. Therefore, each course has chapters and each chapter consists of sections. Each chapter and section includes an *outline*, which consists of a short introduction, and chapter outlines additionally include a description of the learning objectives. Each section has a *lesson* which consists of several *pages of learning material*, *applied self-assessment questions* which are very practical oriented questions that allow students to apply the learned knowledge in practise, and *theoretical self-assessment questions* which are theoretical questions about the learned material. Furthermore, a section can include *activity-related questions* which facilitate experimentation by providing immediate feedback even to parts of students' answers. In addition, a section can have a *case study* which is a large practical problem with several steps to solve.

Table 1 – Patterns of behaviour from course elements of the learning system
(* marks patterns which are related to parameters such as overall number of actions, solved questions etc.)

Pattern name	Description of patterns	act/ref	sen/int	seq/glo
exercise_stay	avg. time spent on solving an exercise question	ref	sen	
exercise_visits	avg. number of attempts to solve an exercise question	act		
exercise_performance_increase	avg. rate of grade increase on exercise questions	ref	sen	
exercise_performance	avg. final grade on exercise questions		sen	
exercise_stay_results	avg. amout of time spent for studying the feedback of exercise questions	ref	sen	
exercise_sequence_skip	number of time of skipping an exercise question*			glo
exercise_sequence_back	number of times of going back to a previous exercise question*			glo
quiz_sequence_revise	number of times of re-entering a quiz*		sen	
quiz_stay	percentage of time took on avg. for submitting a quiz		sen	
quiz_stay_results	avg. amount of time for studying the feedback of a quiz	ref	sen	
studyguide_visits	number of visits of the study guide*			glo
outline_visit	number of visits of outlines*			glo
outline_stay	avg. amount of time spent on outlines	ref		glo
content_visit	number of visits on content pages*	ref		glo
content_stay	avg. amount of time spent on content pages	ref		
content_back	number of times of re-visiting a content page*			glo
content_skip	number of times for skipping content pages*			glo
asa_solution_visit	number of visits of solutions of applied self-assessment questions*		sen	
asa_solution_stay	avg. amount of time spent on solutions of applied self-assessment questions	ref	sen	
tsa _solution_visit	number of visits of solutions of theoretical self-assessment questions*		sen	
tsa _solution_stay	avg. amount of time spent on solutions of theoretical self-assessment questions	ref	sen	
tsa_solution_back	number of re-visits of a solution in the same theoretical self-assessment page*		sen	glo
activityquestions_visit	number of visits of activity pages*	act	sen	
activityquestions_instances	avg. number of attempts tried for each activity page	act	sen	
activityquesitons_stay	avg. amount of time spent on an attempt of activity-related question	ref	sen	
casestudy_visit_same	avg. number of visits of a case study question	act		
casestudy_visit_diff	percentage of solved case study questions		sen	seq
casestudy _stay	avg. amount of time spent on a case study question	ref	sen	

Table 1 provides an overview of the considered patterns based on the course elements mentioned above. The last three columns indicate which learning style is related with a high occurrence of the respective pattern. The pattern and their indications are based on the learning style literature (Felder & Silverman, 1988) as well as on the literature about detecting learning styles from behaviour patterns (Graf et al., 2009).

The patterns in Table 1 were considered in the collection of patterns in the *Learning Style Calculation Module* and the *Data Extraction Module* was built based on those patterns.

3.2 Providing Adaptive Feedback based on Learning Styles

The purpose of student modelling is to identify and frequently update information about students which can then be used to provide students with personalized courses, learning material, learning objects/activities, and/or recommendations based on this identified information. Therefore, the introduced architecture is intended to be combined with *Adaptivity Modules* which access the student model, retrieve students' learning styles and use the information about students' learning styles to provide students with adaptivity. Such modules have strong interdependence with the learning system since they have to consider the characteristics of a course, such as the available types of learning objects/activities, in order to provide students with adaptive courses and/or recommendations.

An *Adaptivity Module* has been developed that aims at providing students with adaptive feedback based on their learning styles. This feedback is shown within the *study guide* and consists of three parts. First, students are presented with their learning styles on each of the four dimensions of FSLSM, as they have been identified through static and dynamic student modelling. A five-item scale is used for each learning style dimension, distinguishing for example between a strong active, moderate active, balanced, moderate reflective and strong reflective learning style. Second, each of the learning styles of a student (i.e., active or reflective; sensing or intuitive, etc.) is explained in more detail, pointing out typical characteristics, strengths and weaknesses of students with the respective learning style in a general learning context. Third, students are provided with personalized learning advices, dealing with suggestions on how to learn more effectively based on their individual learning styles and considering the types of learning objects available in the student's course, distinguishing between assessment-only courses and courses that focus on learning and assessment.

4. DISCUSSION AND CONCLUSIONS

This chapter introduced an architecture for dynamic student modelling of learning styles based on the Felder-Silverman learning style model, aiming at building and frequently updating a cognitive profile of students' learning styles based on students' behaviour in an online course. For a faster initialisation of the cognitive profile, a static student modelling approach in form of a learning style questionnaire has been added to the architecture which can be used by students optionally.

The architecture is developed in a generic way so that it can be integrated into different learning systems. Only two of the modules in the architecture require modifications if integrated in different learning systems. These two modules are the *Notification Mechanism* which sends notifications to the dynamic student modelling approach once a student conducts an action in the learning system as well as the *Data Extraction Module* which requires information on how and where data are stored in the learning system. Furthermore, *Adaptivity Modules* can be added to the architecture which aim at providing adaptivity and personalization in the learning system based on the identified learning styles of students, and are therefore system-dependent as well.

The application of the introduced architecture has been demonstrated for an online learning system, considering the different types of courses and types of learning objects/activities within this system. Furthermore, an *Adaptivity Module* has been developed which presents students with adaptive feedback by making students aware of their learning styles, providing explanations on each learning style of a student in a general learning context, as well as presenting students with personalized advices on how to learn more effectively given the learning styles of the student and the available learning objects/activities in the student's course.

Adding dynamic student modelling of learning styles to learning systems has two advantages over using a static approach: (1) it enables the system to incrementally learn students' learning styles and identify exceptional behaviour of students which can be excluded from the updating process, and (2) it makes it possible to identify changes in students' learning styles over time and updating the cognitive profile respectively. Both advantages lead to more accurate information about students' learning styles and therefore facilitate more effective adaptivity in learning systems. By developing an architecture that can be easily used by different learning systems with few adjustments to the respective systems, dynamic student modelling can enable these systems to provide adaptivity based on students' learning styles, for which benefits such as less study time to achieve on average same grades (e.g., Graf & Kinshuk, 2007) and higher student satisfaction (e.g., Popescu, 2010) have been demonstrated.

Future work will deal with developing and providing additional adaptivity modules for the online learning system described in this paper, such as automatically modifying courses, learning objects, and/or learning activities in order to make them fit to students' learning styles. Furthermore, the collection of patterns is planned to be extended (e.g., with patterns about students' navigational behaviour in a learning system as well as with patterns related to additional types of learning objects/activities) and the application of the architecture to other learning systems such as learning management systems is planned. In addition, a qualitative evaluation is planned in order to investigate students' satisfaction with the provided adaptivity.

5. ACKNOWLEDGEMENTS

The authors acknowledge the support of NSERC, iCORE, Xerox, and the research related gift funding by Mr. A. Markin.

6. REFERENCES

Bajraktarevic, N., Hall, W., & Fullick, P. (2003). Incorporating learning styles in hypermedia environment: Empirical evaluation. In P. de Bra, H. C. Davis, J. Kay & M. Schraefel (Eds.), *Proceedings of the Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems* (pp. 41-52). Nottingham, UK: Eindhoven University.

Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction, 6* (2-3), 87-129.

Carver, C.A., Howard, R.A., & Lane, W.D. (1999). Addressing different learning styles through course hypermedia. *IEEE Transactions on Education, 42* (1), 33-38.

Cha, H.J., Kim, Y.S., Park, S.H., Yoon, T.B., Jung, Y.M., & Lee, J.-H. (2006). Learning style diagnosis based on user interface behavior for the customization of learning interfaces in an intelligent tutoring system. In M. Ikeda, K. D. Ashley & T.-W. Chan (Eds.), *Proceedings of the 8th International Conference on Intelligent Tutoring Systems, Lecture Notes in Computer Science* (pp. 513-524). Berlin, Heidelberg: Springer, Vol. 4053.

Felder, R.M., & Silverman, L.K. (1988). Learning and teaching styles in engineering education. *Engineering Education*, *78* (7), 674–681.

Felder, R.M., & Soloman, B.A. (1997). *Index of Learning Styles questionnaire*. Retrieved 14 March, 2011, from http://www.engr.ncsu.edu/learningstyles/ilsweb.html.

García, P., Amandi, A., Schiaffino, S., & Campo, M. (2007). Evaluating Bayesian networks' precision for detecting students' learning styles. *Computers & Education, 49* (3), 794-808.

Graf, S., & Kinshuk (2007). Providing adaptive courses in learning management systems with respect to learning styles. In G. Richards (Ed.), *Proceedings of the world conference on e-learning in corporate, government, healthcare, and higher education (e-Learn 2007)* (pp. 2576-2583). Chesapeake, VA: AACE Press.

Graf, S., & Kinshuk (2009). An approach for dynamic student modelling of learning styles. *Proceedings of the International Conference on Exploratory Learning in Digital Age (CELDA 2009)* (pp. 462-465). Rome, Italy: IADIS press.

Graf, S., Kinshuk, & Liu, T.-C. (2009). Supporting teachers in identifying students' learning styles in learning management systems: An automatic student modelling approach. *Educational Technology & Society, 12* (4), 3-14.

Khan, F.A., Graf, S., Weippl, E.R., & Tjoa, A.M. (2010). Identifying and Incorporating Affective States and Learning Styles in Web-based Learning Management Systems. *International Journal of Interaction Design & Architectures, 9-10*, 85-103.

Kinshuk, & Lin, T. (2004). Cognitive profiling towards formal adaptive technologies in web-based learning communities. *International Journal of WWW-based Communities, 1* (1), 103-108.

Kuljis, J., & Liu, F. (2005). A comparison of learning style theories on the suitability for elearning. In M. H. Hamza (Ed.), *Proceedings of the IASTED Conference on Web Technologies, Applications, and Services* (pp. 191-197). Calgary, Alberta: ACTA Press.

Lin, T. (2007). *Cognitive Trait Model for adaptive learning environments.* PhD thesis, Massey University, Palmerston North, New Zealand.

Özpolat, E., & Akar, G.B. (2009). Automatic detection of learning styles for an e-learning system. *Computers & Education, 53* (2), 355–367.

Popescu, E. (2010). Adaptation provisioning with respect to learning styles in a web-based educational system: An experimental study. *Journal of Computer Assisted Learning, 26* (4), 243-257.

Shang, Y., Shi, H., & Chen, S.-S. (2001). An intelligent distributed environment for active learning. *ACM Journal of Educational Resources in Computing, 1* (2), 1-17.