

Personalization and Contextualization of Learning Experiences based on Semantics

<http://dx.doi.org/10.3991/ijet.v9i7.3666>

N. Capuano, G.R. Mangione, A. Pierri, S. Salerno
University of Salerno, Fisciano (SA), Italy

Abstract—Context-aware e-learning is an educational model that foresees the selection of learning resources to make the e-learning content more relevant and suitable for the learner in his/her situation. The purpose of this paper is to demonstrate that an ontological approach can be used to define leaning contexts and to allow contextualizing learning experiences finding out relevant topics for each context. To do that, we defined a context model able to formally describe a learning context, an ontology-based model enabling the representation of a teaching domain (including context information) and a methodology to generate personalized and context-aware learning experiences starting from them. Based on these theoretical components we improved an existing system for personalized e-learning with contextualisation features and experimented it with real users in two University courses. The results obtained from this experimentation have been compared with those achieved by similar systems.

Index Terms—adaptive learning systems, learning context, knowledge representation, learning design.

I. INTRODUCTION

Personalized e-learning is defined in [1] as an educational model that is tailored to the individual learner's needs and interests. For this reason it can be used for developing individual learning programs and engage learners into the learning process so that learner's potentials and success can be optimized. Like standard e-learning, personalized e-learning is not restricted by time, place and learner's other requirements. Differently from standard e-learning, personalized e-learning is mostly focused on learner's preferences and current state to provide the learning content correctly.

Personalized e-learning is useful in contexts with heterogeneous learners i.e. when learners have different cognitive backgrounds and learning preferences. It is also beneficial when learning needs are not standardized but each learner has specific requirements (e.g. in vocational training, in training on the job, etc. [2]). Instead, personalized e-learning is not sufficient when the same e-learning system has to deal with different learning contexts (e.g. to provide personalized learning in different institutions or countries, within different instructional systems, etc.). To deal with these situations, it is needed to move from personalized to context-aware e-learning.

Context-aware e-learning provides a learner with highly customized learning content [3] but, differently from personalized e-learning, the customization of content is made by considering not only learners' needs, preferences and background knowledge but also by selecting or filtering learning resources in order to make the e-learning content

more relevant and suitable for the learner in his/her situation. In other words the filtering process is done also considering several parameters related to the environment where the learning takes place.

The purpose of this paper is to demonstrate that an ontological approach is feasible to define leaning contexts and to allow contextualizing learning experiences, finding out relevant topics for each context. In particular we defined theoretical and technological components needed to extend an already existing system for personalized e-learning with learning contextualization features.

The paper is structured as follows: in the second section, related work about learning context modelling and e-learning course sequencing is presented while the third section briefly describes the e-learning system called IWT [4] that we used as a starting point. The fourth section describes the proposed approach from the theoretical point of view by introducing our definition of learning context and methodological components needed to support it and provide contextualisation features. The fifth section describes the developed prototype while the sixth one summarizes the experimentation performed in University settings as well as a comparison of our approach with similar systems.

II. RELATED WORK

This research falls in the Intelligent Tutoring Systems field and deals with personalized and context aware e-learning. In order to contextualize our research in literature, in the next subsections we discuss some related research about learning context definition and e-learning course sequencing. In section VI we compare our approach with some of those described here.

A. Related Research about Learning Context

Several definitions of *context* are available in literature. According to [5], the context is defined as "that which surrounds, and gives some meaning to, something else". In [6] instead it is defined as "any information that is used to characterize the situation of an entity". Moreover, according to the Ubiquitous Computing, "Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application including the user and applications themselves." [2].

Several studies exist that define a *learning context*. With this term some authors refer to a context of learners involved in learning activities while other authors refer to a context in which a learning object or a learning activity is consumed. We do not distinguish between the two meanings because they are strictly related and, even in the

second case, the authors implicitly refer to the context of learners that are performing a learning activity or consuming a learning object.

The IMS Learning Resource Metadata Information Model [7] defines a learning context as “the typical learning environment where use of learning object is intended to take place”. It also proposes a list of feasible contexts for a learning object i.e. school, higher education and training. Other authors, in [8], define the learning context as the learner environment and talk about three main environments: external (e.g. classroom, working space, in-person coaches, etc.), internal (e.g. previous beliefs, thoughts, hopes, etc.) and digital one.

In [9] the authors define a context model for e-learning as composed of seven levels (technological, pedagogical, methodological, organisational, psychological, related to the subject domain and to the course), each one characterised by several aspects and variants. In [10] a “static” context model for context-aware e-learning has been defined. Its static nature is due to the fact that only parameters that do not change within the entire e-learning course structure have been considered. The defined model is composed of several context parameters.

In [2], the same authors have systematized such parameters into the following sub-contexts: Profile (learner’s personal information, personality type and level of expertise), Preference (learner’s approach to learning, intention and learning style), Infrastructure (learner’s situation in terms of network and device used by the learner) and Learning (learning pace, comprehension level of the learner).

According to [11] the learning context describes a class of learners within a technological infrastructure with a set of parameters related to Learner Category (including the concepts already acquired by the class and the learner cultural background), Educative Modality (including duration and typology of the learning experience) and Educational Objective (including target kind of instruction, kind of expertise needed and related depth level). Based on the context description, the proposed system is able to select suitable learning resources by matching context parameters and fields of IMS Metadata connected to learning resources.

B. Related Research about Course Sequencing

Several adaptive e-learning systems that deal with course sequencing exist in literature. The AHA! (Adaptive Hypermedia for All) system [12] employs adaptive techniques like fragment and link hiding for course delivery. Knowledge domain is modelled using concepts and actual content is kept directly in pages. AHA! is used in some universities in Belgium and the Netherlands.

INSPIRE (INtelligent System for Personalized Instruction in a Remote Environment) [13] offers adaptive link annotation and sorting as well as adaptive curriculum sequencing techniques to guide the learner through a learning space using a path respecting his/her learning style. This system supports the definition of learning goals on a concept graph.

InterBook [14] is a tool for creation and delivery of adaptive electronic textbooks. It offers adaptive link annotation and direct guidance. The domain structure is modelled using a concept network, where each concept represents an atomic part of knowledge. The system monitors

student’s progress about each concept and uses this information to recommend pages with all known prerequisites.

ActiveMath [15] is an adaptive e-learning system that has evolved from a prototype to a full-blown platform used by an international community centred in Germany. The course sequencing algorithm uses information about learning resources, the learner and his/her learning goals to generate an adapted sequence of learning objects that supports the learner in achieving his goals.

NetCoach [16] simplifies the course authoring process by allowing to define relations between documents using concept networks. Adaptive annotation of links and curriculum sequencing features are supported.

ACGs (Adaptive Course Generation System) [17] is an agent-based prototype developed by the Vietnam National University of Hanoi able to provide adaptive curriculum sequencing features based on learner profiles.

INES (INtelligent Educational System) [18] is a prototypical adaptive learning system from the University of Vigo, Spain. It exploits ontologies to model the knowledge about the domain and uses natural language processing technology to communicate with students. Based on student progresses, the system is able to suggest to each learner specific tasks to perform in order to achieve his/her specific learning objectives.

The LIP (Learning In Process) system [19] was aimed at providing personalized and contextualized learning experiences addressing the needs of knowledge intensive organizations. LIP is able to capture different dimensions related to the user context, including user profile, technical constraints, organizational and process-related aspects and to generate courses accordingly through composition of learning resources.

GRUNDEV [10] is a “proof of concept” prototype from the Pondicherry University of India able to recommend the appropriate learning content based on a description of a learning context in terms of profile, preferences, infrastructure and learning.

Many of the considered e-learning systems are still in the prototype stage and very few of them are also able to take into account the context where the learning takes place when building a personalized learning experience.

III. THE STARTING POINT

In this section we introduce a learning platform called IWT (Intelligent Web Teacher) that we have adopted as a basis to apply models and methodologies hereafter defined. IWT allows to generate personalized learning experiences and relies on four interacting models as described below [20].

The *domain model* describes the knowledge that is object of teaching through a set of *concepts* (topics to be taught) and a set of *relations* between concepts. Such structure can be represented as a concept graph $G(C, BT, IRB, SO)$ where C is the set of nodes representing domain concepts while BT , IRB and SO are sets of arcs corresponding to supported relations whose meaning is:

- $BT(a, b)$ means that the concept a belongs to b i.e. b is understood if and only if every a so that a belongs to b is understood;
- $IRB(a, b)$ means that the concept a is required by the concept b i.e. a necessary condition to study b is to have understood a before;

- $SO(a, b)$ means that the *suggested order* between the two concepts is that a precedes b i.e. to favour learning, it is desirable to study a before b .

IWT domain models are represented through the Ontology Web Language (OWL) from W3C. Figure 1 shows a sample domain model in the didactics of artificial intelligence stating that understanding of *logics* depends on understanding of *formal systems*, *propositional logics* and *first order logics* but, before approaching any of these topics it is necessary to have an *outline of set theory* first. Also, *formal systems* must be taught before *propositional logics* and *first order logics* while it is desirable to teach *propositional logics* before *first order logics*.

A set of *teaching preferences* TP can be also linked to domain concepts to define feasible teaching strategies that should be applied for such concepts. They are represented as an application $TP(C \times Props \times PropVals) \rightarrow [0, 10]$ where $Props$ is the set of didactical properties and $PropVals$ is the set of feasible values for such properties. Table I shows the supported didactical properties and related values. Most properties correspond to a selection of fields of IMS Metadata, some other properties (like didactic method) correspond to additional fields we used to describe available learning resources [21].

The *learner model* represents a learner. It is composed of a cognitive state and a set of learning preferences. The *cognitive state* represents the knowledge reached by a learner at a given time and associates an evaluation to each available concept of a domain. *Learning preferences* provide an evaluation of what learning strategies are more feasible for a given learner. Both components are automatically assessed by IWT, which analyses the results of testing activities and the user behaviour in the learning experience as described in [20]. The learner model is maintained in a structure obtained by extending IMS Learner Information Packaging specifications.

The *learning resource model* is a metadata compliant with IMS Metadata specifications, representing a learning resource. It includes the set of concepts that are covered by the learning resource and an additional set of didactical properties representing learning strategies applied by the resource itself.

The *unit of learning model* represents a sequence of learning resources needed for a learner to understand a set of target concepts in a given domain. In [22] we have described the process generating a unit of learning starting from a set of target concepts and from a learner model.

The process, by looking at the domain model, generates a feasible sequence of domain concepts able to teach the target concepts. Then it removes domain concepts already known by the target learner by looking at his/her cognitive state. Eventually, it associates to each remaining concept the best matching learning resources taking into account teaching and learning preferences (connected respectively to concept and learner model). The obtained sequence of learning resources is represented in a structure compliant with the IMS Content Packaging specification.

By leveraging on these models, IWT supports a learner centred approach and is able to build the best unit of learning for each learner from a given set of target concepts. Different learners with the same target concepts will consequently have different units of learning generated by the system.

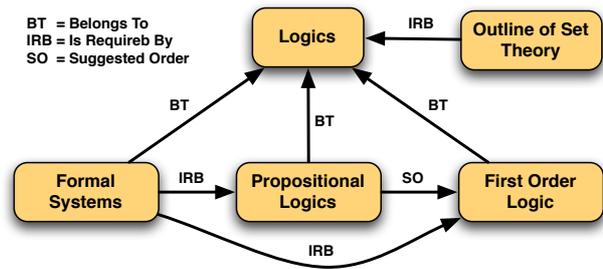


Figure 1. A sample domain model.

TABLE I.
DIDACTICAL PROPERTIES AND FEASIBLE VALUES

Properties	Feasible Values
didactic method	deductive, inductive, etc.
resource type	text reading, video clip, simulation, virtual experiment, etc.
interactivity level	low, medium, high
age range	0-5, 6-10, 11-13, 14-18, 19-24, 25+
language	English, Italian, Spanish, German, etc.
semantic density	low, medium, high
interactivity type	active, expositive, mixed
difficulty	easy, medium, difficult

IV. THE PROPOSED APPROACH

The domain model of IWT is unable to deal with learning contexts. This means for example that different instances of the domain model are needed to teach a given topic in different University faculties or at different school levels: concepts to be taught may vary as well as teaching strategies. To overcome this limitation we propose to give teachers the possibility to define learning contexts as well as domain models and to provide algorithms and tools able to adapt domain models with respect to contexts.

In other words the teachers will be therefore able to define available learning contexts; to include, in domain models, contextualization information that specify how models change according to contexts; to build a contextualized and personalized unit of learning given a context, a learner and a set of target concepts. The proposed approach brings several advantages:

- improves reusability i.e. to repurpose a domain model for a different context, it needs to add a new context profile specifying the due variation instead of creating a new domain model;
- improves scalability i.e. the same domain model can be used to support a virtually infinite number of different units of learning in different contexts;
- improves maintainability i.e. by using the same domain model for different contexts, modifications made in a core part of the model are automatically applied to all contexts.

The next subsection introduces our definition of learning context and defines the context profile that states feasible teaching strategies that may be applied for each available context. Subsequent subsections focus on new models and algorithms we have defined to support domain

model contextualization in IWT, namely: the abstract domain model able to introduce and support the notion of context and the domain contextualisation algorithm able to generate a standard domain model from an abstract one.

A. The Learning Context

Starting from the state of the art presented in section II, we have defined our model of *learning context* by taking into account all aspects of the environment surrounding the learner without considering aspects related to the learner itself such as profile, previous knowledge, learning preferences, etc. that are already considered by IWT in the learner model. We have then organized these aspects into five levels, each one connected with an extensible list of possible values as summarized in table II.

Based on that, a learning context can be defined as a feasible configuration of values associated with context levels. This configuration characterizes a context but does not provide any information about teaching strategies that are most feasible with respect to that context.

To provide this additional information, we introduce a new abstraction named *context profile* i.e. a collection of *teaching preferences* that applies to a whole context rather than to a single concept as seen in section III.

A context profile can be so defined as an application $CXP (CX \times Props \times PropVals) \rightarrow [0, 10]$ where CX is the set of available contexts, $Props$ is the set of didactical properties and $PropVals$ is the set of feasible values for such properties as defined in table I.

B. The Abstract Domain Model

The abstract domain model describes the knowledge that is object of teaching at a higher level with respect to the IWT domain model. It supports the notion of context, it also allows to associate a context to each concept as well as teaching preferences to couples (concept, context) rather than simply to concepts. Such a structure can be represented with:

- a *concepts graph* G representing all concepts that are objects of teaching and relations between them (supported relations are those already used by IWT);
- a set of contexts $CX = \{cx_1, \dots, cx_n\}$ that is a vocabulary of supported contexts;
- a concepts labelling relation $CL \subseteq (C \times CX)$ purposed to label concepts with one or more contexts of CX .

Figure 2 shows a graphical representation of a sample abstract domain model obtained as an extension of the one depicted in the previous section. The set CX is composed of two contexts: *Computer Science Course at University* and *Mathematics Course at University*. The concepts labelling relation CL is graphically represented by circles associated with concepts.

Informally speaking this domain model specifies that understanding of logics depends on understanding of formal systems, propositional logics, first order logics and description logics where first order logics is only required in courses about mathematics and description logics is only required in courses about computer science. In any case, formal systems must be taught before propositional logics and first order logics while it is desirable (but not compulsory) to teach propositional logics before first order logics and first order logics before description logics.

TABLE II. LEARNING CONTEXT MODEL

Level	Description	Values
Educational Context	Identifies the environment where the learning takes place	...
Country		UK, Italy, Spain, Austria, etc.
Educational level		Primary, Secondary, Higher Education, University 1 st or 2 nd Cycle, Post-Grade, Technical School, Professional Formation, Continuous Formation, Vocational Training, etc.
Course Subject Context	Identifies the subject domain	Literature, Mathematics, Physics, Computer Science, etc.
Methodological Context	Identifies the e-learning form and the level of formality to apply in the context	Self-learning, Asynchronous, Synchronous, Blended/Asynchronous, Blended/Synchronous, Informal learning
Instructional Context	Identifies the instructional strategy to apply in the context	Active learning, Collaborative learning, Direct instruction, Drill and practice, Experiential learning, Game based learning, Inquiry learning, Problem based learning, etc.
Technological Context	Identifies the main technological constraints linked to the context	...
Device Constraints		Screen size, computational power, etc.
Network Constraints		Bandwidth, availability, etc.

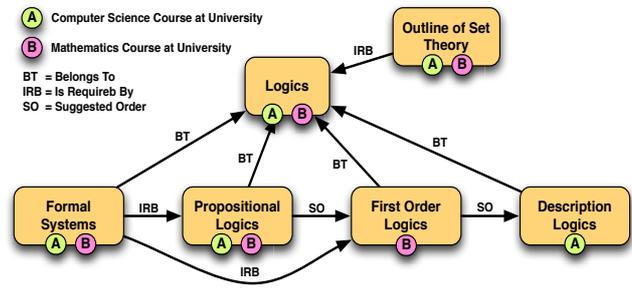


Figure 2. A sample abstract domain model.

To be coherent, the concept graph G must be acyclic and the following coherency properties must be verified for any couple of concepts a and b and for any $cx \in CX$:

- if $BT(a, b)$ then $CL(a, cx) \rightarrow CL(b, cx)$ i.e. if a concept applies to a given context, any descendant on the BT relation applies to the same context too;
- if $IRB(a, b)$ then $CL(b, cx) \rightarrow CL(a, cx)$ i.e. if a concept applies to a given context, any ancestor on the IRB relation applies to the same context too.

Using a context profile CXP as defined in the previous subsection, it is also possible to state feasible teaching preferences that may be applied for any available context in CX . Moreover, a set of additional teaching preferences can be added to define exceptions to the context profile i.e. to specify feasible teaching strategies that may be applied for a given concept in a specific context (so excepting general rules included in the context profile).

Teaching preferences here are defined as an application $TP (C \times CX \times Props \times PropVals) \rightarrow [0, 10]$ where $Props$

is the set of didactical properties and *PropVals* is the set of feasible values for such properties as defined in table I. Since high is the value of a teaching preference, the more the corresponding property value should be preferred while selecting feasible learning resources.

For example, the following definition for *CXP* and *TP* states that the preferred didactic method to be used in the context of a mathematics course at University is the deductive one (the corresponding value of 7 is greater than the value of 4, corresponding to the inductive method). Despite that, to teach *first order logics*, an inductive method should be preferred (to that concept, in fact, the inductive method have a preference degree of 10, greater then the value of 7 corresponding to the deductive one).

- *CXP* (*Mathematics Course at University, didactic method, deductive*) = 7;
- *CXP* (*Mathematics Course at University, didactic method, inductive*) = 4;
- *TP* (*First Order Logics, Mathematics Course at University, didactic method, inductive*) = 10.

To maintain complexity of contextualization as well as usability of editing tools at a feasible level, we have chosen to avoid labelling relations too. This means that, during contextualization, relations are only indirectly contextualized i.e. they can be removed (or propagated) if one or both concepts they connect are removed from the domain model.

C. The Contextualization Algorithm

The contextualisation algorithm is purposed to generate a domain model (as defined in the previous section) composed of a conceptual graph $G' (C', BT', IRB', SO')$ and a set of teaching preferences TP' starting from an abstract domain model composed of a conceptual graph $G (C, BT, IRB, SO)$, a set of contexts CX , a context profile CXP , a labelling relation CL and a set of teaching preferences TP by selecting a $cx \in CX$.

The graph G' only includes concepts that apply to cx and relations connecting such concepts. In other words the G' components can be defined as follows:

- $C' = \{c \in C \mid (c, cx) \in CL\}$;
- $BT' = \{(a, b) \in BT \mid CL(a, cx) \wedge CL(b, cx)\}$;
- $IRB' = \{(a, b) \in IRB \mid CL(a, cx) \wedge CL(b, cx)\}$;
- $SO' = \{(a, b) \in SO \mid CL(a, cx) \wedge CL(b, cx)\}$.

Thanks to coherency properties defined in the previous section, the obtained graph G' is still meaningful and can be used as an input for the unit of learning generation processes. Moreover TP' is built by sequentially executing the following steps:

1. $TP'(c, p, pv) = CXP(cx, p, pv)$ for any $c \in C$, $p \in Props$ and $pv \in PropVals$ (i.e. the teaching preferences for each concept are initially settled to those defined by the selected context profile);
2. $TP'(c, p, pv) = TP(c, cx, p, pv)$ for any defined TP (i.e. if teaching preferences are defined for a given concept in cx , then they override default preferences, otherwise default preferences are used).

Figure 3 shows the two domain models that can be obtained, through the contextualization algorithm, and starting from the conceptual graph in figure 2, by selecting the Mathematics Course context (a) or the Computer Science

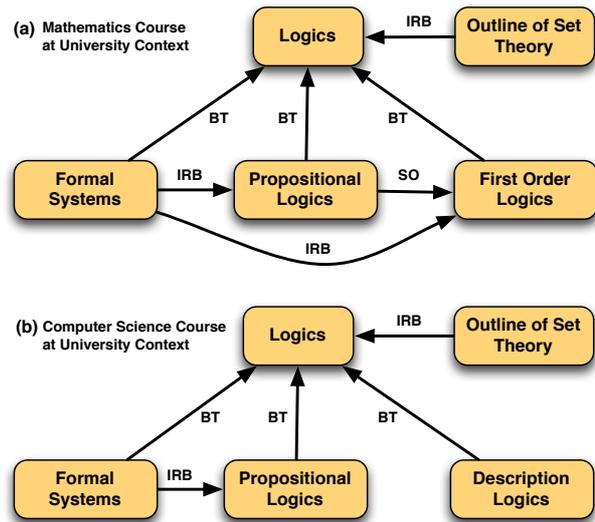


Figure 3. Two samples of contextualized domain models.

Course context (b). Moreover, starting from the sample context profile and teaching preferences, the algorithm will automatically generate the following preferences for the mathematics context:

- $TP'(c, didactic\ method, deductive) = 7$ for any $c \in C$;
- $TP'(c, didactic\ method, inductive) = 4$ for any $c \in C$ different from first order logics;
- $TP'(first\ order\ logics, didactic\ method, inductive) = 10$.

The contextualization algorithm has been integrated in the unit of learning generation process of IWT (defined in the previous section) in order to obtain different sequences of learning resources not only based on the learner model of the involved learner (personalization) but also on the context where the learner learns (contextualization). In particular, given an abstract domain model, a learner and a set of target concepts, the abstract domain model is first of all contextualized with respect to the context the learner belongs to. Then the unit of learning generation process is applied on the contextualized domain model.

D. Improvements

In some cases, the contextualization algorithm breaks *SO* relations useful to correctly order concepts during the unit of learning generation process. For example in figure 2, if we follow *SO* relations, we see that *propositional logics* should be explained before *first order logics* that, again, should be explained before *description logics*. By removing first order logics in the computer science context (figure 3b) we also break the *SO* chain between propositional logics and description logics that can be now explained by the system in any order.

Thanks to coherency properties, defined in IV, this can only happen on *SO* relations. To fix this, we introduce a set of relation propagation rules for *IRB* and *SO* and revise the previous algorithm as follows: all G' components are initialised to be equal to those of G ; then, for each concept c that does not apply to the context cx , *IRB* and *SO* relations are propagated over the concept c ; then the concept c is removed.

The following propagation rules are applied for any concept c that is going to be removed:

- for each incoming *IRB* relation from a concept *a* and for each outgoing *IRB* relation to a concept *b*, a *IRB* relation from *a* to *b* is added (figure 4a);
- for each incoming *IRB* or *SO* relation from a concept *a* and for each outgoing *SO* relation to a concept *b*, a *SO* relation from *a* to *b* is added (figure 4b);
- for each incoming *SO* relation from a concept *a* and for each outgoing *IRB* or *SO* relation to a concept *b*, a *SO* relation from *a* to *b* is added (figure 4c);

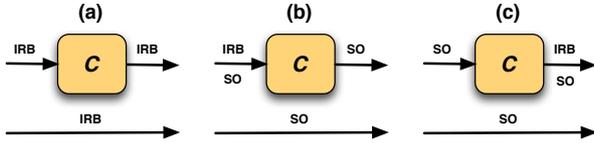


Figure 4. The relation propagation rules.

The number of relations introduced by these rules is a combination of the number of existing relations i.e. for a concept *c* having *n* incoming relations and *m* outgoing relations (of *IRB* and *SO* types), the number of introduced relations is $n \times m$ while the number of removed relations is $n + m$. Figure 5 shows the domain model obtained through the contextualization algorithm, starting from the conceptual graph in figure 2, by selecting the computer science context and by applying relation propagation rules. As it can be seen, a new *SO* relation between propositional logics and description logics is introduced.

V. THE DEVELOPED PROTOTYPE

In this section we introduce the architecture of IWT and the extensions we made to integrate software components implementing models and methodologies defined so far.

A. IWT Logical Architecture

The logical architecture of IWT is divided into four main layers as depicted in figure 6. The first layer is the *IWT Framework* used by developers to design and implement core and application services. The second layer is composed of *Core Services* providing basic features for the management of users, roles, resources, metadata and ontologies, for user profiling and learning personalization.

The third layer is composed of *Application Services* used as building blocks to compose learning applications for specific domains. They include blocks for learning and learning content management, for ontology management, for communication and collaboration. On top of the stack, *Learning Applications* covering specific learning scenarios are built.

IWT server-side components are developed in .NET technology and use Microsoft SQL Server for persistency. IWT is an extensible system both at a learning resources level (with drivers i.e. software components able to edit, manage and deliver a specific kind of resource) and at services level (with plug-ins i.e. software components providing specific back-end services).

B. Extension of IWT

The integration of context management and knowledge model contextualization features have impacted on the second and third levels of the IWT architecture as shown in figure 7 where additional components are represented as grey blocks while improved ones are in black. In the

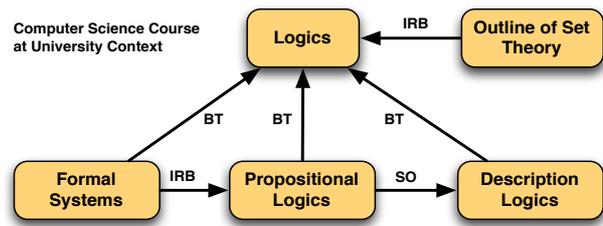


Figure 5. Results of the application of relation propagation rules.

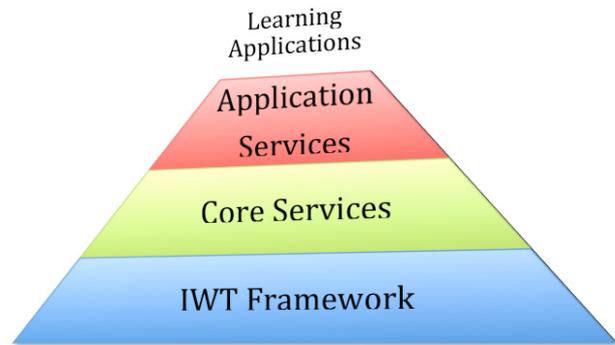


Figure 6. The IWT Logical Architecture.

following list we describe developed and improved components and their impact on the architecture.

- The *Unit of Learning Builder* offers personalization features. The updated version can deal with abstract domain models and implements the contextualisation algorithm defined so far.
- The *Domain Models Storing and Retrieving Service* is responsible for the persistency of defined abstract domain models.
- The *Learner Models Manager* allows teachers to modify learner models. It has been improved to allow the association of a context to a learner.
- The *Context Editor* is a new component that allows knowledge managers to create and modify the list of contexts managed by the system and referenced by learner models and domain models.
- The *Abstract Domain Model Editor* is a component that allows to visually build an abstract domain model including concepts, relations, context profiles, context labels and teaching preferences.
- The *Contextualized Course Manager* is a component that manages contextualised courses. It allows to select target concepts on an abstract domain model and to optionally select a context. If no context is selected at design time, then it is chosen at run-time as the context which the learner belongs to.

The first four components have been developed in C# as IWT plug-ins. The *Abstract Domain Model Editor* is a client-side Java Applet. It communicates with other components through a Web Service interface. Eventually, the *Contextualized Course Manager* has been developed in C# as an IWT driver.

C. The prototype at Work

Figure 8 shows the user interface of the abstract domain model editor. Available contexts are listed on the left side of the window (the two contexts *Computer Science*

Course at University and Mathematics Course at University are displayed.). The user can add, remove or rename contexts by exploiting menu items over the list. A colour is associated to each context. Domain concepts are listed under contexts. The user can add, remove or rename concepts by exploiting menu items over the list. Available relations are listed under the list of concepts.

The user can drag a concept from the list and drop it in the workspace. Dropped concepts are represented as rounded boxes with the concept name and a sequence of coloured circles, one for each available context. Concepts can be moved and connected with available relations.

By double clicking on a context, the context profile is shown and connected teaching preferences can be settled. By clicking on a coloured circle inside a concept, teaching preferences for the concept (in the corresponding context) can be modified. Such preferences are initially settled according to the context profile but may be changed. Once preferences for a concept in a context are changed, the corresponding context label takes the shape of a diamond to emphasise the modification.

Context labels can be removed from concepts by right clicking on them and selecting the “remove” menu item. Removing a context label from a concept means that the concept is not active in that context anymore. A designed model can be saved, printed, redrawn and zoomed. Undo and redo buttons are also provided to remove and redo the last performed action.

Once the abstract domain model is saved and the teacher has connected feasible learning resources to each concept, the system is able to build personalized and contextualized units of learning according to the context that is associated to each learner who enrolls in the course. As an example, figure 9 shows the preview of a sample unit of learning about logics for computer scientists obtained from the abstract domain model shown in figure 8 as well as the related contextualized model.

VI. PROTOTYPE EVALUATION

In this section we provide elements for the evaluation of defined models and methodologies as well as of the related prototype. Two points of view are considered, functions and effectiveness: the former is evaluated against similar systems while the latter is evaluated through an experimentation made with real users. The results of these two analyses are reported below.

A. Comparison with Similar Systems

As seen in section II, although research in adaptive e-learning is very active, few complete course sequencing systems currently exist. Moreover, among the surveyed systems, context adaptation features are provided by only two research prototypes. Table III compares the surveyed systems and prototypes together and with respect to our prototype for contextualized e-learning.

As it can be seen, our system, also thanks to the integration of other functions coming from the reference platform IWT, is able to offer the greatest set of adaptive techniques i.e. *Adaptive Course Sequencing*, *Page Link Annotation* and *Content Recommendation* [23].

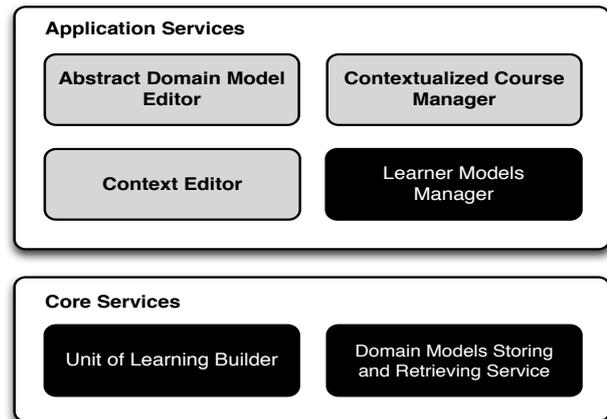


Figure 7. Developed additional IWT components.

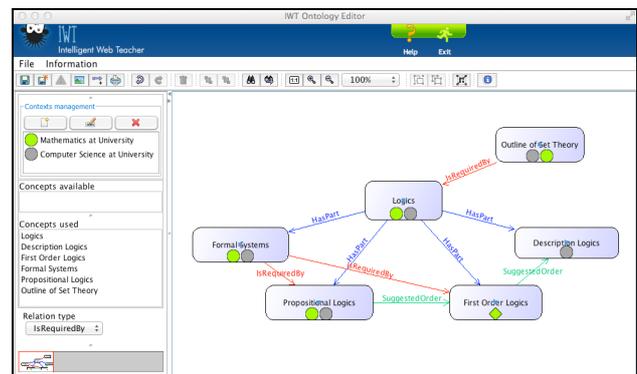


Figure 8. Interface of the abstract domain model editor.

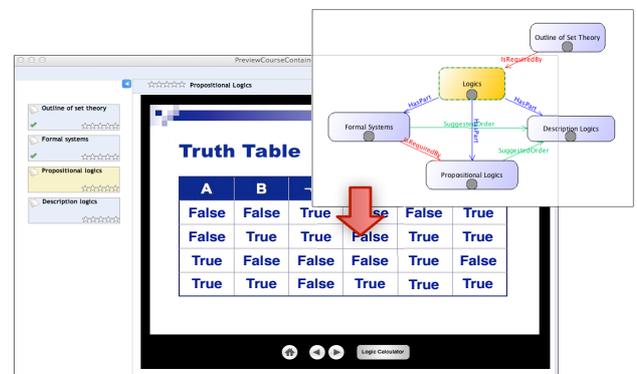


Figure 9. Sample domain of logics contextualized for computer scientists and related unit of learning.

Moreover our prototype, together with GRUNDEV and LIP, is the only one dealing with learning contexts. Despite that, the LIP context model takes into account only few dimensions with respect to our model and it is thought specifically to address needs of knowledge intensive organizations, differently from our general purpose model. GRUNDEV presents a comprehensive context model that unifies in a single view information about learner, environment and technological infrastructure. Despite that, the existing system is only a proof of concept that is still far from a fully working prototype.

TABLE III.
COMPARISON WITH SIMILAR SYSTEMS.

System	Status	Adaptive Techniques	Adaptation based on:
AHA!	Full System	Text fragment hiding, Page link annotation	Knowledge
INSPIRE	Full System	Adaptive course sequencing, Page link annotation	Knowledge and Preferences
InterBook	Full System	Content Recommendation, Page link annotation	Knowledge
ActiveMath	Full System	Adaptive course sequencing, Content Recommendation	Knowledge and Preferences
NetCoach	Full System	Adaptive course sequencing, Page link annotation	Knowledge
ACGs	Prototype	Adaptive course sequencing	Knowledge and Preferences
INES	Prototype	Content Recommendation	Knowledge
LIP	Prototype	Adaptive course sequencing, Content Recommendation	Knowledge, Preferences and Context
GRUNDEV	Prototype	Content Recommendation	Knowledge, Preferences and Context
Our Prototype	Prototype	Adaptive Course Sequencing, Page Link Annotation, Content Recommendation	Knowledge, Preferences and Context

B. Experimentation with Real Users

To evaluate theoretical and technological components defined so far, two experiments were made with real users. A *first experiment* involved two lecturers from two different Universities in the creation of a contextualized course on Scientific Working. The experiment consisted of two sessions conducted in one week.

In the *first session*, each of the lecturers was asked to find concepts and to create a domain model of his/her course on a paper. Then the lecturers were asked to share and discuss their models. As a next step, the lecturers tried to find common concepts for the course together, but also defined individual concepts for each course. Finally, they created a paper-pencil version of a domain model based on these concepts.

In the *second session*, the lecturers created a course on IWT using the concepts they developed in session one. To do that, they created an abstract domain model with two contexts, one for each University (the defined model is shown in figure 10). Then the two lecturers provided feasible learning resources covering identified concepts and proceeded with the creation of a contextualized unit of learning covering all domain concepts.

The focus of this study lied on possible improvements for the tool from the instructor’s viewpoint. So we analysed the tool usability. Both lecturers were asked to fill in the System Usability Scale (SUS) [24] after the creation of customized units of learning by answering 10 questions with values ranging from 1 to 5. The average score obtained for usability is 61.25 (SD 5.3) on a scale ranging from 0 to 100. Considering that the tool is still in a prototype stage, it is very promising.

From an interview made to lecturers, it resulted that both enjoyed the idea of contextualizing the same course

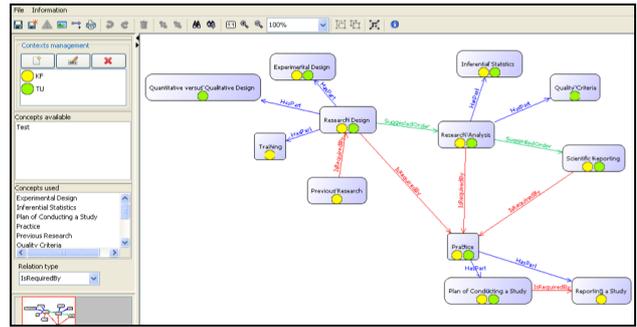


Figure 10. The abstract domain model on Scientific Working made during the experiment.

for both universities as well as the possibility to personalize it with respect to specific student’s needs. Especially the design was described as appealing. Furthermore, the course construction allowed an interdisciplinary exchange and a collaborative work. Both lecturers were in favour of the idea and the tool functions. Indeed, they were convinced that the tool can be used in an educational context in order to support students in their learning process.

Regarding positive aspects of the system, the teachers would have liked to use the system more frequently. Additionally, they did not find the system unnecessarily complex or inconsistent but found its various functions to be well integrated. Concerning negative aspects, the teachers needed to learn how to use the system before being able to adequately use it, in fact, one of the teacher would also need technical support.

A *second experiment* involved 24 students from one of the two universities involved in the first experiment. Participants were between 23 and 29 years old; 18 of them were male and 6 were female; 15 of them finished their Bachelor, 9 of them achieved a Master degree.

The students were divided in two homogeneous groups of 12. The first (control) group was assigned to a static course covering all concepts of the abstract domain model regardless of the context. The second (experimental) group was assigned to the contextualized course defined in the first experiment. Both courses ended with an assessment covering all domain concepts.

All students received a questionnaire including two sections: open questions regarding the course and SUS. The first section included three questions asking whether the course (a) was in line with students’ need, previous knowledge, and learning preferences, (b) contributed to improve understanding of domain concepts, (c) was deemed a worthy educational resource. Answers were provided on a 5 points *Likert scale* ranging from (1) *strongly disagree* to (5) *strongly agree*.

The average ratings to the three questions are summarized in figure 11. As it can be seen, the average ratings coming from the experimental group are sensibly higher than those provided by the control group.

In particular, the experimental group students received personalized and contextualized learning material taking into account not only the context but also their previous knowledge and specific learning preferences. This is visible in the results coming from question (b). In particular half of the students belonging to the control group commented that they already knew most of the learning matters.

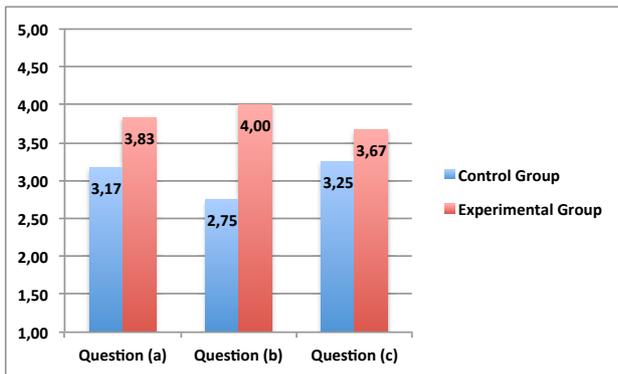


Figure 11. Average ratings for questions on the contextualized course.

In addition to the results coming from question (a) it is also important to consider the results coming from an assessment session performed at the end of the course. From 6 possible points (maximum score), the average overall score in the final assessment was 4.07 (SD 2.12) for the control group and 4.75 (SD 1.03) for the experimental group. Regarding the activity level, the control group students spent on average 35 minutes on the learning system (SD 13.61) while the students from the experimental group spent on average 56 minutes (SD 21.45).

As described above, we also adopted SUS to investigate students' experiences with the system functionality. After calculating the SUS score for each student, we got an average SUS score of 62.19 (SD 17.90) for the static course and an average SUS score of 62.81 (SD 14.09) for the contextualized one. So the students evaluated the usability of IWT in both courses equally. Figure 12 puts together average SUS scores obtained by teachers and students groups.

Almost all of the students liked the idea of the system and stated that it was easy to use. Nonetheless, some of them suggested that other interactive options, such as editing the text or adding notes had to be integrated. For some students it was also important to improve the design and performance of the system. They also proposed to give more ways to access additional information about specific parts of the learning content.

VII. CONCLUSIONS AND FUTURE WORK

This paper describes a research aimed at the definition of theoretical and technological components for context-aware e-learning. Starting from an existing e-learning system able to provide personalized learning (taking into account learner's cognitive state and learning preferences), we added a set of features to provide contextualized learning (also taking into account the environment where the learning takes place).

We defined a learning context model, an ontology-based model able to represent a teaching domain that includes contextualization information and an algorithm to generate personalized and context-aware learning courses based on such structures. To experiment the defined models and algorithms, a prototype was also developed and experimented as well as compared with similar course sequencing systems. The results of both experimentation and comparison are positive and encouraging.

We are currently studying the possibility to define an algorithm to automatically obtain a context profile starting from a learning context model as well as the possibility to

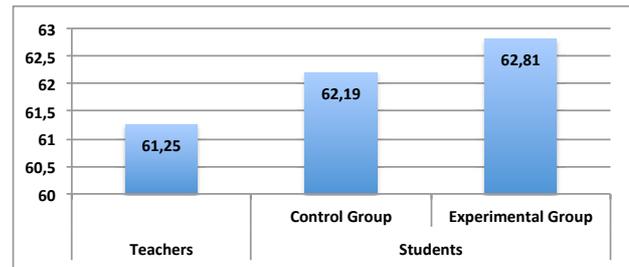


Figure 12. Average ratings for System Usability Score.

extend the list of didactical properties in order to support all levels of the defined context model. Presently, in fact, existing properties can only deal with the first three levels of the defined model (i.e. educational, course subject and, partially, methodological).

An effort aimed at improving the description of the context model by referencing existing ontologies like [25] and [26] for the description of the technological level is an on-going activity together with a study on the use of ontologies to model contexts and relations between contexts.

The possibility to define and use templates for teaching preferences is going to be explored too. In such a way a template can be assigned to a context, to a concept or to a couple (concept, context) instead of defining extensive set of preferences in each one of these cases.

We are also working to improve the usability of the abstract domain model editor and of the contextualized course manager to make contextualization features more user friendly and appealing to both teachers and learners. A further experimentation with more participants is also planned in order to provide additional indicators about the effectiveness of the proposed approach as well as about the most promising directions of improvement.

VIII. ACKNOWLEDGEMENTS

The research reported in this paper was partially supported by the European Commission under the Collaborative Project ALICE [27] "Adaptive Learning via an Intuitive, interactive, Collaborative, Emotional system", VII Framework Program, Theme ICT-2009.4.2, Grant Agreement n. 257639.

REFERENCES

- [1] Thyagarajan, K. Nayak, R. (2007). Adaptive Content Creation for Personalized E-learning Using Web Services. *Journal of Applied Sciences Research*, vol. 9, no. 3, pp. 828-836.
- [2] Capuano, N., Gaeta, M., Ritrovato, P., Salerno, S. (2008). How to Integrate Technology Enhanced Learning with Business Process Management. *Journal of Knowledge Management, Emerald Group Publisher*, vol. 12 no. 6, pp. 56-71.
- [3] Das, M., Bhaskar, M., Chithralekha, T., Sivasathya, S. (2010). Context Aware E-learning System with Dynamically Composable Learning Objects. *International Journal on Computer Science and Engineering*, vol. 2, no. 4, pp. 1245-1253.
- [4] Adorni, G., Battigelli, S., Brondo, D., Capuano, N., Coccoli, M., Miranda, S., Orciuoli, F., Stanganelli, L., Sugliano, A., Vivanet, G. (2010). CADDIE and IWT: Two Different Ontology-Based Approaches to Anytime, Anywhere and Anybody Learning. *Journal of e-Learning and Knowledge Society*, vol. 6, no. 2, pp. 53-66.
- [5] Howe, D. (2006). Free online dictionary of computing. Imperial College Department of Computing London, UK.
- [6] Dey, A., Anind, K. (2001). Understanding and Using Context, *Personal Ubiquitous Computing*, vol. 1, no. 5, pp. 4-7. <http://dx.doi.org/10.1007/s007790170019>

- [7] IMS Global Learning Consortium (2006). IMS Metadata Best Practice Guide for IEEE 1484.12.1-2002 Standard for Learning Object Metadata.
- [8] Downes, S. (2004). What is a Learning Context? Stephen's Web, available on-line at <http://www.downes.ca/post/18>.
- [9] Tankelevičienė, L., Damaševičius, R. (2009). Towards a Conceptual Model of Learning Context in e-learning. Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies.
- [10] Das, M., Chithralekha, T., SivaSathya, S. (2010). Static Context Model for Context Aware e-learning. International Journal of Engineering Science and Technology, vol. 6, no. 2.
- [11] Albano, G., Gaeta, M., Salerno, S. (2006). E-learning: a model and process proposal. International Journal of Knowledge and Learning, vol. 2, no.1/2, pp. 73 - 88. <http://dx.doi.org/10.1504/IJKL.2006.009680>
- [12] De Bra, P., Stash, N., Smits, D., Romero, C., Ventura, S., (2007). Authoring and Management Tools for Adaptive Educational Hypermedia Systems: The AHA! Case Study, in: Studies in Computational Intelligence (SCI) nr. 62, pp. 285-308, Springer Verlag.
- [13] Papanikolaou K., Mabbott, A. Bull S., Grigoriadou M., (2006). Designing Personalised Educational Interactions Based on Learning / Cognitive Style and Learner Behaviour. Interacting with Computers: The Interdisciplinary Journal of Human-Computer Interaction 18(3), 356-384. <http://dx.doi.org/10.1016/j.intcom.2005.11.003>
- [14] Brusilovsky, P., Eklund, J., and Schwarz, E., (1998). Web-based education for all: A tool for developing adaptive courseware. Computer Networks and ISDN Systems, Proceedings of the 7th International World Wide Web Conference, 14-18 April 1998, 30 (1-7), 291-300.
- [15] Melis, E., Goguadze, G., Libbrecht, P., Ullrich, C. (2009). ActiveMath - A Learning Platform With Semantic Web Features in Ontologies and Semantic Web for e-learning.
- [16] Weber, G., Kuhl, H.-C., Weibelzahl, S. (2001). Developing adaptive internet based courses with the authoring system NetCoach. In Bra, P.D., Brusilovsky, P. and Kobsa, A. (eds.) Proc. of 3rd workshop on Adaptive Hypertext and Hypermedia, 35-48.
- [17] Viet, A. N., Si, D. H., (2006). ACGs: Adaptive Course Generation System, An Efficient Approach to Build E-learning Course. Proceedings of The 6th IEEE International Conference on Computer and Information Technology.
- [18] Mikic Fonte, F.A., Rial, J.C.B., Llamas-Nistal, M., Hermida, D.F. (2009). Using semantics in INES, an Intelligent Educational System. Frontiers in Education Conference, 39th IEEE, San Antonio, TX, 18-21.
- [19] Nabeth, T., Angehrn, A.A., Balakrishnan, R. (2004). Integrating "Context" in e-learning Systems Design. Proceedings of the IEEE International Conference on Advanced Learning Technologies, IEEE Computer Society Washington, DC, USA.
- [20] Capuano, N., Gaeta, M., Miranda, S., Orciuoli, F., Ritrovato, P. (2008). LIA: an Intelligent Advisor for e-Learning. M. Lytras, J. Carroll, E. Damiani, R. Tennyson (Eds.), Emerging Technologies and Information Systems for the Knowledge Society, LCNS, vol. 5288, pp. 187-196, Springer-Verlag.
- [21] Capuano, N., Gaeta, M., Salerno, S., Mangione, G. (2011). An Ontology-Based Approach for Context-Aware e-Learning. F. Xhafa, L. Barolli, M. Koppen (Eds.) Proceedings of the 3rd International Conference on Intelligent Networking and Collaborative Systems, Fukuoka, Japan, IEEE Computer Society.
- [22] Capuano, N., Gaeta, M., Orciuoli, F., Ritrovato, P. (2009). On-Demand Construction of Personalized Learning Experiences Using Semantic Web and Web 2.0 Techniques. I. Aedo, N. Chen, Kinshuk, D. Sampson, L. Zaitseva (Eds.), Proceedings of the 9th IEEE International Conference on Advanced Learning Technologies, Riga, Latvia, pp. 484-488.
- [23] Capuano, N., Gaeta, M., Ritrovato, P., Salerno, S. (2014). Elicitation of Latent Learning Needs Through Learning Goals Recommendation. Computers in Human Behavior, Elsevier, vol. 30, pp. 663-673. <http://dx.doi.org/10.1016/j.chb.2013.07.036>
- [24] Brooke, J. (1996). SUS: A "quick and dirty" usability scale. In Usability evaluation in industry. London: Taylor & Francis.
- [25] World Wide Web Consortium (2010). Delivery Context Ontology, J.M. Cantera Fonseca, R. Lewis (eds.), W3C Working Group Note.
- [26] Poveda-Villalón, M., Suárez-Figueroa, M.C., García-Castro, R., Gómez-Pérez, A. (2010). A Context Ontology for Mobile Environments, In Proc. of Context, Information and Ontologies, co-located with EKAW 2010, Lisbon, Portugal.
- [27] Capuano, N., Mangione, G.R., Pierri, A., Lin, E. (2013). Engaging e-learning for risk management: the ALICE experience in Italian schools. L. Barolli, F. Xhafa, H. Chen, A. Skarmeta Gómez, F. Hussain (Eds.), Proceedings of the 7th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS 2013), Taichung, Taiwan, pp. 367-372.

AUTHORS

Nicola Capuano is research assistant at the University of Salerno. His main research interest is artificial intelligence and, among its applications, intelligent tutoring systems and knowledge representation. He works as a project manager and research consultant within several research and development projects. He is author of several scientific papers. He is scientific referee and member of editorial boards for International journals and conferences (e-mail: ncapuano@unisa.it).

Giuseppina Rita Mangione has a PhD in Telematics and Information Society. She is member of the editorial board of "Journal of e-Learning and Knowledge Society" and member of the Elsevier editorial reviewers for Computers & Education. Her main research interests are: technology enhanced learning, adaptive learning system, CSCL and microadaptivity, metacognitive and self-regulated learning.

Anna Pierri has a PhD in Mathematics from the University of Salerno. She is author of more than 20 scientific papers. She has been contract professor of Computer Science and is currently a contract professor for the course of Computerized and Telematics Skills at the University of Salerno. Her research activity deals with Learning Platforms, Artificial Intelligence, Semantic Web and Ontologies.

Saverio Salerno is Full Professor of Mathematics at the Faculty of Engineering of the University of Salerno. He is author of several papers regarding Mathematical Analysis, Operational Research, Simulation, Learning, Knowledge, Computer Science. He has been the Italian Delegate for the IST programme of the EU FP5; he has been expert of the European Commission for Education and Training and member of the Advisory Committee of the Italian Ministry of Education, Universities and Research.

This article is an extended and modified version of a paper presented at the Third International Workshop on Adaptive Learning via Interactive, Collaborative and Emotional approaches (ALICE 2013), held on July 3-5, 2013, in Taichung, Taiwan, in conjunction with the 7th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS 2013). Manuscript received 21 November 2013. Published as re-submitted by the authors 27 April 2014.