In Proceedings of the ICDE-World Conference 2001 (Electronic Proceeding)

Approaches to Quality Management for Developing WBT Courses

Huberta Kritzenberger, Michael Herczeg
Institute for Multimedia and Interactive Systems
University of Luebeck
Seelandstr. 1a, D-23569 Luebeck, Germany
<kritzenberger, herczeg>@imis.mu-luebeck.de

Abstract: The focus of this paper is on the process of producing web-based training (WBT) courses. The development of course material is regarded as a software development process, which suffers from similar problems like other software production processes as well, e.g. incomplete or inappropriate conceptualization of the task and of user needs in the beginning. From a software-ergonomic point of view this lack of relevant design knowledge is a problem, because this information is needed by the developers of the WBT course in order to design and realize a usable and consistent course which supports the learning process and which fits user needs. Otherwise, the result of the course unit, as a result of the production process, is in danger to lack usability and hardly support the learning process of the intended users.

Furthermore, there is another problem teams developing WBT courses suffer from. Information on user needs and various aspects of tasks are contributed by different team members which mostly are separated in place and time. This expert knowledge has to be collected, stored and communicated, so that it can be (re-)used and further completed during all stages of the development process. A solution to this problem is proposed by the authors of this paper. They introduce an xml-based repository and cooperation platform, which is partly realized as a prototype support development teams and to help these teams to overcome some of the problems of WBT course development teams.

1 Introduction

Developing WBT courses, e.g. for a virtual university, does not only to transfer from one medium to another medium and thereby use a new technological platform. Moreover, the development of WBT courses needs an integration into an adequate media production process in order to manage the production process itself and to develop user and task adequate courses.

The first problem is an increasing scope of what user and task adequacy can mean. In the context of lifelong learning it is the fact that learning situations will be pervasive in people's life. In consequence, the division between education, work life and leisure time will fade (cf. Herczeg 1997). Therefore, the requirements on the usability of educational courses will grow as the courses will be used by many different user groups with different needs and in various contexts and use situations. Although new technologies increase the availability of course material, it does not automatically enhance the usability and make the courses fit the increased user needs. In order to meet the user needs it will be necessary to adapt the course material accordingly. However, there are not many usable solutions for the problem of user and task adequacy and how to achieve it. Sometimes adaptation is put forward as a solution to this problem.

Adaptation and adaptability of courseware is mainly discussed in the context of tutorials and adaptive hypermedia systems. The system compares an inferred user's mental model of the application domain (student model) with the system's internal domain model and adapts system parameters, like the instruction strategy, the presentation of the content or the hypermedia link structure. Techniques like link-hiding or link annotation are wide-spread (cf. Brusilovsky 1998; Brusilovsky/Schwarz/Weber 1996; Calvi/DeBra 1997). Other proposed techniques, e.g. link sorting, are known as leading to incomplete or poor mental models and therefore are seldom used (cf. Calvi

1997). There are well-known problems with the representation of huge data bases or inflexibility for the planning of instructional strategies within these systems (cf. Woolf 1992). Another question would be, how adequately these adaptations can meet user needs at all. From a psychological point of view (cf. Ulich 1994), a proper level of demand, which the learning task imposes on the user, is most important to cause positive mental states like motivation, happiness and increase of performance. An inappropriate level of demand, however, will cause negative mental states and decrease performance. Therefore, a individually level of demand from the learning task has to be individually balanced in a dynamic context (cf. Herczeg 1997; Kritzenberger/Herczeg 2000). As this is a highly individual factor, the system should allow the user some scope of action with respect to steps of learning, the time needed for learning processes, goals and sub-goals and so on.

Unfortunately, options for the use of courseware are prepared at a very early point in time of the development process, that is just from the first step when the pedagogic goals, the organizational context and the target groups are thought over. Although these aspects determine the options for system realization, they are mostly neglected, apart from a few exceptions (cf. Nikolova/Collis 1997; De la Teja/Longpré/Paquette 2000). Current development processes of courseware often suffer from missing or incomplete user needs and task analysis. Sometimes this analysis is not done at all. If it is done, the problem of completeness, sorting and storing of the collected data and (re-)use of the data by different members of the development team at different phases of analysis, design and production is mostly unsolved. There is hardly any task-adequate support for development teams. Authoring tools, which are sometimes proposed, only support the production process itself and are therefore not useful for the problems of development teams.

In this paper we propose a framework which can be used as a repository and cooperation platform for development teams, which are distributed in time and place. It helps to improve and manage the development process and thereby to improve the quality of the WBT course.

2 Development of WBT-Courses in Teams

The experience of the authors is based on two projects in the context of development of web-based course material for virtual universities.

The one project is called “Virtual University of Applied Sciences” www.vfh.de (period of duration 1998-2003). It aims at establishing a location independent university with a curriculum for computer science of multimedia systems and for business engineering (Bachelor, Master). The authors of this paper are involved in the production of web-based courses, in the design of user-adequate learning spaces and in the support of the design process. Their focus is on usability recommendations and quality management for the course material during the development process. Other aspects like teaching strategies, learning processes or technical issues concerning the course production are supervised by other dedicated consulting groups within this project.

The other project “Distance Education in Medical Computer Science” (started in January 1999) aims at providing a complete course of studies for the specialization of students in medical computer science. The course is offered at a virtual university (Hagen). Our responsibility is to transfer the linear text documents (mostly MS-Word format) into hypermedia networks and multimedia courses.

The production of courseware is an iterative process, which consists of the following stages: analysis, requirements specification, design, implementation, evaluation and maintenance. In all these stages there are data produced which are needed in the following stages by different members of the software development team. It is normally hard to share and manage the software development models (i.e. information produced in each state of the development process) within a distributed development team. In our approach all members of the design team, which in the development process of learning environments consist of content author, designer (which may be conceptioners, pedagogues, experts in software ergonomics), producer and quality manager (see figure 1), use a relational database with a web front-end. This database should include all the context information as well as the content of the teaching units. In a first step this information is only presented to the involved persons but no contents are automatically generated.

For database organization object oriented techniques are used. Classes of information like “user attributes”, “organizational requirements”, “content”, etc. are identified and then may be freely combined into “views”. These

views are role dependent, for example the content author may concentrate on the analysis data and the contents. Whereas the producer, who has to implement the learning unit, may need additional design rationales added by the designer. A quality manager can base her or his evaluation of the system on the requirements identified and documented during the analysis phase.

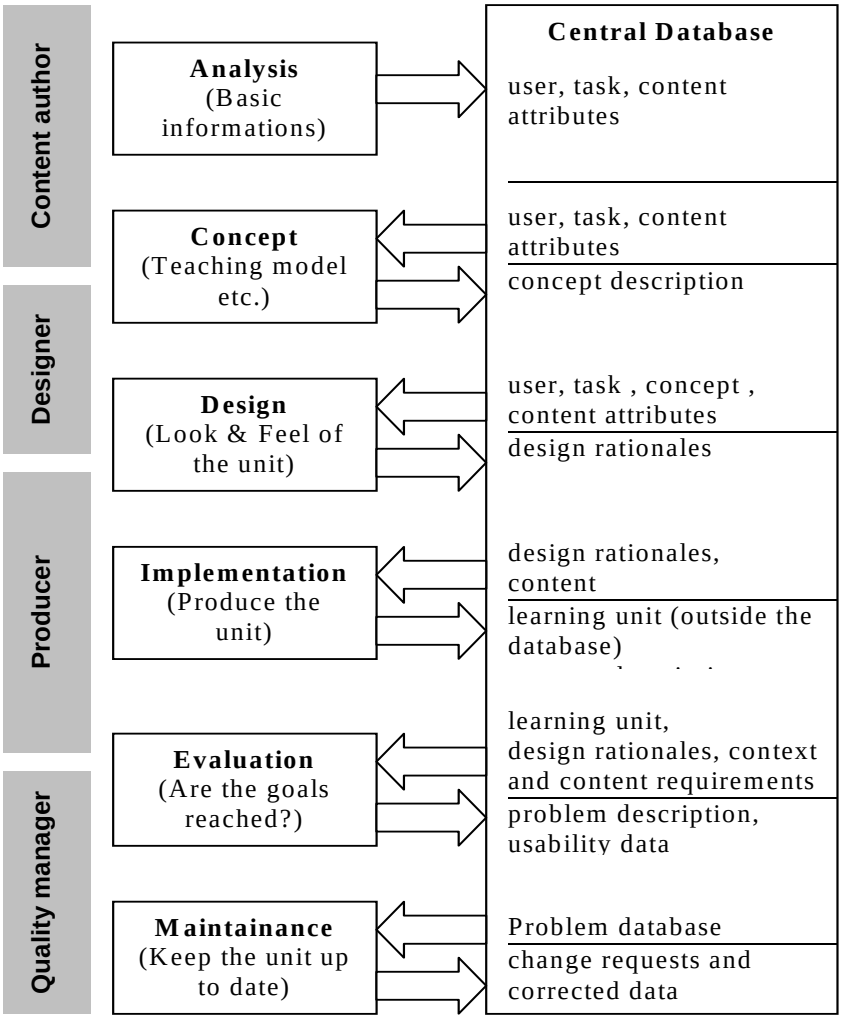

Figure 1: An iterative development process for educational systems (e.g. WBT courses). (cf. Hartwig/Kritzenberger/Herczeg 2000).

As the size and the number of the learning units is difficult to handle, techniques of the OOA are used, e.g. abstraction, inheritance and generalization. If for instance many different user groups are to be considered, they may be ordered hierarchically: “all users”, “users of the teaching unit”, “a special group within these user group”. Attributes of the most general object (“All users”) are inherited by all following objects. This data organization allows views, e.g. a designer’s view includes all attributes, from the general to the specific ones. Furthermore, if it is needed by the developer, for each phase there is appropriate additional information available, like design rationales and the related context information.

Another advantage of other OOA techniques, like underspecification and refinement, is that they make the handling of large object sets easier. For example, they allow all participants to start with rather raw data and to refine them during the during the development process. Another advantage of the approach is that the use of a database supports the complete lifecycle of the course units, because all information and design rationales are not only available at the time of production, but also available again for maintaining or updating the course. If there are HTML-based contents, e.g. coming as updates from the content authors, they are to be included into the courses with little effort, keeping the connection of each unit to the position and role in the database. Updating the content in the database automatically updates the course and avoids inconsistencies.

3 A Framework for User- and Task Modeling for Learning Environments

The first step in the development of learning environments, e.g. WBT courses, is the analysis of pedagogic goals, the organizational context, and the characteristics of the intended user group (learners). If this kind of analysis is done accurately, a lot of data are gathered. In order to become able to make sense of these data in context, it becomes necessary to store the collected data in a structured way. The following framework seems appropriate for structuring, storing and applying these data in the course of a WBT course development process.

3.1 A Framework Layer

The following framework (cf. Herczeg 1999) serves as a generic model, which allows to specify the specific characteristics of user groups as well as the conditions of use of the learning environment. It consists of the following application independent entities: Managed (Learning) Object, Task, Role, Agent and Tool.

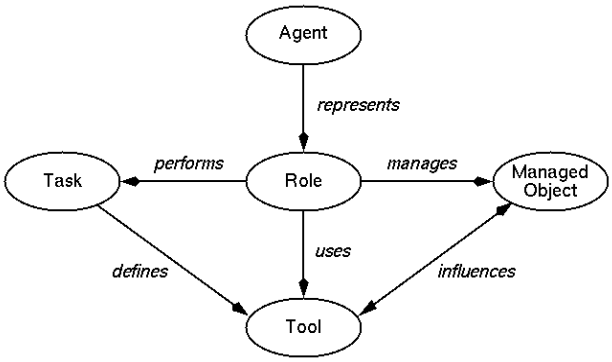

Figure 2: Framework Layer (cf. Herczeg 1999: 29) as possible basic structure serving for an analysis

In the context of modeling learning environments, like WBT courses, these entities are understood in the following way:

Managed (Learning) Object: Learning objects are stored as domain modules (content, knowledge schemes etc.). They are used by different learners with specific characteristics. As users are different, learning objects have to be delivered in learner-adequate ways. It has to be defined, what kind of method fits for teaching which each

managed learning object, that is they influence the choice of an adequate tool. Different user groups may use different managed objects or different sub-sets of managed objects.

Task: Tasks have to be performed by the learner or by the system. In the context of the learning environment they can be understood as steps performed to complete a learning process. On a lower level of task decomposition the naming of the tasks depends on specific pedagogic models. Such tasks could be for example learning of facts, elaboration strategies, cognitive skills, cognitive strategies (like problem solving), etc.

Role: It means the context given by a specific learner. The role realizes of concrete user characteristics (cf. Kritzenberger/Herczeg 2000).

Agents: Specific target groups or user groups for the environment can be specified here. User groups for learning environments serving the purpose of lifelong learning may be diverse. For example, for WBT courses offered in the learning environment of a virtual university there may be a user group 1, which does basic studies comparable to traditional universities or schools. There may be a user group 2, which is interested in further education. Another learner group 3 may want training on the job and fill knowledge gaps. A learner group 4 may be characterized by a concrete problem, which has to be solved by acquiring new knowledge. In order to do so, they need to study only specific knowledge modules or learning objects. Furthermore, a learner group 5 may want to explore the learning modules without any shortage of time. Of course, other user groups may be relevant and characterized as well. The definition of user groups is very important for the development of a learning environment, because teaching always means to address a target group. Therefore, the means and the teaching level have to be adapted accordingly.

Tool: describes the activities which are available for the execution of a task. If the task structure is rather elaborated this can be understood as a network of teaching strategies, which form a complete teaching model. Leading questions are for example “Which method(s) serves best for teaching what kind of content?” (see connection to the entity managed learning object). The task not only contain a pedagogic model in the sense of a teaching model, but also a design model, which is specified from a media psychological or ergonomic point of view. On a more specific level, tools may be classified as “draw the learners attention on the learning object”, “activate knowledge”, “initiate self-directed learning”, “apply knowledge in different contexts”, “initiate transfer of knowledge”, “control achievement of learning goals”, “give feedback”. Each tool gets connected with the relevant learning modules (managed learning objects) and with the task, which specifies the internal or external conditions for learning.

3.2 User and Task Modeling within the Framework

The model specified above helps to structure all relevant parameters of the intended users (learners), of the use context and use process (cf. Herczeg 1994). Most of the descriptions have been collected in an analysis phase. They have been contributed by different team members with different professional backgrounds (compare figure 1). In the beginning of the development process many of these data are vague or imprecise and may be refined later on. The collection of data has been ordered with the framework described in the section above. This order helps to refer data belonging together (which are connected by links) and needed together. It also helps to link data which should be viewed together, because they form a semantic content of development knowledge. As link categories and use of links can be freely defined, the structure is very flexible.

Example: Typical Situation in the Development Process for WBT-courses

For the development of a course of studies in the Project “Distance Education in Medical Computer Science” 24 WBT courses for a complete course of studies are developed, which will be offered at a

virtual university. A team of developers with different background knowledge is working on it and cooperating during the development process.

The content authors deliver the content as Word-Documents. Then the documents are transferred into Web-documents, which is done by media producers. This transfer means translating the documents into hypermedia and multimedia applications. Media adaptation offers new teaching options, which the content authors on the production basis of Word-Documents might not have thought about. But there are other members in the development team of the WBT course, who are aware of the new opportunities. There are designers, conceptioners, pedagogues, quality managers (e.g. specialists in software ergonomics), who contribute to a conceptualization and media adequate transfer of the course unit to web presence. The whole team works distributed in time and place. Their contributions have to be stored and used on demand by others. As the knowledge for the development process is distributed over the team members, it must be collected and communicated at the proper stage in the development process and in an adequate way.

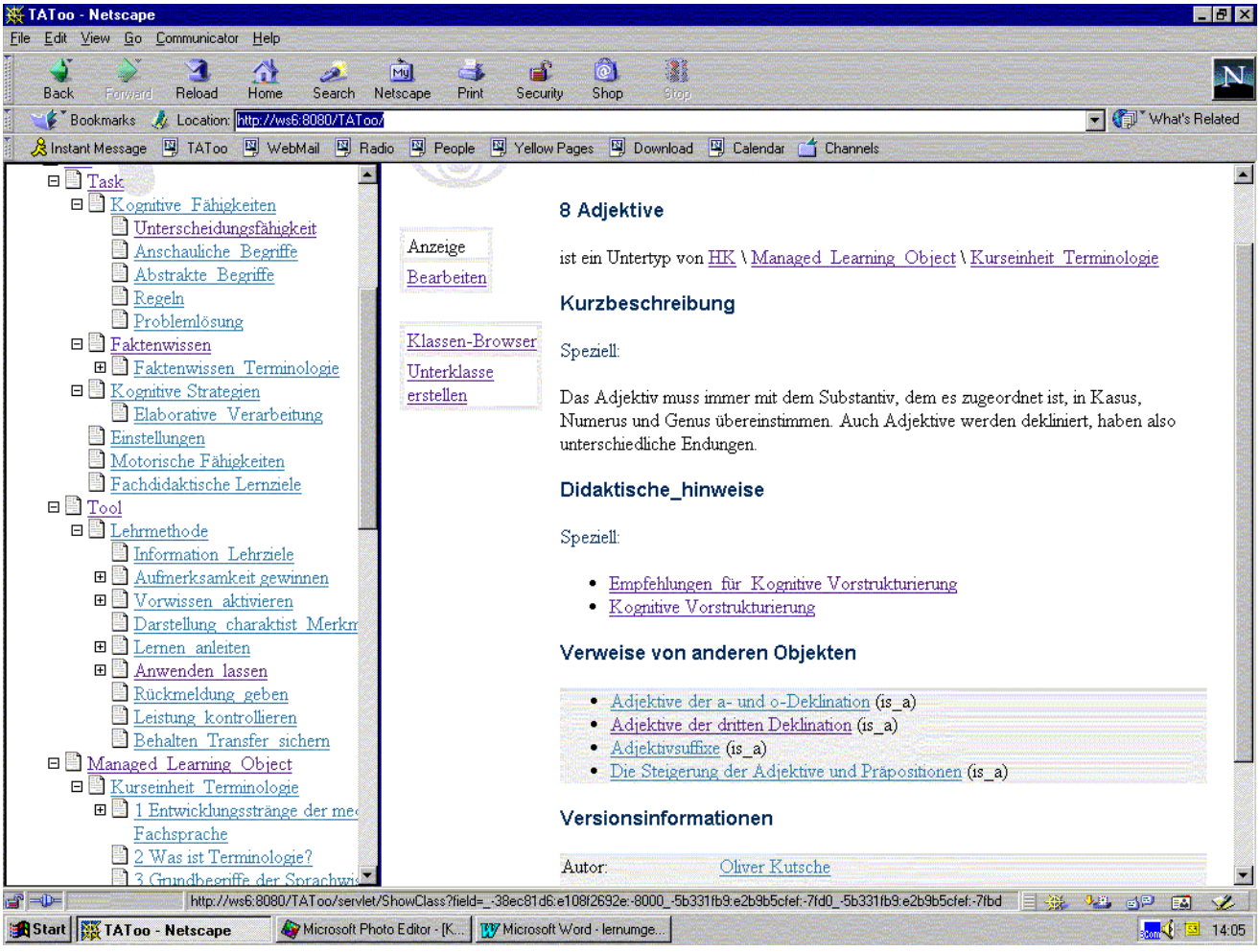

Figure 3: Screenshot of user and task modeling for a WBT course on terminology in the project “Distance Education in Medical Computer Science”.

Figure 3 shows a screenshot with a view on the user and task model from the modeling for a WBT course in our project “Distance Education in Medical Computer Science”. The screenshot should demonstrate a solution to the problem described in the example above.

The WBT course in the example above deals with medical terminology. The screenshot shows the view on the module on adjectives. On the left part of the screen the structure can be viewed in more detail (task, tool, managed object, role), which is helpful for constructing of the meta-information structure and knowledge management for the repository. On the right part of the screen a view on the database is given, which shows how meta-information on the object "adjectives" can look like. First of all, the view informs about the intended structure for the course module, which the content author regards as leading structure from a didactic point of view (see section: "Verweise von anderen Objekten"). The sub-knowledge modules, which are specified there, are about different kinds of declinations (a-, o- and third declination), the suffix formation and the comparative formation for adjectives. These expert knowledge units are contributed by the content author together with a short introduction text (see "Kurzbeschreibung"). If the modules should be given in a specified order, which may be different for user groups, the content author can specify this as well. But this information would not be enough for a developer who wants to build a WBT course. Additional knowledge is needed, which comes from another expert in the development team. Relevant further information is on pedagogic strategies for knowledge delivery, which can support the learning process or motivate the learner. This kind of expert knowledge is specified in other entities and can be linked to appear under another section here. In the example of figure 3 there is a link to module from the task entity, which was named by the author as "Didaktische Hinweise". The author of this didactic section gives recommendations and examples on how to build a text structure which will support the learner's information processing. In this case it is on how to help the learner to pre-structure the information, for example the developer is informed about adjunct questions and advance organizer and so on.

There may also be other relevant sections with relevant information for the development process, e.g. information on media, their use and design, e.g. information on text presentation on the screen (like structured writing, formatting, typography and so on).

The advantage of this tool is, that the information which is contributed by members of the development team at different phases of the development process is stored in a way which gives the meta-information a usable structure to be stored and communicated in a task-oriented way. As this task-oriented way is also based on OOA techniques and an XML-based data structuring (cf. Kutsche 2000), it is highly flexible with respect to the developers structuring, retrieval and selection requirements for views.

4 Outlook

WBT course production has been identified as software development or media production process. Software development processes are done by a team in stages, each needing specific expert knowledge which has to be stored and communicated. The distribution in time and place of the team members, the vagueness and amount of analysis data make the availability and communication of the expert knowledge even more complicated. This process needs support. Supporting development teams with a cooperation platform, as described above, has been identified as an important aspect of quality management for developing WBT courses.

References

Brusilovsky, P. (1998): Methods and Techniques for Adaptive Hypermedia. In: Brusilovsky, P.; Kobsa, A.; Vassileva, J. (Eds.): Adaptive Hypertext and Hypermedia. Dordrecht, Boston, London: Kluwer Academic Publishers, pp. 1-43

Brusilovsky, P.; Schwarz, E.; Weber, G. (1996): ELM-Art: An Intelligent Tutoring System on the World Wide Web. In: Proceedings of ITS ´96, pp. 261-269

Calvi, L. (1997): Navigation and Disorientation: A Case Study. Journal of Educational Multimedia and Hypermedia. Vol. 6 (3/4), pp. 305-320

Calvi, L.; DeBra, P. (1997): Improving the Usability of Hypertext Courseware through Adaptive Linking. In: Proceedings of the 8th ACM Conference on Hypertext. HYPERTEXT `97. Southampton, UK, April 1997, pp. 224-225

De la Teja, I.; Longpré, A.; Paquette, G. (2000): Desinging Adaptable Learning Environments for the Web: A Case Study. In: Proceedings of ED-MEDIA 2000. World Conference on Educational Hypermedia, Multimedia and Telecommunications. 26th June – 1st July 2000. Montréal, Canada. AACE: Association for the Advancement of Computing in Education, pp. 243-248

Hartwig, R.; Kritzenberger, H.; Herczeg, M. (2000): Course Production Applying Object Oriented Software Engineering Techniques. In: Proceedings of ED-MEDIA 2000. World Conference on Educational Hypermedia, Multimedia and Telecommunications. 26th June – 1st July 2000. Montréal, Canada. AACE: Association for the Advancement of Computing in Education, pp. 1627-1628

Herczeg, M. (1994): Software-Ergonomie. Grundlagen der Mensch-Computer-Kommunikation. Addison-Wesley-Longman and Oldenbourg-Verlag

Herczeg, M. (1997): Prospektive Gestaltung von neuen Lehr- und Lernsytemen im Kontext einer virtuellen Hochschule. Eingeladener Vortrag zum Symposium der „Virtuellen Fachhochschule“ am 16. Juni 1997 in Lübeck.

Herczeg, M. (1999): A Task Analysis Framework for Management Systems and Decision Support Systems. In: Proceedings of AoM/IaoM. 17. International Conference on Computer Science. San Diego, California, 6th – 8th August 1999, pp. 29-34

Kritzenberger, H.; Herczeg, M. (2000): Completing Design Concepts for Lifelong Learning. In: Proceedings of ED-MEDIA 2000. World Conference on Educational Hypermedia, Multimedia and Telecommunications. 26th June – 1st July 2000. Montréal, Canada. AACE: Association for the Advancement of Computing in Education, pp. 1374-1375

Kutsche, Oliver (2000): Proof-of-concept der datenbank- und web-basierten Unterstützung von Entwicklungsprozessen für einen Prototypen. Studienarbeit, Informatik, Medizinische Universität zu Lübeck, August 2000

McCalla, G. (1992): The search for adaptability, flexibility, and individualization: Approaches to curriculum in intelligent tutoring systems. In: Jones, M.; Winne, P. (Eds.): Adaptive learning environments: Foundations and frontiers. Berlin: Springer-Verlag, pp. 91-

Nikolova, I.; Collis, B. (1997): Flexible Learning and Design of Instruction. Available on-line: http://193.68.242.15/122/~iliana/TDO/TDO97/CRS_MATERIAL/PEG_PAPER.HTM

Ulich, E. (1994): Arbeitspsychologie. Dritte, überarbeitete Auflage. Zürich: Vdf-Hochschulverlag und Stuttgart: Schäffer-Poeschel

Woolf, B. (1992): Towards a computational model of tutoring. In: Jones, M.; Winne, P. (Eds.): Adaptive learning environments: Foundations and frontiers. Berlin: Springer-Verlag, pp. 209-232