An Approach To Personalized e-Learning
Matteo Gaeta¹, Sergio Miranda¹, Francesco Orcioli¹, Stefano Paolozzi², Antonella Pocè³

¹Università degli Studi di Salerno - DIIMA, Fisciano, Italy
{gaeta,smiranda,orcioli}@diima.unisa.it
²Centro di Ricerca in Matematica Pura ed Applicata - CRMPA, Fisciano, Italy
paolozzi@crmpa.unisa.it
³Università degli Studi Roma TRE - DIPED, Roma, Italy
pocè@uniroma3.it

ABSTRACT
This paper focuses on the concept of personalized e-Learning for the computer science (or informatics) education. Several authors have stated that personalization, in educational context, allows executing more efficient and effective learning processes. On the other side the use of Semantic Web technologies (e.g. ontologies) is more and more often considered as a technological basis for personalization in e-Learning (the so-called self-regulated learning). In this paper we describe how personalization can be exploited in e-Learning systems, focusing on our proposal: the Intelligent Web Teacher (IWT). Therefore we present the evaluation of our personalization tools tested in real academic courses, where e-Learning activities are carried out to complement the traditional lectures.

Keywords: e-Learning, Learning Management System, Ontologies, Personalization, Semantic Web.

1. INTRODUCTION
Recently, e-Learning has become an active field of research and experimentation, with remarkable investments from all parts of the world. It represents the Web-based delivery of personalized, comprehensive, dynamic learning contents, aiding the development of communities of knowledge, linking learners and practitioners with experts.

E-Learning supports the different phases of traditional learning and in some cases it is the only possible method of learning, allowing knowledge acquisition also in particular conditions (e.g. impaired students, absence of teaching structures, etc.). In this context, an important role is played by the definition of educational structure that must be contextualized and tailored on the basis of the requirements of: i) teachers, who have personal teaching approaches, and ii) students, who have personal studying approaches.

Traditional teaching methods used in this context, typically followed a ”one size fits all” approach: the information offer was standardized and equal for everyone. In recent times, on the contrary, it has become more clear that different people learn in different ways and that a personalized approach can improve the learning process helping people becoming effective life-long learners [4] [8].

Personalization helps learner in developing a feeling of competence and autonomy [1], because they are trusted with the management of their own learning process. The topic of personalization is strictly related with the deep changes involving educational systems: the shift from a teacher-centered perspective to a learner-centered, competency-oriented one.

Since it would be difficult to produce several personalized courses for each learner’s cognitive state and preferences, Learning Management Systems (LMSs) have a fundamental importance for education through new technologies, because they allow a modular approach to the content creation process and can track the learner’s performance. Researches in this field (see for example [1]) demonstrated that it is better to let learners consciously construct some parts of their learning profile: the one concerning their learning goals and preferences. On the contrary, other profile sections can be constructed automatically, by tracking each learner’s academic achievements in the e-Learning environment and inferring their current cognitive state. In order to allow an automatic adaptation of learning activities on the basis of learner’s needs, it is also necessary to represent, for instance, learning content properties in a machine-understandable way: the system must understand what are the concepts associated to each content and what are content properties (e.g. if it is a text or a video) in order to provide learners with the contents that best suit them.

For these reasons, it should be important to provide semantic structures, which on one hand allow the definition of the particular educational domain and, on the other side, provide the learning modalities. Ontologies represent the most suitable semantic structure for these purposes.

Gruber, in [12] and [13], affirms that ontologies are an explicit and shared specification of a conceptualization. He also explains that a common ontology defines a vocabulary by which queries and assertions are exchanged among different agents (both human and software agents).

The formal representation of knowledge, and in particular the use of ontologies, has played an important role in many e-Learning projects.

These techniques are also useful for Computer Science learning, as can be seen in the development of an educational domain ontology for C-Programming, described in [16]. The ontology consists of a central node (C Programming) and a set of second level entities, connected to it, representing abstract meta-concepts (Syntax, Programming-Techniques, Platforms), that are further subdivided in more concrete concepts. In [16], the authors used an ontology design approach described in [11], which is based on the following elements: i) development of a glossary, by gathering all the information relevant to the
The main contributions of this paper can be summarized in three points:

- an ontology-based approach to address the personalization issues in e-Learning;
- the application of the proposed approach in a complete e-Learning system;
- the presentation of the experimental results obtained through the testing the overall system in a real scenario of informatics-related courses.

The rest of the paper is structured as follows. Section 2 describes the main elements of our e-Learning system: the Intelligent Web Teacher, focusing of the knowledge model and the personalization preferences system. Section 3 describes an interesting case study in which our approach can be exploited. The results of our experiments are reported in Section 4. Finally, Section 5 concludes.

2. THE INTELLIGENT WEB TEACHER APPROACH

The Intelligent Web Teacher (IWT) [6] is primarily an e-Learning platform that enables the definition and the execution of personalized learning experiences, packaged in a Unit of Learning (UoL) (i.e. a course, a module or a lesson structured as a sequence of Learning Activities represented by Learning Objects and Learning Services). The foundation element of the UoL building process is the Learning Model described in [1]. The Learning Model allows to automatically generate a UoL and to dynamically adapt it during the learning process according to the learners’ preferences and cognitive state (personalization process).

The Learning Model can be seen as divided in two layers:

- **Knowledge Layer** in which there are all the machine-understandable representations of the educational domains, learning objects and other relevant entities that we use in our approach;
- **Computational Layer** which contains a set of algorithms that leverage the information of the first layer to execute the personalized e-Learning experiences building process.

Both teachers and students interact with the Knowledge Layer, providing respectively new artifacts and personal information. In order to achieve the expected adaptation capability, the Learning Model uses two specific sub-models: the Knowledge Model and the Learner Model, which are exploited by a specific process used to define personalized e-Learning experiences (at Computational Layer).

The Knowledge Model

The **Knowledge Model** [7] is used to represent the subset of the educational domain that is relevant for the e-Learning experience. Educational domains are modeled using ontologies with an approach similar to that of Topic Maps. In our approach the vocabularies are composed of terms representing subjects that are relevant for the educational domain we want to model. In IWT an e-Learning ontology can be represented with a graph in which nodes are relevant concepts (arguments, topics, etc.) within the educational domain of interest and edges are binary relations between two concepts.

The most important relations of the model are: **HasPart** (HP) that is a part-of relation and **IsRequiredBy** (IRB) that is an order relation. There are other relations which characterize the model by but they are not completely relevant for the sake of this article (for a complete reference see [7]). It is important to observe that when we refer to concepts in the e-Learning ontologies we are referring to the subjects of the educational domain we are modeling.

Let us now consider how to build an e-Learning ontology.

Supposing we have to model the educational domain $D$ (depicted in Figure 1), we try to conceptualize the knowledge underlying $D$ and find a set of terms representing its relevant concepts. The result of the previous step is the list of terms $T = \{C_1, C_2, C_3, C_4\}$ where $T$ is one of the plausible conceptualizations of $D$. The existence of the relations **HasPart**(C,C$_3$), **HasPart**(C,C$_2$) and **HasPart**(C,C$_1$) means that in order to learn a subject $C$ learners have to learn subjects $C_1$, $C_2$, and $C_3$ without considering a specific order. If we add the relations **IsRequiredBy**(C$_3$,C$_2$) and **IsRequiredBy**(C$_2$,C$_1$) to the previous set of relations we can state that $C_1$ has to be necessarily learned before $C_2$ and $C_3$ has to be necessarily learned before $C_2$.

Now, we would like to introduce the Learning Objects (LOs). You can interpret the connection between a concept and a LO, for instance $C_1$ and LO$_1$, as a **HasResource** (HR) relation. The relation **HasResource**(C$_1$,LO$_1$) means that the educational content packaged in Learning Object LO$_1$ explains concept C$_1$. Therefore, if we assume that **HasResource**(C$_1$,LO$_1$), **HasResource**(C$_2$,LO$_2$), **HasResource**(C$_3$,LO$_3$) and that our Learning Objective is $C_1$ then the corresponding assembled e-learning experience is composed only by [LO$_1$], otherwise if the Learning Objective is $C_3$ then the assembled e-learning experience will be composed as [LO$_1$;LO$_2$;LO$_3$].

---

The Learner Model

The Learner Model describes the main actor of the e-Learning process. Each learner is represented by a cognitive state and a set of learning preferences. The cognitive state is composed by a list of subjects (concepts within e-Learning ontologies) each with an associated grade that shows how much the student knows about that subject. The grade can range from 0 to 1, where 0 testifies the complete absence of acquired knowledge with respect to a given subject, whilst 1 represents a complete mastery of the subject. A subject will be considered as “learnt” by the learner if the above defined grade is greater than a fixed threshold, determined through experimentation. The learning preferences declare the properties that learning resources (learning objects or learning services) should have in order to fit with the learner’s characteristics.

They are expressed using a couple (propertyName, propertyValue). Examples of properties are Learning ResourceType, Interactivity Type, Interactivity Level, Typical Learning Time, Difficulty, Language, and Context.

The Unit of Learning construction process

In this section we provide details about the Unit of Learning construction process. In particular, we present the steps for the creation of an e-Learning experience tailored to a single user’s preferences. For the sake of simplicity we will consider that an e-Learning experience is represented by a sequence of learning objects, but the approach is also suitable when e-Learning experiences are made up of complex sets of learning activities.

In our approach, a learning object is a learning content (or a packaged aggregation of learning contents) that can be delivered through a Web Browser, and annotated with an instance of a metadata schema (interoperable with IEEE Learning Object Metadata), and stored and indexed into a Learning Object Repository.

In the Learning Model, an e-Learning experience is composed by:

- a set of Target Concepts (TCs) (known also as Learning Objectives),
- a Learning Path (LP)
- a Presentation (PR).

TCs are high level concepts that are the final goal of an e-Learning process. They can be set by a teacher or by the students themselves and can be obtained by manually selecting concepts on ontologies or by selecting pre-defined groups of concepts. Excluding the selection of TCs and other customization parameters, the building process is fully automatic and realized through the execution of several algorithms. The most important are Learning Path Generation Algorithm and Presentation Generation Algorithm constituting the aforementioned Computational Layer [2][3].

In the Learning Path Generation Algorithm the TCs are used to generate the LP, the ordered sequence of atomic concepts needed to reach a satisfactory level of knowledge about the selected TC. The right order of concepts is identified by taking into account the Learner’s cognitive state and all the dependencies between concepts described into the ontologies.

Following the example described in Figure 1, in order to understand the concept C a student has to learn concepts in this order: C1, C2, C3.

The Presentation Generation Algorithm creates the PR, an ordered list of Learning Objects that the learner has to use in order to acquire knowledge about subjects included in the LP. PR is created starting from LP and querying one or more LO Repositories to find the Learning Objects that have a HasResource relation with the concepts in the LP. The algorithm acts trying to minimize the number of learning objects within the Presentation that are necessary in order to cover the whole Learning Path. This problem can be formulated as a Plant Location Problem on a bipartite graph.

In the IWT implementation, the Plant Location Problem is solved with a Greedy Algorithm [5] that constructs the required set of learning objects step by step starting from an empty set. At each step, the algorithm selects among the not yet used learning objects the one that implies the maximum decrement for the sum of all distances of learning objects currently included in the set. The Greedy Algorithm is really quick, but its solutions are not very good because it cannot go backward and modify decisions taken in previous iterations. Let’s see a comprehensive example to understand the complete process.

Consider the situation of Figure 1 when IWT has to define a personalized e-Learning experience for a learner named Jane, whose TC is equal to C.

The Learning Path Generation Algorithm analyzes the structure of the domain ontology D (see Figure 1) and Jane’s cognitive state and learning preferences state (see Figure 2) and extracts the following personalized Learning Path:

$$LP(TC) = [C3, C2, C4]$$

Note that the subject C3 has already been “learnt” by Jane, so it has been deleted from the path. At this point the Presentation Generation Algorithm performs a binding between available learning objects and subjects in LP.

<table>
<thead>
<tr>
<th>Cognitive State</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1 (0.8)</td>
</tr>
<tr>
<td>C4 (0.3)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Learning Preferences State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language (EN)</td>
</tr>
<tr>
<td>Learning Resource Type (Narrative Text)</td>
</tr>
<tr>
<td>Context (Higher Education)</td>
</tr>
</tbody>
</table>

**Figure 2 - Jane’s cognitive state.**

In particular, the binding must minimize the number of learning objects and select the learning objects whose metadata (illustrated in Figure 3) satisfy better Jane’s learning preferences. In our example, PR = [LO2, LO4]. In particular, LO4 is preferred to LO3 because the second one doesn’t match with Jane’s learning preferences.
As a description of the case study we firstly introduce the reference e-Learning ontology for the Foundation of Informatics university course. Therefore, we discuss about the Learner Profiles and associated Learning Objects.

**e-Learning Reference Ontology**

The creation process of a reference ontology is often an error-prone and onerous activity [11]. One suitable way to minimize the cost of e-Learning ontologies is to reuse those already created. Our system supports reuse not only of learning resources but also of domain knowledge. An existing learning concepts can be a starting point for new perspectives or used for interchange. Indeed our IWT platform supports ontology reuse through matching and merging techniques. Together with the proposal of a common understanding of concepts through ontology matching algorithms, it allows the addition of new meaning to preexisting e-Learning concepts by means of ontology merging tools.

In the definition of the Foundation of Informatics ontology we have considered in five steps. Namely they are:

- **Vocabulary filling**: gathering of the information relevant to the particular learning domain with the identification of the specific terms. This step is usually performed by a domain expert (i.e. the teacher of the course) through the use of the vocabulary tool of our systems. This tool helps to create new terms and to locate existing terms that may be reused in the new context (avoiding renaming of the terms by means of semi-automatic matching techniques [8]). In our case we may introduce, for example, the concept Programming and reuse the existing terms Algorithm that has been used in other courses. The vocabulary is stored using the SKOS schema [14].

- **Hierarchization**: finding the relationship among the concepts and representing them in a hierarchical way. As an example, in this context, the concept Computer_Architecture may have a HasPart relation with the concept Von_Neumann_model in the Foundation of Informatics ontology.

- **Decomposition**: detailing coarse-grained concepts into a set of more fine-grained ones via top-down strategy. As an example, if we considers the concept Passing_parameters, we have the possibility to decompose it in the two related concepts Passing_by_reference and Passing_by_value.

- **Categorization**: grouping similar concepts together, in order to create “high-level” concepts to generalize the groups.

- **Refinement**: analyzing the created ontology in order to eliminate contradictions, synonymy and useless relations among the different concepts

All the steps can be supported by our tool, the IWT Ontology Editor which is a visual CASE tool that can be used for the definition of the e-Learning reference ontology of a course.

**Learner Profiles and Learning Objects**

An important element we must consider during the predisposition of a course is the final users of the e-Learning materials, i.e. the learners. In order to maximize the learning experience the course must be adaptable to the user’s preferences (e.g. way of studying, preferred materials, etc.). As described in Section 2 (e.g. in the Learner Model description), also in the preparation of Foundation of Informatics course, we can describe the general profile of learners, that will be used for the first interaction with the e-Learning objects.

The learning preferences defined in these steps are part of all the students’ profile and are now general. Indeed, the course is customized by means of the preferences and preferences change as a result of the behavior of the particular student during the course (e.g. making a test related to concepts explained by specific learning objects). In this way the system becomes adaptive not only on knowledge (the concepts learned and to learn) but also on the type of learning resources to be presented.

Attached to each course we can insert a number of Learning Objects of different types (plain text, HTML pages, PowerPoint slides, etc.). The preparation of the learning objects which we package in the wider concept of UoL (Unit of Learning).

Finally, each concept of the Foundation of Informatics course which, practically, represents a concept to be learned, may have a set of correlated metadata which explain the didactic context, the difficulty, the interactivity level, etc.

**4. EXPERIMENTAL RESULTS**

We experiment the overall approach during the first term of the current academic year in six different courses of the Electronic Engineering faculty. The fruition of the courses is conventional in the sense they include both classroom lectures and laboratory exercises.

IWT was used in addiction to the traditional learning activities and as a complimentary virtual classroom environment for
study and additional classes before the exams. In each course we support the teacher with a skilled IWT tutor as a help to manage the technical stuffs and to avoid student’s confusion.

We analyze the situation in the year before the introduction of our system, considering the results of the exams at the end of the term for each course. The collected data are summarized in Figure 4.

<table>
<thead>
<tr>
<th>Course</th>
<th>Students</th>
<th>Passed</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foundation of Informatics</td>
<td>186</td>
<td>96</td>
<td>51.6%</td>
</tr>
<tr>
<td>Informatics II</td>
<td>69</td>
<td>12</td>
<td>17.4%</td>
</tr>
<tr>
<td>Principles of Databases</td>
<td>125</td>
<td>57</td>
<td>45.6%</td>
</tr>
<tr>
<td>C Programming</td>
<td>68</td>
<td>17</td>
<td>25.0%</td>
</tr>
<tr>
<td>Computer Nets I</td>
<td>95</td>
<td>62</td>
<td>65.3%</td>
</tr>
<tr>
<td>Mathematics I</td>
<td>171</td>
<td>46</td>
<td>26.9%</td>
</tr>
<tr>
<td>Web Information Systems</td>
<td>38</td>
<td>30</td>
<td>78.9%</td>
</tr>
<tr>
<td>Software Engineering</td>
<td>135</td>
<td>90</td>
<td>66.7%</td>
</tr>
</tbody>
</table>

Figure 4 - Exams results before the introduction of IWT.

After this analysis, we experiment our platform during the last term, and we noticed more participation by students especially during the laboratory sessions in which the use of the platform were more massive. The collected data after the introduction of IWT is summarized in Figure 5.

It can be observed that we make our test on both the two exam sessions for each course, in order to have better statistics.

<table>
<thead>
<tr>
<th>Course</th>
<th>Students</th>
<th>Passed</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foundation of Informatics</td>
<td>152</td>
<td>97</td>
<td>63.8%</td>
</tr>
<tr>
<td>Informatics II</td>
<td>81</td>
<td>32</td>
<td>39.5%</td>
</tr>
<tr>
<td>Principles of Databases</td>
<td>110</td>
<td>60</td>
<td>54.5%</td>
</tr>
<tr>
<td>C Programming</td>
<td>81</td>
<td>30</td>
<td>37.0%</td>
</tr>
<tr>
<td>Computer Nets I</td>
<td>98</td>
<td>65</td>
<td>66.3%</td>
</tr>
<tr>
<td>Mathematics I</td>
<td>149</td>
<td>55</td>
<td>36.9%</td>
</tr>
<tr>
<td>Web Information Systems</td>
<td>52</td>
<td>49</td>
<td>94.2%</td>
</tr>
<tr>
<td>Software Engineering</td>
<td>104</td>
<td>89</td>
<td>85.6%</td>
</tr>
</tbody>
</table>

Figure 5 - Exams results after the introduction of IWT.

It can be observed that we make our test on both the two exam sessions for each course, in order to have better statistics.

In Figures 6, 7 and 8 a comparison between the courses with and without the introduction of our personalized e- Learning system is reported.

The average results for all courses are summarized in Figure 8.
Future works will concern the augmentation of the personalization approach through the analysis of other Semantic Web features (e.g. linked data for ontologies).

Moreover we are investigating some algorithms and techniques to extract concepts from knowledge bases of heterogeneous documents (e.g. plain text, PowerPoint slides, XML, etc.). This can lead to an improvement in the ontology creation that may also be very helpful for the preparation of personalized e-Learning courses.

Finally, we are testing the overall systems in wider scenarios, including more courses and faculties, to better understand both strong and weak points of the approach.

6. REFERENCES


