We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,300

Open access books available

172,000

International authors and editors

190M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Multi-agent Architecture for Retrieving and Tailoring LOs in SCORM Compliant Distance Learning Environments

Pierpaolo Di Bitonto

*Department of Computer Science, University of Bari
Italy*

1. Introduction

The E-Learning research field proposes mechanisms for managing teaching materials that are coherent with the emerging technologies. Attention is focused on the one hand on the definition of standards allowing interoperability and reuse (the success of Learning Objects is an example), and on the other hand on the adoption of methodologies and technologies which ensure a controlled fruition of teaching contents (such as Learning Management Systems).

On both these fronts, many advanced solutions have been proposed, and have proven undoubtedly useful and much appreciated. All the same, they are not yet sufficient to guarantee an “optimal training” experience because they do not take into account some significant variables of the learning process (Di Bitonto & Roselli, 2008). In fact, one of the main problems that has emerged from the e-learning literature is that if different students undertake the same search for a content the LMSs supply the same results without considering the learner profile. Another problem is that these tools often do little more than present the information content, without making any attempt to subject this content to intelligent processing to tailor it to the specific needs of the learner. In fact, they do not allow personalization of the teaching to the true needs and abilities of the individual, nor do they promote adoption of the many different existing teaching strategies (Di Bitonto et al., 2008).

The architecture proposed in the chapter defines a possible solution to both these problems. It supplies methods and techniques for both recommending the Learning Object (LO) that best fits the learner’s profile and tailoring the navigation of the content of recommended LOs on the basis of the learner’s knowledge, cognitive styles and metacognitive abilities. In particular, the chapter focuses on methods and techniques for carrying out personalized teaching actions in SCORM (Sharable Content Object Reference Model) (ADL, 2004) compliant Learning Management Systems.

In order to better understand the research work proposed, the fundamental concepts in the technological, pedagogical and methodological context are described. Section 2 presents pedagogical and technological premises of the research work. From the pedagogical point of view, it examines the literature on the different cognitive styles theories used in the adaptive

process. Regarding the technological issues the section will focus on the description languages (Roselli & Rossano, 2006) and SCORM specifications (in particular the Sequencing and Navigation rules), that are useful to understand the constraints and potentialities of the available technologies. Section 3 illustrates the methodological context, investigating the state of the art of tailoring approaches in e-learning applications and, in particular, the use of rule-based systems in defining personalised learning paths. Section 4 describes the multi-agent architecture for a SCORM compliant distance learning environment designed using TROPOS methodology. Section five describes the adaptation logic used by the adaptation agent in order to tailor the user navigation within a SCORM package. Finally, some conclusions and future works are discussed.

2. Pedagogical and technological basic issues in distance learning environments

To gain a full understanding of the work described, a brief introduction needs to be made to the pedagogical and technological issues faced. In particular, cognitive styles and the Felder model are outlined to help to understand the working logic of the reference scenario. Then the agent-based approach is contextualized to the present work. Finally, a brief introduction to the description languages in e-learning applications is given, as well as an illustration of the essential features of the SCORM standard.

2.1 Cognitive styles

Each learner usually shows a tendency or a preference for one or more modalities in the way that teaching actions are received and elaborated. For instance, some students prefer learning by doing, others, learning by images or by example. This implies that different learners approach learning tasks in different ways. Kefee (1985) defines cognitive styles as “characteristic cognitive, affective, and psychological behaviours that serve as relatively stable indicators of how learners perceive, interact with, and respond to the learning environment”. Alonso (1993) defines learning styles as “personal manners to perceive and process information, and how they interact and respond to educational stimuli”.

In the last thirty years many researchers (mainly pedagogues and psychologists) have studied how to characterize cognitive styles and how to exploit the different styles during the educational process. A brief description of the most commonly used models in e-learning literature is presented.

One of the popular models in e-learning is the Felder and Silverman model (Felder & Silverman, 1988) that defines the appropriate teaching strategy for each user's cognitive style. In order to distinguish the different cognitive styles, it considers four dimensions: “source of information”, that can be used to define perceptive or intuitive cognitive styles, “information code”, used to distinguish between visual or verbal styles, “information processing”, to differentiate active or reflective styles, and “summarizing information”, to define sequential or global styles.

For each of them, Felder states the typical student's behaviour and suggests the best way to teach each of them. Perceptive students can easily learn information received from the environment through their five senses, whereas intuitive students prefer information generated by associations drawn from memory, by reflection or by interpretation. Therefore the perceptive student needs detailed examples of concepts and procedures or applications

of theoretical concepts, whereas the intuitive student prefers theories or interpretations that associate the facts.

Visual students prefer a graphic representation of the didactic material: they are better able to acquire information through images, even complex ones, schemes, graphs, and so on. They have some difficulty in memorising information provided during an oral presentation. Verbal students learn best using texts or oral explanations, and have no difficulty in remembering what they hear. Therefore, learning resources with images are the best suited for visual students, whereas text, video or audio resources are the best ones for verbal students.

Active students memorise the information by applying it in practical situations, whereas reflective students are theorists, modellers, and mathematicians. They prefer to define problems and find possible formal solutions. Therefore, an reflective student needs theory or interpretations that associate the facts, whereas a active student needs simulation or teaching resources built using cooperative learning or learning by doing approaches.

Sequential students learn information by using an ordered approach and building up their knowledge step by step. Global students need to know how the information can be contextualised in the whole domain world. Therefore, a sequential student needs to study concepts step by step, noting their inter-relations, whereas a global student needs to study first of all the general scenario, and then to examine the specific content within it.

Another model used in the e-learning literature is the VARK defined by Fleming and Mills in 1992 (Fleming, 2001). The approach used is very similar to the one in the Felder model. It classifies the learner in four different categories (Visual, Aural/auditory, Read/Write, and Kinaesthetic) and describes the main medium used to learn. According to the VARK model visual students usually prefer images (i.e. tables, graphics, flow chart); aural/auditory students learn by listening (i.e. conference, lessons) or discussing; read/write students tend to learn best by reading, taking notes, and kinaesthetic students prefer concrete learning experiences (real or virtual) like simulations or activities that engage them.

Gardner's multiple intelligences theory offers a different point of view in the field of cognitive styles. He identifies different key aspects through which students think and learn; in (Gardner, 1983), he defines seven kinds of "intelligences" which guide and influence the learning process, namely linguistic, logical-mathematical, spatial, bodily-kinaesthetic, interpersonal, intrapersonal, and musical.

Students with a linguistic intelligence display a particular facility for words and languages. Usually they are good at reading and writing, they are able to memorize texts and dates and they tend to learn best by reading, taking notes, listening to lectures, and discussing; usually they are able to manipulate language syntax and structure. Students with a logical-mathematical intelligence are usually good at logical reasoning, abstraction and mathematical computation, so they are good scientists (i.e. mathematicians, computer programmers and so on). A spatial intelligence makes students very good at visualizing and mentally manipulating objects. Spatial students have a strong visual memory and they are often artistically inclined. Students with a bodily-kinaesthetic intelligence learn better if they are physically involved in the learning experience, so they prefer teaching strategies involving learning by doing and simulations. In general, they are good at building and making things. Students with an interpersonal intelligence are extroverts, sensitive, and good at cooperating with their colleagues, discussion and debate, so they learn better if they work in a group. On the contrary, introspective students are typically introverts and prefer

to work alone. They learn best when allowed to concentrate on the subject by themselves. Finally, students with a musical intelligence are able to recognize rhythms and tones. They prefer to learn by hearing and reading.

2.2 Description language standards

The International Organization for Standardization (ISO) defines a standard as *a documented agreement containing technical specifications or other precise criteria to be used consistently as rules, guidelines, or definitions of characteristics, to ensure that materials, products, processes and services are fit for their purpose* (<http://www.iso.org>). In the context of e-learning technologies, standards are generally developed to be used in system design and implementation for the purposes of ensuring interoperability, portability and reusability. The adoption of specific standards for metadata would encourage exchanges of material among the different actors in the learning process (McGreal & Roberts, 2001).

Already by the end of the '90s, various committees and organizations operating in the e-learning field had proposed a number of sets of specifics in the hope that they would be imposed as de facto standards. The most common one in use at the moment is the Dublin Core Metadata Initiative (DCMI). This began in 1995 with an invitational workshop in Dublin, Ohio, that brought together librarians, digital library researchers, content providers, and text mark-up experts aiming to improve discovery standards for information resources. The original Dublin Core emerged as a small set of descriptors, Dublin Core Metadata Element Set that quickly attracted global interest from a wide variety of information providers in the arts, sciences, education, business, and government sectors. Thus, the Dublin Core Metadata Element Set (DCMES) is a standard for cross-domain information resource description. However, the DCMI was the first initiative from which the need to introduce descriptors for online resources emerged, so it acted as the starting point for other organizations (ARIADNE, IMS, IEEE, etc.) that are currently focusing their attention on the definition of LO metadata sets.

In 2002 the LOM (Learning Object Metadata) standard was defined by IEEE. Unlike the Dublin Core, it is a set of specifications that serves to specifically describe learning objects and their components. It includes more than 80 descriptive elements subdivided into the following 9 categories: general (includes all the general information that describes the resource as a whole); lifecycle (groups the descriptors of any subsequent versions of the LO); meta-metadata (include the information on the metadata themselves); technical (indicates the technical requisites needed to run the LO and its technical characteristics); educational (contains the pedagogical and educational information about the LO); rights: (indicates the intellectual property rights and any conditions of use); relation: (describes any relations with other Los), annotation (allows insertion of comments) and classification (makes it possible to classify the LO). The main novelty of the LOM is the educational category that contains elements like: Interactivity type, Learning resource type, Semantic density, Typical learning time, that supply indications on how the LO can be inserted in the teaching program. Another strong point of the LOM is that it can be used to describe many closed vocabulary items. These facilitate automatic processing of the information.

As stated in (Roselli & Rossano, 2006) as regards the LOM definition, the e-learning community has concentrated hard on the task of defining standards that can guarantee the interoperability and reusability (from the informatics standpoint) of materials, but has tended rather to neglect their reusability (from the pedagogical standpoint) and the

personalization of learning paths. In the LOM the quality, as well as the quantity, of the pedagogical information is somewhat lacking and it is not sufficiently detailed to provide a full description of the educational context where the teaching resource to which it refers should preferably be used. To overcome this problem, in 2003 the IMS Global Learning Consortium developed the IMS Learning Design. It consists of a set of specifications aiming to provide a detailed description of a learning scenario by means of a special language denominated Educational Modelling Language (EML). The specifics can describe a wide range of pedagogical models or learning approaches, including work groups, and collaborative learning. The ultimate aim is to define new learning models to describe how the actors in a given scenario interact among themselves, using the teaching resources (in terms of both teaching materials and support services), and how the whole procedure can be coordinated and channelled to create a learning path.

2.3 SCORM

The Sharable Content Object Reference Model (SCORM) provides a set of technical specifications serving to create content that will run on any conforming LMS. SCORM is not a standard but it is one of the best and most recent examples of the application and integration of current e-learning standards. It was defined in early 2000 in the Advanced Distributed Learning (ADL) initiative of the US Federal Government. The ultimate aim of the ADL initiative was to supply a comprehensive set of guidelines that enable interoperability, accessibility and reusability of e-learning content within the Department of Defence. The US military, therefore, can all use, exchange, manage, track and reuse all of the learning content and data no matter what its source or application.

The ongoing work of the ADL has produced different versions of the SCORM specifications. The latest was released in 2004. It is organized in the form of four books: Overview, Content Aggregation Model (CAM), Run Time Environment (RTE), Sequencing and Navigation (SN).

As stated in (ADL, 2004) the Content Aggregation Model (CAM) book describes *“the types of content objects used in a content aggregation, how to package those content objects to provide for successful exchange from system to system, how to describe those content objects using metadata to enable search and discovery, and how to define the sequencing rules for the content objects to complete the design of the learning experience”*. In particular the content aggregation model defines how to create the manifest file. It is an XML document that contains a structured inventory of the content of a package and sequencing information that can be placed in order to define sets of sequencing strategies for the learner activities.

Run-Time Environment (RTE) book describes *“the LMS requirements that allow interoperability of content across different LMSs (i.e., a standardized content launch process, standardized methods of effecting communication between content and LMSs, and standardized data model elements used for passing information about the learner’s interactions with the content)”*. The SCORM RTE defines the set of data and functions that can be transferred to and from the LMS and enables interoperability between SCOs and LMSs. The Sequencing and Navigation (SN) book defines *“a method for representing the intended behaviour of a learning experience such that any SCORM-compliant LMS will sequence learning activities in a consistent way. It also defines the required behaviors and functionality that SCORM-compliant LMSs must implement to process sequencing information at run-time”*. In other words, the SN book describes how to define personalized learning paths according to the tracings of student activities. The proposed

solution, to tailor the LOs in SCORM compliant distance learning environments, is based on the SN specifications that are described in detail in section 5.

3. Adaptivity in distance learning environments

Adaptivity in distance learning environments is one of the main problems faced by researchers (Ruiz et al., 2008; Chun-Hsiung et al., 2008; Popescu, 2008). Their primary goal is to meet the individual user's learning needs. In the literature, two different solutions are proposed: to tailor the content of a single learning resource by modifying the navigational path within the LO, that can be named the *intra-adaptivity approach*, and to combine different learning resources in order to build up a personalized learning path, that can be named the *inter-adaptivity approach*.

The intra-adaptivity approach allows the development of very complex learning resources for simple LMS. The single LOs should implement different learning paths for different cognitive styles, background knowledge, goals, etc. On the contrary, in the inter-adaptivity approach the learning resources are very simple and the LMS should be able to build personalized learning paths by assembling different LOs using the best teaching strategies according to the learner profile. In both approaches (intra-adaptive and inter-adaptive) the key points are the description of the student and of the LOs, as well as the logic used to associate the learner's profile to the teaching strategy. The trend in adaptive system research is to face this issue using the cognitive styles theories. The state of the art analysis reported below illustrates different applications of the two approaches in e-learning environments and the use of different cognitive styles to personalize learning paths in distance learning environments.

3.1 Adaptivity approaches

3.1.1 Intra-adaptivity approach

The Intra-adaptivity approach is supported by many researchers (Arapi et al, 2003; Brusilovsky, 2004; Friesen, 2005) and by the SCORM standard (starting from version 1.3).

A lot of works on Learning Objects are contributing to identify patterns for developing instructional contents characterized by an adaptive, generic, portable nature and sufficiently scalable to improve their potential for reusability (Alvino et al, 2008; Bodendorf et al, 2005; Earle, 2002). In addition, a wide range of virtual learning environments has been proposed to support the use of learning objects, their properties and characteristics (Fischer, 2001; Pollyana & Silveira, 2006; Brady et al, 2008).

When using an Intra-adaptivity approach, highly complex LOs need to be built, owing to the numerous variables that need to be considered in the adaptivity process, as stated in (Ruiz et al. 2008). In (Monacis et al, 2009) the authors built an adaptive learning object using the standard SCORM, which dynamically related different learning contents to students' cognitive styles. In order to define the personalized learning paths for each student, the authors defined a set of navigation rules and a questionnaire to acquire the cognitive style.

In Pollyana and Silveira (Pollyana & Silveira, 2006) a six-tiered architecture for adaptive learning objects is presented, that organizes the information in a macrostructure in order to standardize learners' profiles, learning objectives, structure of courses and so on. The

“Course tier” allows learning objectives to be defined so that for each course the expected skills and abilities are stated; the “Reusable LOs tier” allows the characteristics related to each learning objective to be defined; the “Apprentice Model Tier” keeps track of students' historical information; the “Learning Styles Tier” keeps track of the student's learning style in all its dimensions; the “Presentation tier” is responsible for the dynamic generation of a suitable display of learning objects according to the learner's profile, taking into account his/her inferred learning styles for a certain learning context. The Presentation tier is composed of two parts: a “Learning Objects Model”, which describes all the learning objects that are to be used in the learning context, and an “Apprentice Profile”, formed by the learner's historical profile and by learning styles that are detected according to the specific learning context.

Such an approach, although extremely interesting, is difficult to apply on a large scale because it has many constraints and is expensive in terms of the creation and description of learning contents. Moreover, the countless descriptions that have to be considered make the process of seeking the most suitable LO for the user too slow.

3.1.2 Intra-adaptivity approach

The inter-adaptivity approach allows us to think of the LO as “any digital entity which can be used, reused or referenced during a technology-mediated learning process”, according to Wiley's definition (Wiley, 2000). Moreover, this makes it possible to create simpler LOs that can then be dynamically assembled according to the student's learning needs. The problem arising in this case is how best to combine these LOs.

Different methods and approaches have been proposed for determining the learning paths that can best support the learning experience of navigation through the set of LOs. It is possible to distinguish three kinds of techniques: rule-based, ontology based and planning based.

The rule-based techniques use an explicit knowledge representation to describe characteristics, constraints and relations in the domain knowledge. They consider, for each LO, a set of skills that describes it. In this way it is possible to add to the LMS some adaptation components that use the knowledge about the LO skills, together with the student description (learning goal, prior knowledge and mental aptitude), to produce sequences that fit the user's requirements and characteristics, within the limits of the available learning objects. Working at the skills level makes it easier to consider the relations among LOs, and this makes reuse of the learning objects easier, as well as their application in goal-directed reasoning processes. Primitive forms of adapting systems were presented in (Seal & Przasnyski, 2001), where the authors propose a system that can collect students' feedback during interaction with the course in order to suggest to the teacher how to adapt the course itself to the classroom needs. No adaptation was conducted automatically. A more complete solution is presented by (Trigano & Pacurar-Giacomini, 2004). They propose CEPIAH, a system that can help teachers to develop pedagogical web sites and on-line courses. Using CEPIAH, the teacher (starting from two questionnaires) can automatically generate educational Web site structures, add pedagogical contents, and finally visualize, manage and participate in the courses. The questionnaires are related to pedagogical issues (i.e. teaching scenario, pedagogical approach ...) and interaction issues (i.e. colours, shape of menus, buttons...). In short, the automatically generated course structure is based on teaching scenarios which integrate the features of different pedagogical theories. The

system's inference mechanism is based on rules that specify how the pedagogical models are assembled.

Another example of a rule-based system for automatic content creation is the ECSAIWeb presented in (Sanrach & Grandbastien, 2000). The system describes each LO as a learning unit composed of "content", what the student has to learn, any "pre-condition", what the student has to know before starting to study the LO, and "post-condition", what the student has to know after studying the LO. Using combination rules that fit together LOs on the base of content, pre-condition, post-condition and student's profile, the system builds the personalized learning path.

In general, using rule-based systems yields interesting results in small and static domains where it is possible to build a set of rules to manage the entire possible situation that can develop. In more complex cases, it could be necessary to consider different domain knowledge. In these cases it is necessary to use a more flexible knowledge representation: the ontology.

The ontology-based techniques, like the rule-based techniques, use an explicit knowledge representation to describe characteristics, constraints and relation of the domain knowledge. Unlike rule-based, ontology-based techniques allow the domain to be described in great detail, so it is possible not only to define rules to associate learning resources to the student, but also to browse the knowledge domain.

(Ronchetti & Saini, 2003) present an e-learning environment able to manage a course on data base, organized as a set of LOs. Each LO is associated by metadata to its topic, and each topic is organized by ontology where a set of relations is defined (membership, subclass, pre-requisite, conceptual similarity ...). The defined ontology is used not only to assembly the best learning path for the student, but can also be browsed in order to allow the student to find topics which are related to a given LO and then find other similar topics whose study is correlated.

(Benayache & Abel 2005) present an e-learning environment for automatic course generation. The knowledge about resources, domain and users is organized in two ontologies: the first, "application ontology", describes the specific didactic domain; the second, "domain ontology", describes teaching resources like teachers, competences, didactic material and so on. Using pedagogical relations, like pre-condition and post-condition, the system defines topic maps that contextualize each LO. The ontologies are used by teachers and learners to find the learning resources.

The planning based techniques are able to construct different learning paths considering the domain constraints and they are able to choose the best learning path according to the user's needs. Planning based techniques do not exclude rule-based or ontology-based approaches to describe the domain knowledge. For instance, (Peachy & McCalla,) use planning techniques to summarize the educational resources that achieve the learning goals and rule-based techniques to define the domain constraints. Likewise, in (Karampiperis and Sampson) use ontologies and metadata in order to calculate the best learning path through the didactic resources automatically generated using a planning system.

In the WLog system (Baldoni et al, 2004a) the learning objects are represented as actions, each of which has a set of prerequisites (skills needed to be able to use the learning object) and a set of effects (skills supplied). The skills can be linked by different relationships, causal or requirement, for instance, (Baldoni et al, 2004b). Using planning algorithms, the adaptation components can infer the best way to combine LOs.

Other adaptive and dynamic courseware generation systems based on planning approach are Tangram (Jovanovic et al 2005), OntAware (Holohan, et al, 2005), and Paser (Kontopoulos et al, 2008). These systems provide guidance and direction towards the most appropriate learning path that the student could follow each time. The principal limits of the approaches based on the planning algorithm are the computational costs, that are usually very high (Kontopoulos et al, 2008; Morales et al, 2008) and the inability to decompose didactic resources into smaller units which can be reassembled in different learning paths. A primitive solution to this problem is presented in (Jovanovic et al 2005).

3.2 Cognitive styles in adaptive educational systems

Cognitive styles are used in intra-adaptivity and inter-adaptivity approaches, as shown in (Pollyana & Ismar, 2006). As reported in (Popescu, 2008), one of the first adaptive educational systems to consider cognitive styles is shown in (Carver et al, 1999) where an adaptive hypermedia interface is developed. It provides dynamic tailoring of the presentation of course material based on the individual student's cognitive style. In particular, it is based on three cognitive dimensions of the Felder-Silverman model (Felder & Silverman, 1988): perception, input, understanding. Each of them is evaluated by applying Felder-Soloman's questionnaire. The adaptation is based on suitability for each particular learning style. Another example of an adaptive hypermedia system that uses the Felder-Solomon Learning Style Questionnaire to measure the cognitive style of students is described in (Bajraktarevic et al 2003). The system uses global and sequential learning styles to adapt the learning path, to cater for individual learner preferences. The work shows the importance of cognitive styles adaptation in e-learning settings, comparing the "matched session" (global student interacts with global approach) with the "mismatched session" (global student interacts with sequential approach and vice versa). In (Monacis et al, 2009) the Italian Cognitive Styles Questionnaire defined by De Beni, Moè, and Cornoldi (De Beni et al, 2003) it is used to decide how to tailor the learning content to the students' profiles.

Other models of cognitive styles used in literature are the VARK (Auditory, Visual, Read/Write, Kinesthetic) proposed by Kolb in 1995 and Gardner's theory of multiple intelligences (Gardner, 1993). The VARK model is used in Arthur's system (Gilbert & Han, 1999) and in the SACS (Style-based Ant Colony System) (Wang et al, 2008); Gardner's theory is used in EDUCE (Kelly & Tangney, 2006). Other systems consider different cognitive styles at the same time, as reported in (Pollyana & Ismar, 2006) or are independent of any particular learning style model, like the AHA! system (Stash, 2007).

4. Multi-agent Architecture for distance learning environments

The technologies, methods and techniques presented in the previous sections are able to furnish a personalized educational process in a distance education environment. The main problem is the low level of integration of such solutions, in particular between the inter-adaptive and intra-adaptive type. This section presents a multi-agent architecture proposal that integrates both approaches. This idea was presented for the first time in (Di Bitonto & Roselli, 2008), where a rule-based technique to adapt SCORM compliant learning objects was proposed.

The basic idea (showed in Figure 2) is that the student interacts with a SCORM compliant learning environment that suggests, by means of a Search Engine, the resources best suited

to the learner's characteristics (inter-adaptive approach). Each recommended resource is then adapted to the student's cognitive styles by means of an Adaptation Engine that modifies the navigation within the resources (intra-adaptive approach) using the sequencing and navigation rules of the manifest file. A first implemented prototype is presented in (Di Bitonto, 2009) where a multi-agent system was presented to support the learning process in an open community.

Fig. 1. Conceptual framework of adaptive learning environments

Figure 1, shows the conceptual framework of an e-learning environment. When the student searches for teaching material, the Search Engine selects the LOs best suited to the student's needs using the LOM metadata collected in the LOR (Learning Object Repository), and the student profile (inter-adaptivity). For intra-adaptivity, the adaptation engine defines how to modify the file manifest according to the student's cognitive styles, other data tracked and memorized in the student profile and the teaching strategies implemented in the Knowledge Base (KB). The problem at this point is that on account of its internal features, the LO may be more or less adaptable to the student's needs. Thus, the adapting agent must calculate the level of changes possible and select the most adaptable option. After doing this, it transfers the file manifest to the LMS so that it can be presented to the student.

This section presents the multi-agent architecture design used to develop the conceptual framework. In particular, it highlights how the adaptive agent builds a personalized learning path within a LO.

4.1 The TROPOS methodology

According to Russell and Norvig (Russell & Norvig, 2003) an agent is any software entity that can perceive its surrounding environment by means of sensors and modify it using actuators. One of the most common techniques used to define an agent is the PEAS (Performance Environment Actuator Sensor) description. It is important to define the agent's performance so as to provide it with heuristics on which to choose the strategy to be actuated from among those available. The environment supplies the context of use, characterizing the problem that the agent is designed to solve. The actuators are the tools that allow the agent to carry out actions on the environment. Finally, the sensors are the sources from which the agent acquires the information on the environment that it needs to work on.

Agents are structured in two distinct parts: the architecture and the agent program. The architecture makes the perceptions available to the program, carries out the program and passes on the actions it has chosen to the actuators, as they are generated. Instead, the agent program relates the perceptions to the actions. The agent has goals to satisfy, it is autonomous and it collaborates with the other agents to satisfy the common goals. A group of agents that communicate with each other is defined as a Multi-Agent System (MAS).

The complexity of MAS has required an ad hoc methodology to be defined for the MAS analysis and design: GAIA (Wooldridge et al 2000), MASE (Wood & DeLoach, 2001)) and TROPOS (Giorgini et al 2006) are some examples reported in literature. The methodology used in this work is TROPOS. It covers the entire software development process, from requirement analysis to system implementation. It consists of five phases: early requirements, late requirements, architectural design, detailed design, and implementation.

In the Early Requirements phase the environment and its organization are studied. The outputs of this phase (goal diagram, and actor diagram) are organizational models that include the relevant actors and the goals that need to be satisfied for each of them, the plans to be carried out and the resources necessary. In the Late Requirements phase a new actor is included, named the "system-to-be", that represents the software architecture to be implemented. Its goals are individuated, as well as the dependencies on the other actors in the environment. The output of this phase is the extended actor diagram.

In the Architectural Design phase the overall structure of the system is defined as a series of subsystems that are interlinked by control data and flows that are modeled to cater for dependencies. Instead, the agents, their dependencies and interactions are detailed in the Detailed Design phase. Finally, what has been defined in the previous phases is coded in the final "Implementation" phase.

4.2 Early and late requirements

In the early and late requirements phase the MAS actors are defined. Starting from the conceptual framework of an adaptive learning system the following actors were defined: student, Teaching Manager, Student Assistant, Learning Path Builder, Content Manager, and User Profiler.

The students are the main users of the learning environment. They interact with the learning environment through the e-learning platform in order to communicate with other users, to search the LOs and to navigate into personalized learning paths.

The Teaching Manager (TM) manages the monitoring and tracking of students' navigation and the administrative activities. The Student Assistant (SA) evaluates the student's needs.

The Learning Path Builder (LPB), represents the adaptation engine, it decides if the LO must be modified and builds a personalized learning path according to the student's cognitive styles. It also manages information on the cognitive styles in the student model. The LPB is the main actor, in the sense that it coordinates the main communication between the other actors. The Content Manager (CM), represents the search engine of the adaptive learning environment; it seeks the best suited learning resources to the learner. Finally, the User Profiler (UP) manages the student information that serves to personalize the teaching strategies. The interactions among the actors are depicted in Figure 2

Fig. 2. Interactions among the actors

Some examples of interaction among actors are presented. Let us suppose that a student logs on to the system for the first time. The UP actor checks on the student information and points out the lack of knowledge about the student's cognitive styles. So it alerts the LPB actor to produce and submit to the student (through the SA actor) a questionnaire that helps to make for the cognitive styles evaluation. The questionnaire is evaluated by the SA that indicates the student's cognitive style to the LPB.

Let us envisage another interaction scenario: the adaptation of the learning path within the LO according to the student's cognitive style. Let us suppose that the LPB knows the student's cognitive styles. The SA actor, interacting with the CM, seeks for the LO that will be submitted to the student according to her/his cognitive styles. When the LO has been selected, it is passed to the LPB, that evaluates the student's cognitive style and decides how to personalize the learning path. If the LO is modified the file manifest will change, and the new SCORM content package is then passed to the SA for use by the student.

4.3 Architectural and detailed design

In the architectural and detailed design phase, the LPB is decomposed into sub-actors, each of which has specific sub-capabilities. The sub-actors are: the Cognitive Styles Manager (CSM), the Learning Path Manager (LPM), the Selection Path Manager (SPM) that decides how to change the learning paths.

Fig. 3. LPB decomposition

After the LPB decomposition has been defined the agents that must be implemented in the MAS and their capabilities are defined (according to the actor's capabilities).

The agents are the CM, CSM, LPM, SPM, UP, and SA. The CM (Content Manager) agent is able to search, insert, modify, and delete the LO; the LPM (Learning path manager) agent that is able to extract, modify and re-zip the SCORM package; the SPM (selection path manager) agent that is able to decide how to change the learning paths; the CSM (cognitive styles manager) agent that evaluates and updates the student's cognitive style; the UP (user profiler) agent that authenticates the student, manages and updates her/his profile; the SA (Student assistant) agent, that alerts the system if new LOs are added or if a user profile is changed. It is also able to check the learner's knowledge and seek for knowledge gaps.

4.4. Implementation

Some of the agent capabilities, defined in the architectural and detailed design phases, are supported by the e-learning platform [rif]. The UP agent, for instance, manages the system registration and authentication procedures. Since those capabilities are partially supported by e-learning platform, the UP agent manages the data exchanged between E-learning platform and the MAS system. For instance in the log-in process the UP agent checks the user-id and password in the E-learning platform DB. Figure 4(a) shows the login page. Finally, the UP agent evaluates the user's cognitive style by implementing the Felder/Soloman questionnaire. The resulting cognitive styles are memorized in the e-learning platform DB.

Fig. 4. System screenshot

The CM agent is implemented using the repository functions offered by e-learning platform. Figure 4(b) shows the page for the selection of LOs that are memorized in the repository. If new LOs are added in the repository they will automatically appear in the list of available LOs.

Unlike the UP and CM agents, the CSM, LPM, SPM agents are not supported by E-learning platform services. They work in the background and are not visible to the student. The CSM agent examines the student's activity tracking to infer any change in her/his cognitive styles. If some change has occurred, it modifies the user's profile.

The LPM and SPM agents work together to modify the learning path within the LO. First of all the LPM extracts the manifest file from the LO zip package and parses the manifest file starting from the root node. For each node, if the name of the tag is `<imsss:sequencing>`, it selects the set of tags that can be adapted to the learner's characteristics and passes them to the SPM. The SPM agent implements a rule-based technique that, according to the student's cognitive style, determines how to modify the `<imsss:sequencing>` tags and attributes. The changes are passed to the LPM that modifies the manifest, creates a new content package and passes the LO to the e-learning platform for the use by the student.

5. Adaptation logic

In order to understand how an adaptive learning environment based on the multi-agent architecture presented works, it is important to briefly illustrate the SPM adaptation logic. The basic element is the IMS SS Sequencing Definition Model. It defines a set of elements that can be used to describe and affect various sequencing behaviours within the LO. There are four main sections for the technique presented here: *Sequencing Control Modes*, *Constrain Choice Controls*, *Sequencing Rule Description*, and *Limit Condition*.

The six elements in the Sequencing Control Modes section are: *Sequencing Control Choice*, *Sequencing Control Choice Exit*, *Sequencing Control Flow*, *Sequencing Control Forward Only*, *Use Current Attempt Objective Information*, and *Use Current Attempt Progress Information*. They have a Boolean value (true or false) and affect the student's navigation.

The *Sequencing Control Choice*, (if it is true) indicates that the learner is free to choose any learning activity in any order without restriction; The *Sequencing Control Choice Exit* element (if it is true), indicates that a learning activity can be terminated in order to pass on to another activity; the *Sequencing Control Flow* (if it is true) indicates that the system-directed sequencing for the user's activities is supported, *Sequencing Control Forward Only* (if it is true) indicates that the student should not go back.

There are two elements in the *Constrain Choice Controls* section: *Constrain Choice* and *Prevent Activation*. They define constraints on the navigational process according to the logical dependencies specified in the LO. In particular, *Constrain Choice* (if it is true) indicates that only activities that are logically “next” and “previous” in the activity tree may be successfully connected. *Prevent Activation* (if it is true) indicates that the student can’t begin another lesson if the current lesson is not yet finished. The purpose of this element is to prevent a learner from “jumping” too deeply into the content without first gaining some prerequisite knowledge.

The *Sequencing Rule Description* section furnishes a set of condition-action rules to modify the learning paths. The conditions are evaluated using the tracking information associated with the activity. This section is not yet operative in the technique.

The two elements in the Limit Condition section are: *Attempt Limits* and *Attempt Absolute Duration*. They define the minimum temporal limit for an activity. The *Attempt Limit* element contains a non-negative integer value that specifies the maximum number of attempts that may be made of the associated activity; the *Attempt Absolute Duration Limit* element contains a value that specifies the maximum time that a learner is permitted to spend on a single attempt.

Let us suppose that the SPM agent receives the <imsss:sequencing> elements for the purpose of modifying the LO navigation according to the student's cognitive styles. For each cognitive style and <imsss:sequencing> element a production rule changes the manifest settings. Table 1 shows some of the production rules defined, when components of the LO are text and images. In the case of simulation, collaborative or practical activities the production rules defined will be different.

	Information summarizing		Information processing	
	sequential	global	reflective	active
S.C. Choice	not relevant	true	true	false
S.C.Choice Exit	true	true	true	false
S.C.Flow	true	true	true	false
S.C.Forward Only	false	false	false	true
Att. Abs. Dur. Limit	not relevant	not relevant	increased	not relevant
Attempt Limit	not relevant	not relevant	increased	increased
Prevent Activation	false	true	false	true
Constrain Choice	true	true	false	true

Table 1. Summary of production rules

The sequential cognitive style implies that the student tends to study step by step without jumping from one concept to another. So, for the sequential student the *Sequencing Control Choice* is not relevant because she/he generally prefers the sequential navigation that is possible in any case (*Sequencing Control Choice* set to true or false). *Sequencing Control Choice Exit* and *Sequencing Control Flow* are set to true and *Prevent Activation* is set to false because the student does not tend to jump from one lesson to another; *Sequencing Control Forward Only* is set to false. *Attempt Limits* and *Attempt Absolute Duration* are not relevant to the sequential cognitive style. *Constrain Choice* is set to true because the sequential student can have some difficulties in studying arguments that are not strictly connected.

Global students need to know how the information can be contextualised in the whole domain world. Therefore, they need to study first of all the general scenario, and then to examine the specific content within it. So, for the global student the *Sequencing Control Choice*, *Sequencing Control Choice Exit*, *Sequencing Control Flow* are set to true in order to give her/him the possibility to explore the whole domain world. Too much freedom in the hyperspace can confuse the global student. So, *Constrain Choice* and *Prevent Activation* are set to true. *Attempt Limits* and *Attempt Absolute Duration* are not relevant to the global cognitive style.

The Reflective student tends to examine the concepts very carefully before answering; moreover she/he proceeds with care using a step by step approach and reflects deeply before answering. Usually the reflective student is very responsible. For those reasons the *Sequencing Control Choice*, *Sequencing Control Choice Exit*, and *Sequencing Control Flow* are set to true, whereas *Sequencing Control Forward Only*, *Constrain Choice*, *Prevent Activation* are set to false in order to give her/him freedom in the navigation. *Attempt Limits* and *Attempt Absolute Duration* are incremented in order to give the student more time to think.

Finally, active students prefer simulation and practical activities. Sometimes they are superficial, so they should be guided during their navigation. So, *Sequencing Control Choice*, *Sequencing Control Choice Exit*, and *Sequencing Control Flow* are set to false, while *Constrain Choice*, *Sequencing Control Forward Only* and *Prevent Activation* are set to true. The *Attempt Limit* is increased.

Each student can have a different cognitive style and the different cognitive style can activate different changes in the manifest. Conflicts among the different rules can arise. In order to solve this problem, salience values (from 1 to 3) are defined according to empirical evidence, that serve to choose the best rule to apply. In (Di Bitonto, 2009) other kinds of production rules are presented to select the teaching strategies. The other two dimensions of the cognitive styles “source of information” and “information code” are not yet considered because they are not related to navigation of the LO. To consider them, it is necessary to use the *Sequencing Rule Description* section of the manifest. This will be done in future works.

5.1 Example of LO adaptation

Let us suppose that a new user is registered in the system. When she/he makes the first login, she/he chooses a LO. The system recognizes the absence of a cognitive style evaluation and proposes the questionnaire. Let us now suppose that the student is active and global. The saliencies of the rules and the result of conflict resolution are shown in Table 2.

	global	salience	active	salience	result
S.C. Choice	true	1	false	2	false
S.C.Choice Exit	true	3	false	2	true
S.C.Flow	true	2	false	3	false
S.C.Forward Only	false	2	true	3	true
Att. Abs. Dur. Limit	n.r		n.r.		n.r.
Attempt Limit	n.r		incr.	2	incr.
Prevent Activation	true		true		true
Constrain Choice	true	2	true	1	true

Table 2. Conflict resolution among production rules

In Figure 5 some results of the adaptation process are shown. Figure 5 (a) shows the LO before the modification: the activity tree and “previous” button are on the left and the top of the page, respectively. After the adaptation those tools are removed.

Fig. 5. An example of LO adaptation

6. Conclusions and future works

One of the main advantages of distance education is that it offers the possibility of personalizing the learning path. The challenge for researchers is to define methods, techniques and tools to support intelligent searches and tailoring of the retrieved contents. Nevertheless, the main problem of the technological solutions defined so far is that they are often ad hoc solutions.

The present chapter faces both these problems, presenting a multiagent architecture that is able to search for LOs and personalize educational processes in a SCORM compliant distance learning environment.

First of all, unlike other systems reported in literature, where the solutions use only one of the two approaches (intra-adaptive and inter-adaptive), the presented framework uses a combined approach. Thus, it merges searching techniques, developed in an intra-adaptive area, with learning path personalization techniques developed in an inter-adaptive area.

Moreover, the multi-agent architecture was defined starting from the conceptual framework of an adaptive learning environment. In order to retrieve and tailor the resource best suited to the user, it adopts the Felder and Silverman cognitive styles theory. The difference from

other systems presented in literature is that the proposed MAS modifies the SCORM package in accordance with users' cognitive styles using sequencing and navigation rules. The use of the SCORM standard makes the adaptive technique very general and adaptable in any context. The prerequisite is that the LO should have been designed to be adapted. One of the probable weak points of the adaptation technique is that it may develop problems when it is applied in a context where many students are using the system at the same time. Simultaneous access can slow the system performance, reducing the efficiency of the presented solution. In order to measure the performance of the system in such contexts, an experiment is being designed to cope with a large number of students. Future work will involve the completion of the adaptation rules in order to integrate the Sequencing rule description section of the imsss Definition Model.

7. References

- ADL (2004) Sharable Content Object Reference Model (SCORM) 2004 3rd Edition Documentation Suite
<http://www.adlnet.gov/downloads/DownloadPage.aspx?ID=237>
- Alonso, C. M. (1993). Educational Technology and Learning Styles. Proceedings of the Tenth International Conference on Technology and Education - Rethinking the Roles of Technology in Education, pp. 1277-1279. USA, M.I.T., Cambridge, U.S.A., March 1993
- Alvino, S., Forcheri, P., Ierardi, M. G. & Sarti, L. (2008). A general and flexible model for the pedagogical description of learning objects. In: Learning to Live in the Knowledge Society, Michael Kendall and Brian Samways (Eds), IFIP International Federation for Information Processing, page numbers 55-62, Springer, 978-0-387-09728-2, Boston
- Arapı, P., Moumoutzis, N. & Christodoulakis, S. (2003). Supporting interoperability in an existing elearning platform using SCORM, Proceedings 3rd IEEE International Conference on Advanced Technologies, pp. 388-393, ISBN 0-7695-1967-9, Athens, Greece, 2003, IEEE Computer Society, Los Alamitos, CA, USA
- Bajraktarevic, N., Hall, W. & Fullick, P. (2003). ILASH: Incorporating learning strategies in hypermedia, Proceedings of AH2003: Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems, pp. 41-53
- Baldoni, M., Baroglio, C., Patti, V. (2004a). Web-based adaptive tutoring: an approach based on logic agents and reasoning about actions. Artificial Intelligence Review, Vol. 22, No. 1, (September, 2004), pp. 3-39, ISSN 0269-2821
- Baldoni, M., Baroglio, C., Patti, V., & Torraso, L. (2004b). Reasoning about learning object metadata for adapting SCORM courseware. In: Int. Workshop on Engineering the Adaptive Web (EAW), L. Aroyo and C. Tasso, pp. 4-13
- Benayache, A. & Abel, M.-H. (2005) Using knowledge management method for e-learning. In: Chiazzeze, G., Allegra, M., Chifari A., Ottaviano S. (eds.) Methods and Technologies for Learning. WIT Press, ISBN 1-84564-155-8
- Bodendorf, F., Schertler, M. & Cohen E. (2005). Producing reusable web-based multimedia presentations, Interdisciplinary Journal of Knowledge and Learning Objects, Vol. 1, pp. 27-44, ISSN 1552-2237

- Brady, A., Conlan, O., Wade, V. & Dagger, D. (2008). Supporting Users in Creating Pedagogically Sound Personalised Learning Objects. In: Adaptive Hypermedia 2008, W. Nejdl et al. (Ed.), pp. 52-61, LNCS 5149 Springer-Verlag, ISBN 978-3-540-70984-8, Heidelberg, Berlin
- Brusilovsky, P. (2004). KnowledgeTree: A distributed architecture for adaptive e-learning, Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, pp. 104-113, ISBN 1-58113-912-8, New York, NY, USA, May, 2004, ACM New York, USA
- Carver, C. A., Howard, R. A., Lane W. D. (1999). Enhancing Student Learning through Hypermedia Courseware and Incorporation of Student Learning Styles, IEEE Transactions on Education, Vol. 42, No. 1, pp. 33-38, ISSN 0018-9359
- Di Bitonto, P. & Roselli, T. (2008) Dynamic tailoring of SCORM sequencing and navigation rules. Proceedings of The 8th IEEE International Conference on Advanced Learning Technologies (ICALT) pp. 373-375, ISBN: 978-0-7695-3167-0, Santander, Spain, July, 2008, Diaz, P., Kinshuk, Aedo, I., Mora, E., (Eds), Advanced Learning Technologies, IEEE Computer Society Press, CA: Los Alamitos
- Di Bitonto, P. ; Roselli, T. & Rossano, V. (2008). A rules-based system to achieve tailoring of SCORM standard LOs. Proceedings of International Workshop on Distance Education Technologies (DET), part of the 12th International Conference on Distributed Multimedia Systems, pp. 378-383, ISBN 1-891706-23-3, Boston, USA, September, 2008, Knowledge System Institute Graduate School, Skokie, Illinois
- Di Bitonto, P. (2009). Adapting SCORM compliant LOs in a knowledge engineering scenario. Submitted to International Journal of Distance Education Technologies ISSN:1539-3100
- Earle, A. (2002). Designing for Pedagogical Flexibility - Experiences from the CANDLE Project. Journal of Interactive Media in Education, Vol. 2002, No. 4, July 2002, ISSN1365-893X
- Felder, R. M. & Silverman, L.K. (1988) Learning and teaching styles in engineering education, Engineering education, Vol. 78, No. 7, 1988 674-681
- Fischer, S., (2001). Course and exercise sequencing using metadata in adaptive hypermedia learning systems. Journal on Educational Resources in Computing (JERIC), Vol.1, No. 1es, (Mar. 2001), ISSN 1531-4278
- Fleming, N.D. (2001). Teaching and Learning Styles: VARK Strategies, Community College, ISBN 0-473-07956-9, Honolulu
- Friesen, N. (2005). Interoperability and learning objects: An overview of e-learning standardization, Interdisciplinary Journal of Knowledge and Learning Objects, Vol. 1, pp. 23-31, ISSN 1552-2237
- Gardner, H. (1983). Frames of mind: the theory of multiple intelligences, Basic Books, ISBN 0465025080, New York
- Gilbert J. E. & Han C. Y. (1999). Adapting instruction in search of 'a significant difference, Journal of Network and Computer Applications, Vol. 22, No. 3, (July 1999), pp. 149-160, ISSN 1084-8045
- Giorgini, P. and Kolp, M. and Mylopoulos, J. and Pistore, M. (2006). Methodologies and Software Engineering for Agent Systems, Springer US, ISBN 978-1-4020-8057-9

- Holohan, E., Melia, M., McMullen, D., and Pahl, C. (2005). "Adaptive E-Learning Content Generation based on Semantic Web Technology", In Proc. Int'l Workshop on Applications of Semantic Web Technologies for E-Learning, Amsterdam, The Netherlands.
- Jovanovic, J., Dragan, G. & Devedzic, V. (2005). TANGRAM: An Ontology-based Learning Environment for Intelligent Information Systems. In G. Richards (Ed.), *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2005* (pp. 2966-2971). Chesapeake, VA: AACE.
- Karampiperis, P., Sampson, D. (2004). Adaptive instructional planning using ontologies. *Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies (ICALT 2004)*, pp. 126-130
- Keefe, J. W. (1985). *Assessment of Learning Style Variables: The NASSP Task Force Model*, Vol. 24, No. 2, Spring, 1985, pp. 138-144, ISSN: 0040-5841
- Kelly D. & Tangney, B. (2006). Adapting to Intelligence Profile in an Adaptive Educational System. *Interacting with Computers*, Vol. 18, No. 3, (May 2006), pp. 385-409, ISSN 0953-5438
- Kontopoulos, E. and Vrakas, D. and Kokkoras, F. and Bassiliades, N. and Vlahavas, I. (2008). An ontology-based planning system for e-course generation, *Expert Systems with Applications*, Vol. 35, No. 1-2, pp. 398-406, ISSN 0957-4174
- Lee, C. H. Lee, G. G. & Leu, Y. (2008). Analysis on the adaptive scaffolding learning path and the learning performance of e-learning, *WSEAS Transactions on Information Science and Applications*, Vol. 5, No. 4, April 2008, pp. 320-330, ISSN 1790-0832
- McGreal, R. & Roberts, T., (2001). A primer on metadata for Learning Objects: fostering an interoperable environment. *E-learning Magazine*, Vol. 10, No. 2, pp. 26 - 29
- Monacis, L., Finamore, R., Sinatra, M., Di Bitonto, P. ; Roselli, T. & Rossano, V. (2009). An Adaptive Learning Object SCORM standard. *Proceedings of International Workshop on Distance Education Technologies (DET)*, part of the 13th International Conference on Distributed Multimedia Systems, pp. 378-383, ISBN 1-891706-23-3, San Francisco Bay, USA, September, 2009. In printing.
- Morales, L. , Castillo, L., Fernandez-Olivares, J. & Gonzalez-Ferrer, A., (2008). Automatic Generation of User Adapted Learning Designs: An AI-Planning Proposal. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, LCNS 5149/2008, (pp. 324-328), Heidelberg, Berlin: Springer-Verlag.
- Peachy, D. R., Mc-Calla, G. I. (1986). Using planning techniques in intelligent tutoring systems. *International Journal of Man-Machine Studies*, Vol. 24, pp. 77-98
- Pollyana N. M. & Ismar F. S. (2006). Learning Objects: Adaptive Retrieval through Learning Styles. *Interdisciplinary Journal of Knowledge and Learning Objects* Vol. 2, pp. 35-46, ISSN 1552-2237
- Popescu, E. (2008). *Dynamic adaptive hypermedia systems for e-learning*. Unpublished doctoral dissertation, University of Craiova, Romania. Retrieved March 3, 2009, from http://hal.archives-ouvertes.fr/docs/00/34/34/60/PDF/thesis_ElviraPopescu.pdf
- R. De Beni, A. Moè, and C. Cornoldi, (2003) *AMOS. Abilità e motivazione allo studio: valutazione e orientamento. Questionario sugli stili cognitivi*. ISBN 9788879465540 Erickson: Trento, 2003.

- Ronchetti, M. & Saini, P. (2003) The knowledge management problem in an e-learning system: a possible solution, Proceedings of the Workshop on Artificial Intelligence and e-learning, Eighth National Congress of Italian Association for Artificial Intelligence, 23-26, Pisa, Italy
- Roselli, T.; Rossano, V. (2006). Describing learning scenarios to share teaching experiences, Proceedings of International Conference on Information Technology Based Higher Education and Training (ITHET), pp 180-186, ISBN: 1-4244-0406, Sydney, Australia, July 2006, IEEE Computer Society Press, Sydney, Australia
- Ruiz, M el P., Diaz, M. J. F., Soler, F. O., & Perez, J. R. P. (2008). Adaptation in current e-learning systems, Computer Standards & Interfaces, Vol: 30, No: 1-2, January, 2008, pp. 62, 70, ISSN 0920-5489
- Russell, S. J., Norvig, P. (2003) Artificial Intelligence: A Modern Approach, Prentice Hall, ISBN 10: 0131052241, NJ: Upper Saddle River
- Sanrach, C., Grandbastien, M. (2000). ECSAIWeb: A web-based authoring system to create adaptive learning systems. In: Brusilovsky, P., Stock, O., Strapparava, C. (eds.) Adaptive Hypermedia 2000, pp. 214-226. LNCS 1892 (2000)
- Seal, K.C. & Przasnyski, Z.H. (2001). Using the World Wide Web for teaching improvement, Computers and Education, (January, 2001) Vol.36 , No.1, pp.33-40, ISSN 0360-1315
- Stash, N. (2007). Incorporating cognitive/learning styles in a general-purpose adaptive hypermedia system, SIGWEB Newsletter, Vol. 2007, No. Winter, ISSN 1931-1745
- Trigano P. C., Pacurar-Giacomini, E. (2004). Toward a Web based Environment for Evaluation and Design of Pedagogical Hypermedia, Journal of Educational Technology & Society, Vol. 7, No. 3, pp. 21-37, ISSN 1436-4522
- Wang, T.I. & Wang, K.T. and Huang, Y.M (2008). Using a style-based ant colony system for adaptive learning, Expert Systems with Applications, Vol. 34, No.4, (month and year of the edition) pp. 2449-2464, ISSN 0957-4174
- Wiley, D. A. (2000). Learning object design and sequencing theory. PhD Thesis. Brigham Young University.
- Wood, M., F., & DeLoach, S., A. (2001). An Overview of the Multiagent Systems Engineering Methodology, Springer Berlin, ISBN 978-3-540-41594-7, Heidelberg
- Wooldridge, M., Jennings N. R. & Kinny, D. (2000) The Gaia Methodology for Agent-Oriented Analysis and Design, Autonomous Agents and Multi-Agent Systems, Vol. 3, No. 3, (September, 2000), pp. 285-312, ISSN 1387-2532

IntechOpen

IntechOpen

Advances in Learning Processes

Edited by Mary Beth Rosson

ISBN 978-953-7619-56-5

Hard cover, 284 pages

Publisher InTech

Published online 01, January, 2010

Published in print edition January, 2010

Readers will find several papers that address high-level issues in the use of technology in education, for example architecture and design frameworks for building online education materials or tools. Several other chapters report novel approaches to intelligent tutors or adaptive systems in educational settings. A number of chapters consider many roles for social computing in education, from simple computer-mediated communication support to more extensive community-building frameworks and tools. Finally, several chapters report state-of-the-art results in tools that can be used to assist educators in critical tasks such as content presentation and grading.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Pierpaolo Di Bitonto (2010). Multi-agent Architecture for Retrieving and Tailoring LOs in SCORM Compliant Distance Learning Environments, *Advances in Learning Processes*, Mary Beth Rosson (Ed.), ISBN: 978-953-7619-56-5, InTech, Available from: <http://www.intechopen.com/books/advances-in-learning-processes/multi-agent-architecture-for-retrieving-and-tailoring-los-in-scorm-compliant-distance-learning-envir>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen