OPUS One
An Artificial Intelligence - Multi Agent based Intelligent Adaptive Learning Environment (IALE)

Attilio Pedrazzoli PhD, MSc , Luisa Dall'Acqua PhD

GSi Edu Research - Lugano, Switzerland – (avp,da.lu@gsiserver.com)

Abstract: This paper proposes a concept for an *Intelligent Adaptive Learning Environment (IALE)* based on a holistic *Multidimensional Instructional Design Model*, applied on OLAT, an open source, Java LMS, developed at the University of Zurich, to support student and/or groups defined as "Learning Entities" (LE). Based primarily on an Artificial Intelligence – Tutoring Subsystem, used to identify, monitor and adapt the student's learning path, considering the students actual knowledge, learning habits and preferred learning style. The proposed concept has the peculiarity of eliminating any didactical boundaries or rigid, implied course structures (also known as unlimited didactical freedom). Relying on "real time" adapted profiles, it allows content authors to apply a dynamic course design, supporting tutored, collaborative sessions and activities, as suggested by modern pedagogy. The AI tutoring facility (eTutor), coupled with the LMS, is intended to support the "human tutor" with valuable LE performance - / activity data, available from the integrated "Behavior Recorder Controller" (BRC), allowing to confirm or manually modify actions suggested by the eTutor. The student has the option to select the level of tutoring interventions or switch to a "subject matter" exercise mode if he feels, and is allowed to do so. The concept presented combines a personalized level of surveillance, learning activity- and/or learning path adaptation suggestions to ensure the students learning motivation and learning success.

Keywords: OPUS One, OLAT Learning Management System, eTutor, Artificial Intelligence based tutoring, Adaptive Learning Environment, Open Source Software, PENTHA Model

1 Introduction, vision and research finality

The proposed project is the result of a collaboration between an international group of Researchers (grouped together in a nonprofit organization called GSi Edu-Research Group) composed of Pedagogy experts, College teachers, IT professionals, Instructional Design- /Knowledge Management experts. The objective was to create, using existing open source products, an e-Learning platform, easy to use for non IT oriented Authors (Teachers), allowing to create advanced "subject matter content" based on their own pedagogical models and teaching habits.

Considering the fact that students differ in learning preferences and learning approach (such as: language, perspective, typical learning time/-involvement, interactivity type and level, learning resources, semantic density, etc.), amount and kind of prior knowledge, cognitive skills, etc., one and the same instructional content cannot provide optimal knowledge for all students. Educational theorists recognized the value of personalized instruction since late 1960's [11], but technology was not ready to deliver such type of instruction on a global scale.

The proposed platform concept allows a personalized learning approach based on the actual learning curricula of the student, taking in consideration positive or negative progress made during the completion of the learning path. Of vital importance was the necessity of a Teacher -/ Human Tutor- / Student tutoring facility, to compensate possible differences in "human" tutoring quality and involvement. The platform should be build using "best of breed" open source components, backed up by an active developer community. This platform concept is designed to transparently support known models of Instructional Design (ID) references. Actually, twenty of this models are considerate significant internationally. Some of them emphasize collaborative learning and problem solving approaches [1], other promote experiential learning [2], or content understanding [3]. As base reference, we selected the PENTHA Model [8], a multidimensional Instructional Design (mID) model defined and developed specifically for this project. It describes the specifications needed for an educational environment, able to: increase productivity and operability, create conditions for a cooperative dialogue, develop participatory research activities of knowledge, observations and discoveries ("ecological" learning environment), and customize the learning design in a complex and holistic vision of the learning / teaching process. In particular, the mID model proposes a didactical scenario evolving on five conceptual dimensions (Knowledge-, Cognitive-, Didactical-, Semiotic- and Social dimension), and defining five essential didactical functions to be performed on the selected LMS platform, like**:** *profiling action* (analysis of personal characteristics of students or LE's, their needs and expectations); *behavior recording action* (analysis of the student behavior during the learning cycle, the ability to monitor the student during collaborative activities and recognizing the completion of tasks from students participating in group assignments); *presenting action* (structuring, visualization, storytelling and re-draft of didactical sequences); *planning action* (semantic analysis of concept maps and production of flowcharts); *scanning action* (analysis of activities, associated to social- and knowledge networks).

The key to success is in the ability to provide a complete tutoring concept, represented by a combination of an "automatic tutor" (e-Tutor), covering the majority of the needed tutoring requests. Tracing the student's step-by-step solution enables the e-Tutor to provide personalized advice in the problem solving approach. Prototypically tutors provide immediate feedback on each problem solving action: recognizably correct actions are acknowledged, erroneous actions are flagged. This gives the student maximum opportunity to reason about the current problem state, by

---

[1] i.e. Constructivist Learning Environments (CLE) of Jonassen D., or Collaborative Problem Solving (CPS) of Nelson L.M.

[2] i.e. Open Learning Environments (OLE) of Hannafin L., Land S., Oliver K., or Goal Based Scenario (GBS) of Schank R., Berman T., Macpherson K.

[3] i.e. Multiple Approaches to Understanding of Gardner H, or The Elaboration Theory of Reigeluth C.M.

monitoring and assisting his/her approach, based on the "tutoring level" defined in the Learning Entity profile. Generally, the eTutor will provide feedback messages ("hints") if the student appears confused about the nature of the current problem definition or problem solving attempt. The projected platform concept recognizes three general levels of advice: a) a reminder of the current target; b) a general description of how to achieve the solution; c) a description of exactly which problem solving action steps should be taken. Each of these three levels may be represented by multiple assistance steps.

To summarize, the research finality, the described concept should be able to: *recognize* a large variety of student solutions; *diagnose* student "Subject Matter" understanding and *recommend* target oriented, optimized "learning approach adaptations"; *tailor* tutorial actions accordingly; *support* collaboration; *support* specific forms of adaptation for collaboration activities, like recommending suitable collaborators and actions; *adapt* the interface to facilitate collaboration activities (enforce specific roles and rules); *advise* students how to interact efficiently; *reasoning*, specify techniques to acquire and propose additional knowledge material about a domain or subject matter; *use* the knowledge base to solve problems in that domain or subject matter; *support* educational workflow sequences [13]. Being strictly Student centric. Focus and priorities have been set on usability, quality of service, modularity and scalability. The project relies on the experience of existing implementations like CTAT [7] and SOUL [18].

2 IALE Concept Overview

The described platform concept foresees the following three key system components: 1) an *Learning Management System (LMS) Environment*; 2) an *AI – MAS subsystem* including *Rule Engine support* for the "AI – Tutoring Environment" function; 3) a *dedicated "User Area"* for socialization.

2.1 Learning Management System (LMS) Environment

The LMS should perform the classical functionalities, like: User management; Role management; Course content erogation; advanced Group management; include an easy to use Course Editor for content creation; Achievement management; Test-/Assessment facility; support "state of the art" eLearning standards (IMS, SCORM etc.); include integrated collaboration features (like Wiki-, Forum-, Blog functionality, etc.).

The development team decided to evaluate open source, JAVA based LMS's, primarily due to the integration complexity of the AI based tutoring functionality. The LMS should not impose any pedagogical limitation to the course structure (support unlimited didactical freedom), it should allow to develop courses in any known didactical-/pedagogical model, should be based on dynamically modifiable XML structures, be modular and scalable. To facilitate the integration of a native, dynamic "learning path adaptation" the LMS should support a section / subsection based access-, execution- and visibility mechanism, supported by a parameter driven grading system. After an intensive benchmark and verification period the development team decided to select OLAT, developed at the University of Zurich [15] as the R&D LMS platform.

To fulfill all our requested features in OLAT, we developed the OPUS One extension package, which includes: a) DB based *course activity, logging / tracking facility*, interfaced with the AI – "Behavior Recorder Controller", allowing a real time, granular, learning progression analysis and immediate LE profile update. b) eTutor portlet / eTutor Administration / eTutor Assistance facility based on extended LMS "Role profiles". c) Improved OLAT course navigation with visual sequence status and course navigation flow control. d) Extended *"User Role based Homepage"* personalization, showing exclusively user related-/ owned functions like: "My courses", "My Groups", "My Roles" etc. This feature will only show Student / Author / Teacher owned resources on the personalized Homepage for easier and more direct access. e) "*Personal Notes Board"*, including the following features: *Course Notepad* – Multiple personal notes per course identified by Subject, Date/Time, Keywords; *Sharable Course Notes* accessible by same "Group" members (Project Group or Learning Group) identified by Subject, Date/Time, Keywords; *Free Form general personal notes* identified by Subject, Date/Time, Keywords. f) *"Collaborative writing facility",* (Personal-, Course-, Group based) as generic LMS function or course module, integrated into the course editor. g) *Multimedia aggregator* facility as generic LMS function or course module integrated into the course editor allowing to dynamically integrate Multimedia content as course material. h) *e-Tutor "tutoring on demand" requests*, "Walk trough" mode selection (Exercise Mode), manual learning path suggestion request functionality, driven by the actual user profile. i) *Video conferencing* (VC) *facility* including a "state of the art" Whiteboard, Chat, Desktop sharing, File sharing and Recording functionality using OpenMeeting [16] as general VC facility, DimDim [9] as collaborative course module, integrated into the course editor. The VC facility allows the dynamic creation of public-, group-, or private temporary Meeting Rooms / Auditoriums for the Videoconference function. The user access mode is defined in the OPUS One / OLAT user profile. Foreign Videoconference Members can be invited via email or personal message. j) *OPUS One / OLAT bidirectional asynchronous, multichannel, external environment wrapper*, a six agent - AI community, designed and implemented to tightly integrate external, reusable learning content into the OLAT LMS. The external wrapper agents are profile driven, able to capture data structures and data to be transferred and integrated into / from the OLAT LMS. A practical example is the integration of LAMS Content sequences [13] as generic OLAT course modules, able to pass data, as an example assessment- / test results, done on the external LAMS environment, into the native OLAT LMS "My Achievements" structure. This functionality is also used to synchronize the global Student "Grade book" into the e-Portfolio facility (Mahara Subsystem [14]).

2.2 AI Tutoring Environment

The AI - Tutor function (eTutor) was developed using the Cougaar framework [6], running autonomously on a separate platform. The architecture selected for the eTutor are AI MAS agent communities with dedicated responsibilities. The eTutor is tightly integrated (coupled) with the OLAT LMS trough a dedicated OLAT interface agent community. The eTutor implementation can be viewed as a repository of generic agents organized in a two level hierarchy:

Level-1 - Activity Management Agents, like: Learning entity agents; Subject matter agents; Presentation agents; Prediction agents etc.

Level-2 - Tutoring function performing Agents, like: LE tutoring request agents; Subject matter tutoring request agents; Hint agents; Rule access agents etc.

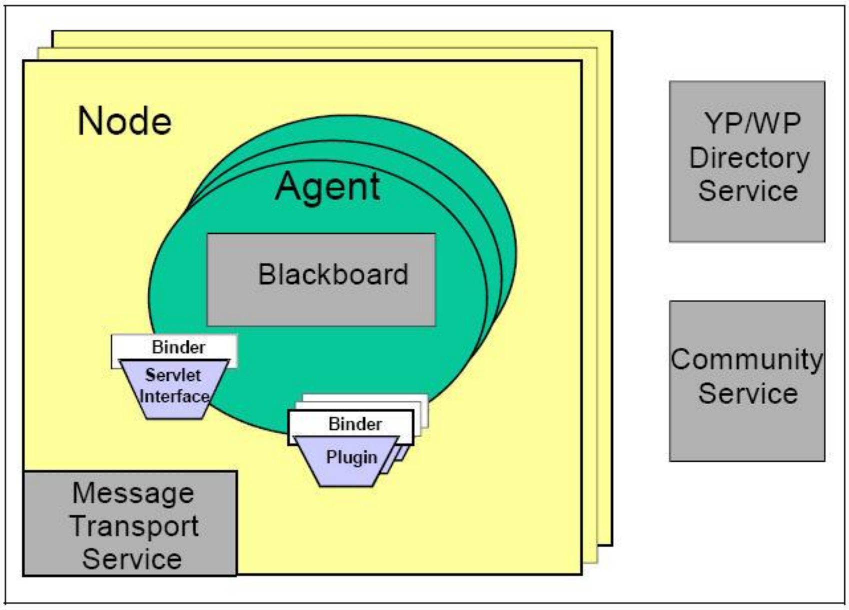

Figure 1. Cougaar Components Schematic [6]

The described eTutor Agent concept was realized according to the Cougaar Component Model (CCM), a framework that loads and manages Java software modules (called *Components*), that connect to, and interact with one another through abstract interfaces (called *Services*). (Fig. 1)

eTutor Agents take full advantage of the flexibility of the Cougaar Component Model (Fig. 2) to dynamically load components (plugins), inserting component "binder" proxies between components to mediate interactions with system services. Agent relationships are dynamically negotiated, using a hierarchical service discovery mechanism. Agents organize themselves into communities, to monitor security conditions and agent availability, allowing them to adaptively control their behaviors. See chap. 3 for a more detailed Agent concept description.

The eTutor concept includes a *"Behavior Recorder Controller*" (BR-C), able to supervise the LE during his actual learning path. The BR-C logs/tracks and evaluates learning activities according to the "subject matter" profile and the associated rules.

The *"Learning Entity Controller* (LE-C)" schedules and supervise all activities related to the LE. The LE-C checks the access- and execution credentials of the LE associating a Learning Entity Agent (LE-A) with the appropriate LE profile. When the LE selects a course, the LE-A will initialize a Subject Matter Agent (SM-A) with the appropriate "subject matter profile". The subject matter profile owns all relevant information to activate the correspondent Level-2 agents with all relevant instructions about decision Rules, exception condition, misconception detection, adaptation rules, additional study material list, tests etc. On every LE action tread, event exceptions are recorded in real time by the BR-C and registered in the LE personal profile.

2.3 OPUS One User Area based on Apache Pluto Portlet Container

The Apache Pluto portlet Container provides the environment for the OPUS One "User Access Area", a dedicated "socialization space" (sub portal), integrated into the OLAT LMS. It provides the runtime environment for portlets implemented according to the portlet API specification JSR-168/JSR-268. Conceptually, it provides the interface between the OLAT LMS and the specific portlets. Following portlets are available in the OPUS One User Area: del.icio.us API portlet to facilitate and integrate the access to the private del.icio.us user space with a possible feed (RSS) into the LE Mahara e-Portfolio subsystem [14]; Facebook API portlet (same as the del.icio.us functionality); Wordpress MU/Wordpress Buddy environment portlet, allowing users to access the global multiuser/multiblog and extended functionality tool and publishing environment with the possibility to integrate threads into the private Mahara e-Portfolio; Several other "generic" open source portlets like Google Map, Google Earth, YouTube etc. just to name a few, are available and ready to be deployed when needed.

The access to the user area is established from the personalized Student Home page on the OLAT LMS platform. The User Access Area is per concept definition not supervised.

3 AI Tutoring Agent definition and concept (Cougaar framework)

Agent definition: A Cougaar agent (see Fig. 2) contains a "blackboard" and a number of dynamically loaded components such as *plugins* and *servlets*. The blackboard is the collective memory interchange for agents. Each component is given one or more binders that may audit, authorize, or modify communications between a component and services with which it interacts. Cougaar plugins are software components that provide a specific piece of application business logic to the agent. The behavior of an agent depends primarily on its set of loaded plugins. Cougaar servlets provide a distributed Web-based user interface to Cougaar agents. The components of an agent communicate through the agent's blackboard via a publish-subscribe mechanism. For interactions with other agents, blackboard objects are transformed into messages by domain-specific *Logic Providers*. Agents developed for the tutoring system can be defined as cognitive (symbolic) agents which have a symbolic model of the environment, updating it continuously on the basis of which it plans all its actions according to an associated profile. The associated profile will determine the type and activity of the agent in question. Level-1 agents are defined as primary- or management agents, they are able to duplicate (scale) themselves and collaborate with each other. Level-1 agents are "Main Function" oriented agents performing functions like *Learning Entity supervision* or *Subject Matter related functions*. Level-1 agents have per definition a Supervisor – Role, they activate and supervise Level-2 agents to perform requested or dedicated, tutoring- or adapting specific tasks.

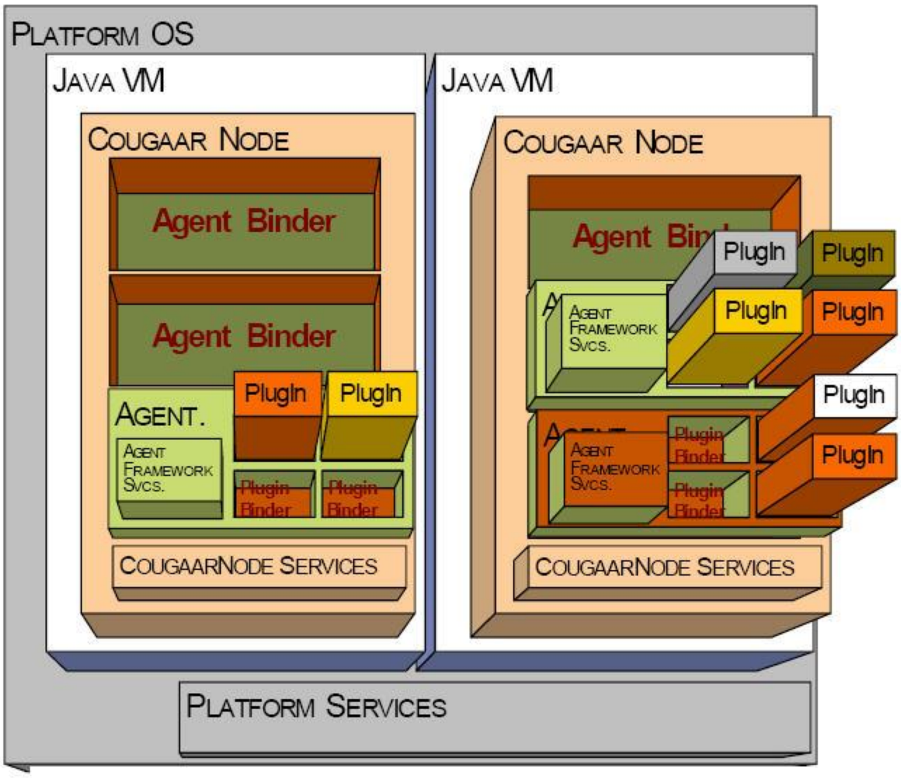

Figure 2. Cougaar Component AI Model [6]

Profile definition: Agent profiles are parameter structures intended to characterize the agent in question. The profile contains all necessary information to allow the scheduled agent to perform his foreseen activity. In simple terms, a "profile" is a plugin. Idle agents are per concept definition "generic", they become dedicated with the assignment of a profile.

Profile construction concept: A profile consists of four parameter sections. Section 1 specifies parameters concerning the inter agent communication, defining the agent credentials (ID and associated agent community members). Section 2 specifies the prime model parameters like learning entity parameters / subject matter parameters etc. Section 3 defines the associated sub profiles like knowledge base profile / knowledge base rule collection / misconception definition / alternative content etc. Section 4 specifies the additional loadable plugin list for foreseen activities.

4 OPUS One as an Adaptive Learning Environment

The term "adaptive" is one of the "trends" in the e-Learning industry today. It's being associated with a range of system characteristics and capabilities of learning. Therefore, it is necessary to qualify the qualities one attributes to a system when using the term. A learning environment is considered adaptive if it is capable of: *monitoring* the activities of its users; *interpreting and evaluating* these on the basis of domain-specific result expectations; *understand user requirements and preferences* analyzing the performed activities and appropriately *representing these in associated models*, and, finally, acting upon available knowledge base / rules or misconception exceptions on its users and the subject matter in question. Being able to dynamically adapt and facilitate the learning process according to the defined learning targets.

4.1 Categories of adaptation in learning environments

Adaptive Interaction; refers to an adaptation process at the User interface level, intended to facilitate or support the user's interactions with the learning platform, never modifying however, in any way the learning "content" itself. OPUS One/OLAT allows to a certain extend to personalize the User Home page in enabling or disabling functions and define the graphical appearance according to user needs.

Adaptive Course Delivery; represents the most common used anthology of adaptation techniques applied in learning environments today. In particular, the idiom is used in reference to adaptations intended to alter a course (or, series of course sequences) for a specific LE. A major factor behind the implementation of adaptive techniques include the compensation of a constantly present human tutor (who is capable of judging the student capacity, approach and target orientation etc., advising the student on a personal base) with a reliable, expert base / rule driven subject matter specific environment, improving subjective evaluations of achievements by the student. Typical examples of adaptations in this category are: dynamic course (re-)structuring; learning path adaptation; adaptive selection of alternative (or sequences of) course materials [3]. The OPUS One eTutor facility, trough the integrated BR-C is supervising the student in "real time" according to the students profile. If the student encounters difficulties in solving course activities (detected by the BR-C and SM-A) or if the student specifically requests tutoring, a "context specific sequence", rule driven advise mechanism will be initiated. Based on the actual subject matter sequence- / session position and the actual valid internal grading value, additional, tailored "problem solving" content / hints will be proposed.

External Content Discovery; refers to the discovery and storage of additional, subject matter related learning material from external sources like other LMS's, websites or specialized repositories. The adaptive component of this process consists in the detection, integration and publication of this additional material among the course community. OPUS One has the capability to detect, using the eTutor surveillance facility, external search activities of LE's or groups, advising the LE or group in question to publish or share this additional material among the learning community and integrate the additional content in a dedicated course repository notifying the course members accordingly.

Adaptive Collaboration; refers to the involvement between multiple students in groups (and therefore, social interaction), proposing collaboration towards common objectives and solutions. This is an important dimension to consider, since modern pedagogy increasingly promote the importance of collaboration activities, cooperative learning, communities of students and social negotiation. Adaptive techniques can be used in this direction to facilitate the communication- / collaboration process and to ensure a good balance between learning communities. OPUS One supports a variety of "tutored collaborative activities" integrated as components into a course or as standalone LMS functions like Wikis, global or dedicated Forums, collaborative writing or collaborative assessment functions, just to mention a few examples. In an exception case, the eTutor facility is able to propose additional collaborative activities to students or groups, if the associated exception rules foresee such type of activities.

4.2 AI Models used in an adaptive learning environment

AI tutoring models, procedures and processes used to realize "learning path adaptation" on intelligent eLearning environments are well-established. OPUS One supports the described models with dedicated, AI based agent communities using fine grained sub models to the main domain model categories.

The domain (or subject matter) model: The domain model task and procedures are focused on adaptive course (content) delivery. The domain-model is usually a representation of the course being executed. In OPUS One, every "subject matter" is controlled by a subject matter main profile, and serviced by an SM-A. Every domain- or subject matter model has access to a knowledge base profile, where adaptive-/ personalization rules and procedures are defined. Every adaptive action is the result of an action request generated by a "knowledge exception", usually originated by the BR-C via the SM-A or by the student, initiating a manual tutoring request. The possible adaptations to the actual subject matter content can be summarized as follows : adapt the learning path with additional, problem- or context oriented content to overcome student specific difficulties; propose repetition of the section(s) in question with an increased tutoring and surveillance level. Every initiated adaptation action is traced and logged by the BR-C, the student profile is updated in real time accordingly.

The learning entity (or Student) model: In OPUS One the term learning entity refers to Student's or Group's working on a "subject matter". The learning entity model is used to reference the characteristics of the learning entity defined in the learning entity profile. The specific approach to modeling and adaptation is accomplished by combining decision parameters from the learning entity profile and the associated subject matter profile.

Group entity model extension: The group entity model extension defines the characteristics of a group of students and their additional opportunities. In OPUS One, Group- and Student models are considered "Learning Entity" models. The main differentiating factors are: a) The different approach in tutoring collaborative activities, b) group models are based on group identification and student members sharing common subject matters, characteristics, global objectives, etc. OPUS One handles Group tutoring and resulting adaptation actions according to a combined Group- and associated learning entity member profile and group decision parameters, driven by Group specific subject matter action rules.

The adaptation model: This model incorporates the adaptive theory of OPUS One. This theory is based on context sensitive , subject matter knowledge base entries, associated with misconception detection and associated adaptation rules, considering progressive grading factors applied to the LE during his learning path. Specifically, the (possibly implicit) adaptation model defines *what* can be adapted, as well as *when* and *how* it is to be adapted (adaptation profile and adaptation knowledge base rules, are part of the knowledge base profile).

5 OPUS One user selectable tutoring functions

eTutor functions can be selected from the user LMS home page tutoring function selection. Selecting this function will activate the eTutor supervisor / administration agent. Profile and role of the requester will be identified and a personalized eTutor function selection windows will be displayed. All functions presented are under control of the Artificial Intelligence subsystem, a separate environment from the LMS. Examples of such functions are: - LMS learning mode change (tutored, walk trough, silent), - On demand eTutor or “human” Tutor request, - Role based, eTutor Administration functions, - Author related functions like “subject matter” functions (create modify, delete profiles, rules, exceptions, predictions), knowledge base extensions, etc.

6 Conclusions and Ongoing Work

The OPUS One / OLAT LMS extensions add support and functionality for an “Intelligent Adaptive Learning Environment” using an AI based e-Tutor subsystem, considering today vital pedagogical aspects (ID - Models, teaching styles etc.). Able to support the student fulfilling his/her educational goals, considering his/her learning style, actual- and progressive knowledge level. The concept is able to support “human tutors” with accurate “student centric” data to better qualify, judge and support the student. The eTutor reliefs the “human tutor” from time consuming, low level tutoring interventions, supporting the student directly with a variety of support tools and hints. The “human tutor” can always, if appropriate, overrule, add or modify proposed adaptation activities by the eTutor. Monitoring, support and tutoring capability of extensive collaborative functions (internal and external) allowing a more fine grained adaptation / personalization process. Using the student adaptation monitoring data and progress results, we create the ability to verify the feasibility of personalization actions applied for the learner.

A major area of ongoing research are “Authoring subsystems”. A needed function to facilitate the definitions and concatenation of profiles, rules and “adaptation functions”. Today this is accomplished using dedicated “technical” utilities. Furthermore Java or Flash based dedicated tutoring code is created using CTAT, linking the code then with the corresponding subject matter- / knowledge base profile, an interim solution that needs to be addressed and modified. Another area of research is the issue of dynamically assembled, reusable content modules as result of an adaptation process, implying the integration of a LOR subsystem. The OPUS One extensions version 1.0 will be released in the 4th Quarter 2009 - as “Open Source” modules, a Demo platform will be available in the late October 2009 timeframe.

References

1. Aleven, V. & Koedinger, K.R.: An effective metacognitive strategy: Learning by doing and explaining with a computer-based Cognitive Tutor. Cognitive Science, 26 (2) (2002)
2. Anderson, J. R.: Rules of the Mind. Mahwah, NJ, Lawrence Erlbaum (1993)
3. Brusilovsky P. : From adaptive hypermedia to the adaptive Web. In: Szwillus G., Ziegler J. (Hrsg.): Mensch & Computer 2003: Interaktion in Bewegung, Stuttgart: B. G. Teubner, pp 21-24 (2003)
4. Cheung B. Kwok L.K.,: Teaching and Learning through Space Online Universal Learning (SOUL) Platform. In: the E-Education Era The 20th IASTED, Innsbruck, (A), (2002)
5. Corbett, A.T. & Anderson, J.R.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User modeling and user-adapted interaction (1995).
6. Cougaar Open Source web site, http://www.cougaar.org
7. CTAT Homepage : http://ctat.pact.cs.cmu.edu
8. Dall’Acqua L.: A Model for an Adaptive e-Learning Environment. Proceedings of WCECS, section “Education and Information Technology, San Francisco, ISBN: 978-988-17012-6-8 (2009)
9. DimDim Homepage: http://www.dimdim.com/
10. Eberts R.E.: Computer-based instruction. In: Helander MG, Landauer TK, Prabhu PV, editors. Handbook of Human-Computer Interaction. Amsterdam: North-Holland; 1997. pp. 825–47
11. Glaser R.: Some implications of previous work on learning and individual differences. In R. M. Gagné (Ed.), *Learning and individual differences* (pp. 1-18). Columbus, OH, Charles Merrill (1967)
12. Helander, M. G., Landauer, T. K., & Prabhu, P. V. (Ed.s): Handbook of Human-Computer Interaction. Amsterdam, The Netherlands: Elsevier Science B. V. (1997)
13. LAMS Homepage: http://www.lamsinternational.com/
14. Mahara Homepage: http://mahara.org/
15. OLAT LMS Homepage : http://www.olat.org
16. OpenMeeting Link: http://code.google.com/p/openmeetings/
17. Paramythis, A., & Loidl-Reisinger, S.: Adaptive Learning Environments and e-Learinng Standards, Electronic Journal of e-Learning, 2 (1), 181-194 (2004)
18. SOUL Homepage: http://soul.hkuspace.hku.hk
19. Thome M.: Managing Applications Comprised of Untrusted Components. Proceedings JavaOne (2002).
20. Zhang, J., Cheung, B. & Hui, L. (2001): An Intelligent Tutoring System: Smart Tutor. In C. Montgomerie & J. Viteli (Eds.), Proceedings of ED-MEDIA 2001 (pp. 2130-2131). Chesapeake, VA (2001)