

Automatic Student Personalization in Preferred Learning Categories

Nicola Capuano^{1,2}

1 CRMPA, Centro di Ricerca in Matematica Pura ed Applicata, c/o Università degli Studi di Salerno – DIIMA, Via Ponte Don Melillo, 84084 Fisciano (SA), Italy.
capuano@crmpa.unisa.it

Matteo Gaeta^{1,2}

2 DIIMA, Dipartimento di Ingegneria dell'Informazione e Matematica Applicata, Università degli Studi di Salerno, Via Ponte Don Melillo, 84084 Fisciano (SA), Italy.
gaeta@crmpa.unisa.it

Alessandro Micarelli³

3 DIA, Dipartimento di Informatica e Automazione, Università degli Studi Roma Tre, Via della Vasca Navale, 79, 00146 Roma, Italy
micarel@dia.uniroma3.it

Enver Sanginetto⁴

4 CRMPA, Centro di Ricerca in Matematica Pura ed Applicata, Sezione "Roma Tre", c/o Università degli Studi Roma Tre – DIA, Via della Vasca Navale, 79, 00146 Roma, Italy.
sanginet@dia.uniroma3.it

Abstract

We present a computer-based system for the automatic generation and personalization of courses. The system takes trace of the student's behavior by analyzing the results of suitable on-line tests. The tests are used to infer information on both the student's knowledge about the topics of the system's domain and her/his learning preferences. Both kinds of data are stored in the Student Model and subsequently used in order to customize future courses. The pedagogical approach we have adopted refers to the Felder and Silverman's proposal in which both learning material and students are categorized in different teaching/learning style classes and then matched together.

1 Introduction

In this paper we present the Student Modeling aspects of a computer based learning system developed in the European founded project *Diogene* (Diogene 2001), ended in October 2004. The e-learning platform built during the project is able to automatically generate and personalize e-learning courses depending on the student's knowledge and preferences. The platform's learning material is composed of a set of Web deliverable objects (textual or hyper-textual documents, interactive exercises, simulations and others), called *Learning Objects* (LOs). The LOs are mostly provided by professional teaching centers and stored in their own servers (the *content providers' servers*). Furthermore, each system's LO is associated with one or more abstract topics (called *Domain Concepts* or DCs) describing the domain of interest of the system. An example of didactic domain could be "Computer Science", whose Domain Concepts (with different granularity levels) are: "Operative Systems", "Programming Languages", "Object Oriented Programming Languages", "Class Definition", "Private Variables", and so on.

The student submits a query to the platform specifying a set of arguments (i.e. Domain Concepts) she/he is interested in (e.g.: "Method Call", "Private Variables"). First of all the platform selects all the DCs necessary to the specific student in order to learn the concepts contained in the query and generates a *Learning Path*. This is done both using a description of the didactic relations among the DCs (represented by the system's *Ontology*) such as prerequisite and other relations,

and using a *Student Model* which represents the current student's knowledge about each DC. An example of Learning Path can be: "Class Definition", "Method Definition" "Method Call", "Private Variables". Finally, the system "translates" the Learning Path in a course (called *Presentation*), i.e. associates to each DC of the Learning Path a LO which explains it. The LOs of the Presentation are selected among all the available LOs of the system's repository by taking into account the student's *Learning Preferences*, described in the Student Model too.

The system's Student Model is continuously updated by the results of special interactive LOs of type *test* which are included in the Presentations and delivered to the student. The main objective of the LO customization is to choose those LOs which best fit the student's *Learning Styles*, following a pedagogical approach suggested by Felder and Silverman (Felder and Silverman 1999) (see Section 2). The system's knowledge and its inference mechanisms allow the platform to automatically customize the Presentation on a given student and modify the creation of future courses taking into account the student's feedback provided by answering the tests. For this reason we have an intelligent and adaptive platform which actively participates in the learning process, as opposed to common e-learning platforms usually playing only the role of passive LOs' containers. For instance, the well known Ariadne platform (Duval 2001) is based on a digital library of LOs which are indexed by using educational metadata standards (as in our project). Nevertheless, the Ariadne platform only aims at building a library of reusable learning components for the sharing of such components among different (human) teachers. The library is then a passive repository and there is no automatic building of courses nor any adaptation on the user profile.

This situation is very common in the new generation of Web-based learning platforms, specially in the commercial systems (e.g., (Blackboard, 1997) (Lotus 2004)). Indeed, they are born to support modern distance learning (e.g., in long-distance university courses) and usually only work as passive content containers, in which teachers and students can exchange documents and information. In (Seal 2001) the authors propose a system able to collect students' feedback during the whole duration of the course in order to give the teacher the possibility to adapt the course itself on classroom needs. Nevertheless, no automatic adaptation is expected and the feedback is analyzed by hand.

On the other hand, what we describe in this paper and we have realized in the Diogene project is a platform able not only to deliver to the students learning material taken from a (indexed) LO repository, but a system which can automatically customize the courses monitoring both the student's acquired knowledge about the platform's topics and her/his learning preferences.

The student's knowledge degree about the platform's DCs is updated by the results of the tests, which are also analyzed in order to improve the categorization of the student's learning styles. The main idea concerning this latter derives from the pedagogical studies of Felder and Silverman (Felder and Silverman 1988), who categorize the students in different classes of *learning styles* and the teaching activities in corresponding classes of *teaching styles*. Learning styles and teaching styles are finally matched in order to improve the overall teaching efficiency.

The Felder and Silverman's pedagogical theory is briefly introduced in Section 2. In Section 3 we give an overview of the whole system showing its knowledge representation structures and the course generation/personalization main features. In Section 4 we present the Student Modeling features of the system used in the course generation/personalization process and which refer to the Felder and Silverman's pedagogical approach. In Section 5 we show some preliminary experimental results. Finally, we conclude in Section 6.

2 The Felder and Silverman's Categories of Learners

In (Felder and Silverman 1988) the authors propose five couples of student's categories (which will later become four couples). These categories derive both from previous psychological studies

(such as the Jung's well-known theory of psychological types) and from some author's empirical observations. The most important observation is that each person usually shows her/his preference for one or more modalities in the way she/he receives and elaborates information. For example, one can prefer to receive information by means of either the visual sensory channel or the auditory sensory channel. In the former case, her/his learning process is better suited for visual material (e.g., pictures), in the latter for sounds or texts (which are composed of words, then strictly related to the auditory information processing).

For this reason, Felder and Silverman propose to categorize each learner in different learning categories. Each category is characterized by two opposite attributes (e.g. Visual versus Verbal) which represent the extremes of the range of possibilities for that category (a student can be more or less visual or verbal). Moreover, for each learning category, there is a corresponding teaching category, which indicates the type of teaching most suited for the corresponding preferred way to receive-process information during learning. The Felder and Silverman's main four categories are the following:

- *Sensing versus Intuitive Learner.* It represents the abstraction level of the learning material the student prefers. A Sensing student tends to like learning facts, she/he likes to use the same methods repeatedly. The student will need more practical case studies. An Intuitive student often prefers discovering possibilities and relationships. He likes innovation and dislikes repetition and too much memorization. The student is more comfortable with abstractions.
- *Visual versus Verbal Learner.* It indicates whether the student prefers auditory (textual) or visual documents. A Visual student remembers best what he sees: pictures, diagrams, flow charts, movies, demonstrations, etc. A Verbal student gets more out of words, written and spoken explanations. She/he has to write summaries or outlines of course material in his own words. Working in groups (through discussion groups, chat or teleconference) can be effective. She/he gains understanding of material by hearing co-students explanations and she/he also learns more when she/he does the explaining.
- *Active versus Reflective.* It indicates how the student prefers to process information: actively (through engagement in activities or discussions) or reflectively (through introspection).
- *Sequential versus Global.* It indicates how the student progresses toward understanding. Sequentially: in continual steps and globally: in large jumps, holistically. Sequential students prefer sequential explanations while global students usually prefer an initial overview of the involved topics which possibly shows them the most important steps and relations they are going to study.

The first two categories regard the way in which we prefer to receive information during the learning process while the other two relations regard the way in which we prefer to elaborate it. In its initial proposal, Felder and Silverman also have included a fifth category (Inductive versus Deductive learning/teaching) which has been subsequently discarded.

It is worth noting that the 4 couples of categories are not mutually excluding. On the contrary, each learner can be classified using a combination of values, one for each of the 4 categories. For instance, a given learner can be very intuitive, quite visual, strongly reflexive and indifferent with respect to the sequential-global choice.

We have adopted the Felder and Silverman's categories for two reasons. First of all, their approach is based on a sufficiently large experimentation which has validated the proposed classes on an engineering student population. Second, although other approaches are perhaps based on a stronger cognitive model formalization, the Felder and Silverman's theory provides some useful pragmatic instruments to customize teaching depending on the student's profile.

Of course, our proposed automatic course generation platform cannot take into account all the teaching suggestions proposed by the two authors. Indeed, some of these suggestions have been thought for human made frontal lessons. Nevertheless, if we are able to categorize each student in her/his learning categories and associate (off-line) the most suitable teaching categories with each LO of the system's data base (see Section 4), then we have an efficient and effective way to select the most appropriate learning material for each given student.

Finally, Felder and Soloman proposed a psychological test (Felder and Soloman, 1993) for learners' categorization which is also (optionally) provided to the Diogene's students in their first approach to the system (see Section 4).

3 The Course Generation and Personalization Framework

The Diogene platform is able to act as an intermediary between the learners and different "content providers", specialized training organizations which provide learning material suitably indexed by means of common Semantic Web standards and following a knowledge representation paradigm developed during the project (Capuano et al. 2002), (Capuano et al. 2003). The knowledge representation framework allows the system to efficiently assembly and personalize the learning material in order to answer to the student's training requests.

The knowledge representation framework the system is based on is composed of three different sub-structures, respectively used for the representation of the student (the *Student Model*), the Learning Objects (indexed by means of *Metadata* described with the well-known e-learning standard: "IMS Metadata Standard" (IMS, 2001)) and the *Ontology* which describes the pedagogic relationships among the DCs. When the learner submits a training query (a set of DCs she/he is interested in) to the system, Diogene generates a course by choosing and assembling the available Learning Objects and taking into account the didactic information cabled in the Ontology as well as the student's profile.

The pedagogical information of the Ontology includes prerequisite, decomposition and ordering relations among DCs, which altogether allow the system to generate a set of DCs (called *Learning Path*) starting from the DCs contained in the student's query and including and suitably ordering all the necessary other DCs. The Student Model is used in order to remove, from the *Learning Path*, those DCs already known by the student. From now on, if not otherwise explicitly indicated, we mean with "Learning Path" the set of ordered DCs after this pruning operation.

Finally, we select a set of LOs explaining all the concepts of the *Learning Path* by choosing those LOs which best fit the student's learning styles as described in the Student Model.

We refer to (Capuano et al. 2002), (Capuano et al. 2003) for more details on the course generation procedure and on the didactic knowledge representation. In the rest of this article we focus on the Student Modeling aspects of the system used for the personalization purposes.

4 The Student Model

The Diogene's Student Model is composed of two modules: the *Cognitive State* and the *Learning Preferences*. We use the former to describe the knowledge degree achieved by each student about every DC of the Ontology. This evaluation regards both previously acquired student knowledge and skills learnt using the Diogene's platform. On the other hand, the Learning Preferences module represents information about the student's learning styles (see Section 2) and other student's data which have to be taken into account during the LO selection process (see Section 3). The Cognitive State is represented by the set $CS = \{B_1, B_2, \dots, B_n\}$, each B_i being a triple of the type:

$$B_i = \langle d_i, e_i, N_i \rangle, \quad (1)$$

where d_i is the i -th Ontology's Domain Concept, e_i is a fuzzy value ($e_i \in [0, 1]$) representing the current knowledge degree reached by the student about the argument d_i and N_i is the number of test evaluation performed so far by the platform on the concept d_i (see Section 4.1). For instance, the triple: \langle "Private Variables", 0.6, 2 \rangle means that the student has shown a sufficient degree of knowledge/comprehension on the concept "Private Variables" and this has been tested twice by the system.

When a given student first interacts with the platform, for every DC d of the Ontology, the system includes in the student's Cognitive State the triple $\langle d, 0, 0 \rangle$, thus making the assumption that d is completely unknown by the learner. This evaluation is maintained until a test on the concept d is evaluated: the result of the test will update the system's estimation on the student knowledge about d (see Section 4.1).

In order to decide if the concept d is known to the student and has to be removed from the Learning Path (Section 3), we use a fixed threshold K (e.g., $K = 0.5$): every DC d , such that $\langle d, e, N \rangle \in CS$ and $e \leq K$, is removed from the Learning Path during the course generation process.

The Learning Preferences (LP) are represented by the set $LP = \{P_1, P_2, P_3, P_4\}$, where:

$$\begin{aligned} P_1 &= \langle \text{"Educational Context"}, v_1 \rangle, P_2 = \langle \text{"Age"}, v_2 \rangle, P_3 = \langle \text{"Languages"}, v_3 \rangle, \\ P_4 &= \langle \text{"Learning Styles"}, Categories \rangle. \end{aligned} \quad (2)$$

The Property P_1 indicates the educational context of the learner, as proposed by the IMS Metadata Standard (IMS, 2001). Indeed, even if the standard is thought to be used for the LO representation, it is also used by Diogene as a reference for the representation of some student's attributes that are matched by the system with the LOs' Metadata in the course generation/personalization process. The values of v_1 range in the set of admissible values selected by the standard (i.e., strings such as: "University", "Secondary School", "Continuous Formation", and others). Similarly, the Properties P_2 and P_3 indicate, respectively, the age and the main languages of the learner. The values of v_2 are integers while v_3 is a set of at most 3 strings chosen among the IMS vocabulary (e.g.: "English", "Italian", "French", and so on (IMS, 2001)). Finally, Property P_4 describes the learning styles preferred by the student following the Felder and Silverman's categorization (see Section 2) and is represented by means of the 4-ple:

$$Categories = \langle C_1, C_2, C_3, C_4 \rangle, \quad (3)$$

each C_i ($1 \leq i \leq 4$) being in its turn a couple of the type: $C_i = \langle T_i, e_i \rangle$, where T_i represents one of the 4 possible types of learning style categories ($T_i \in \{\text{"Sensing-Intuitive"}, \text{"Visual-Verbal"}, \text{"Active-Reflective"}, \text{"Sequential-Global"}\}$) and e_i is a fuzzy value ($e_i \in [0, 1]$) representing the current system's estimation of the student's preference with respect to the category T_i . For instance, a student whose Learning Styles are given by:

$$Categories = \langle \langle \text{"Sensing-Intuitive"}, 0.7 \rangle, \langle \text{"Visual-Verbal"}, 0.3 \rangle, \langle \text{"Active-Reflective"}, 0.6 \rangle, \langle \text{"Sequential-Global"}, 0.9 \rangle \rangle \quad (4)$$

is more Intuitive than Sensing (0.7 against 0.3), more Visual than Verbal (the opposite with respect to the previous one: 0.3 against 0.7), a bit more Reflective than Active and very much oriented to a Global processing of information rather than a Sequential one (0.9 against 0.1).

The first time a student approaches to the Diogene's platform, the system sets the values of the Properties P_1 - P_3 of the Learning Preferences directly asking the necessary information to the student. Concerning the Learning Styles (property P_4), the platform offers to the learner the optional possibility to use the Solomon and Felder's psychological test (Felder and Soloman,

1993) for her/his on-line classification. The results of this test is used to instantiate the values of the Learning Styles. The use of the Soloman-Felder test is not mandatory for the learner in order to avoid the user a heavy impact with the system. In the case in which the student prefers not to use this preliminary psychological test, the initial values of all the category types are set to 0.5:

$$\text{Categories} = \langle\langle\text{"Sensing-Intuitive"}, 0.5\rangle, \langle\text{"Visual-Verbal"}, 0.5\rangle, \langle\text{"Active-Reflective"}, 0.5\rangle, \langle\text{"Sequential-Global"}, 0.5\rangle\rangle, \quad (5)$$

which means that the student is (initially evaluated as) indifferent with respect to any learning style preference. In the following subsection we show how the values of both the Cognitive State and the Learning Styles are automatically updated by the system taking into account the results of the (didactic) tests included in each Presentation.

Before concluding this section, we show how the system performs the matching among the LOs and the student's Learning Preferences. In fact, each LO is off-line (manually) indexed by means of a Metadata using the IMS Metadata Standard (IMS, 2001) when it is initially stored in the content provider's repository (see Section 3). A Metadata is a set of attributes describing information concerning the LO. This information is organized in several categories and it concerns both content-dependent and content-independent (e.g., copyright issues, title, version and so on) data. Diogene uses the "Classification" and the "Educational" (content-dependent) categories of the standard in order to represent and use information concerning the LO. For example, we use the attribute "TaxonPath" contained in the category "Classification" to link the LO with one or more DCs of the Ontology. At run-time, once the Learning Path has been generated and the known concepts deleted (see Section 3), for each remaining DC d of the Learning Path the system selects a LO among all those LOs linked to d by means of the TaxonPath attribute.

The choice of the best LO for d is performed by the system by matching the attributes of the "Educational" category of the LOs' Metadata with the student's Learning Preferences. The attributes of the "Educational" category of the IMS Metadata Standard we use are: "Learning Resource Type", "Context", "Typical Age Range" and "Language". The last three attributes are matched with the corresponding properties of the student's Learning Preferences: the system will exclude those LOs whose "Context" attribute is different from the property "Educational Context" contained in the Learning Preferences of the student (Property P_1). In a similar manner, LOs whose "Typical Age Range" do not include the value of the "Age" property (Property P_2) of the student or LOs whose "Language" attribute is not included in the set of values of the "Languages" property (Property P_3) of the student are excluded from the set of possible choices for d .

Finally, if $A(d)$ is the set of the admissible LOs for d , i.e. those LOs which are linked to d by the TaxonPath attribute and which are consistent with the properties P_1 - P_3 of the student's Learning Preferences, the system selects among $A(d)$ a LO taking into account the value of the attribute "Learning Resource Type" of its Metadata. This is done as follows. The values of the "Learning Resource Type" attribute of the IMS Metadata Standard are strings ranging in the set: {"Exercise", "Simulation", "Questionnaire", "Diagram", "Figure", "Graph", "Index", "Slide", "Table", "Narrative Text", "Exam", "Experiment", "ProblemStatement", "SelfAssesment"}. With the help of pedagogic experts we have classified all these types of resources using the Felder and Silverman's teaching styles, by associating to each resource type (R) a 4-ple ($Teaching\ Styles(R)$) of the same type of Equation 3. For instance, "Exercise" and "Simulation" are respectively classified as:

$$\text{Teaching Styles(Exercise)} = \langle\langle\text{"Sensing-Intuitive"}, 0\rangle, \langle\text{"Visual-Verbal"}, 0.5\rangle, \langle\text{"Active-Reflective"}, 0\rangle, \langle\text{"Sequential-Global"}, 0.5\rangle\rangle, \quad (6)$$

$$\text{Teaching Styles(Simulation)} = \langle\langle\text{"Sensing-Intuitive"}, 0.3\rangle, \langle\text{"Visual-Verbal"}, 0\rangle, \langle\text{"Active-Reflective"}, 0\rangle, \langle\text{"Sequential-Global"}, 0.5\rangle\rangle. \quad (7)$$

At run-time, if l is a LO belonging to $A(d)$ (for a given DC d), and $R = Resource(l)$ is the resource type of l as described in the “Learning Resource Type” field of its Metadata, the system computes the matching between l and the student’s learning styles as follows. Suppose the student’s learning styles are represented by:

$$Categories = \langle \langle T_1, e_1 \rangle, \langle T_2, e_2 \rangle, \langle T_3, e_3 \rangle, \langle T_4, e_4 \rangle \rangle \quad (8)$$

and the teaching styles associated with the resource type are represented by:

$$Teaching\ Styles(R) = \langle \langle T_1, f_1 \rangle, \langle T_2, f_2 \rangle, \langle T_3, f_3 \rangle, \langle T_4, f_4 \rangle \rangle, \quad (9)$$

then we can compute the distance between l and the student’s profile by means of the following:

$$Dist(R) = (e_1 - f_1)^2 + (e_2 - f_2)^2 + (e_3 - f_3)^2 + (e_4 - f_4)^2. \quad (10)$$

The Presentation is built by choosing a set of LOs which minimizes the sum of $Dist(R_j)$ for all the resource R_j of the set.

4.1 Student Model Updating

In each Presentation delivered to the student the system includes some special LOs of type *test* which consist of interactive multiple-choice tests evaluated by Diogene in order to assess the knowledge degree reached by the learner concerning the DCs of the Learning Path. All the tests are attached at the end of the Presentation, which is then split in two consecutive sequences (delivered to the student respecting this order): a learning phase, composed of normal LOs, and a test phase, composed of tests. Suppose that:

$$Learning\ Path = \{d_1, \dots, d_n\} \text{ and:} \quad (11)$$

$$Presentation = \{l_1, \dots, l_m, t_1, \dots, t_{m'}\}, \quad (12)$$

where $t_1, \dots, t_{m'}$ are the LOs of type test representing the test phase. Note that the neither the number of normal LOs (m) nor the number of test LOs (m') is requested to be equal to the number of DCs (n). In fact, a given LO (normal or test) can be used to explain/evaluate more than one DC, depending on the TaxonPath relation with the elements of the Ontology. On the contrary, a given DC of the Learning Path is associated with one and only one LO of the Presentation (see (Capuano et al. 2002), (Capuano et al. 2003) for more details). Finally, the course generation/personalization procedure guarantees that, altogether, l_1, \dots, l_m cover the arguments d_1, \dots, d_n and $t_1, \dots, t_{m'}$ test the same topics d_1, \dots, d_n . Furthermore, suppose that the result of the whole test phase gives the following set of fuzzy values: $\{v_1, \dots, v_n\}$, v_i ($1 \leq i \leq n$) being the degree of knowledge shown by the student for the concept d_i in the just evaluated test phase. Finally, if $\langle d_i, e_i, N_i \rangle \in CS$ (where CS is the student’s Cognitive State before all the tests’ evaluation), then CS is updated by substituting $\langle d_i, e_i, N_i \rangle$ with $\langle d_i, e'_i, N'_i \rangle$, where:

$$N'_i = N_i + 1, \quad (13)$$

$$e'_i = (e_i N_i + v_i) / N'_i. \quad (14)$$

Equation 14 updates the system’s evaluation about the student’s knowledge on d_i by computing the average value of all the test phases’ results (included the last v_i) performed so far.

Concerning the Learning Preferences of the Student Model, there is no modification of the initial values of the properties P_1 - P_3 , while the student's Learning Styles (property P_4) are updated as follows. For each type of learning style T_k ($1 \leq k \leq 4$, see Section 4) the system stores in the student's Learning Preferences an auxiliary data structure composed of an histogram $H[T_k]$ which represents information related to the (discretized) statistic distribution of the preferences so far shown by the student concerning T_k . For example, $H["Sensing-Intuitive"]$ is given by an histogram of 10 bins (a discretization of the range $[0, 1]$ in 10 steps) which intuitively describes the statistic data collected by the system on the preferences shown by the student for Sensing and Intuitive resources. $H[T_k]$ is represented by the following function:

$$H[T_k]: \{1, 2, \dots, 10\} \rightarrow \mathbb{R} \quad (15)$$

The highest bin of $H["Sensing-Intuitive"]$ represents the preference shown by the student for the Sensing-Intuitive choice of the learning resources. For instance, in Figure 1 the bin with the maximum value of $H["Sensing-Intuitive"]$ is the third, which corresponds to the couple $\langle "Sensing-Intuitive", 0.3 \rangle$ in the *Categories* representation in the Student Model. We have that, for each $\langle T_k, g_k \rangle \in \text{Categories}$ ($1 \leq k \leq 4$):

$$g_k = (\text{argmax}_{h \in \{1, 2, \dots, 10\}} H[T_k](h)) / 10. \quad (16)$$

We initialize the histograms by setting:

$$H[T_k](h') = M, \quad (17)$$

$$H[T_k](h) = 0, \text{ for all } h \in \{1, 2, \dots, 10\}, h \neq h', \quad (18)$$

where M is a (small) pre-fixed value and h' is the initial preference of the student, either obtained by means of the Solomon-Felder test or a default value (in this case $h' = 5$: see Section 4).

The histograms are used to take trace of the student's behavior related to the different learning resources used in the Presentation. For every LO l_j ($1 \leq j \leq m$) of the Presentation (Equation 12) and every learning category we update the histogram bins corresponding to the resource type of l_j and, then, we take the (possible new) maximum bins of the new histograms as the representation of the student's learning styles. More in details, suppose that:

$$\text{Categories} = \langle \langle T_1, g_1 \rangle, \langle T_2, g_2 \rangle, \langle T_3, g_3 \rangle, \langle T_4, g_4 \rangle \rangle \quad (19)$$

is the classification of the student's learning styles before the last test phase and, for each $d_i \in \text{Learning Path}$, $w_i = e'_i - e_i$ is the difference between the evaluation of the DC d_i before and after the test phase as abovementioned. Moreover, for each LO l_j contained in the Presentation, suppose that l_j has been used to explain to the student the set $D_j = \{d_{j1}, \dots, d_{jz}\}$ (this information is stored in the *TaxonPath* attribute of l_j). Hence, we can compute the overall benefit of l_j with respect to D_j as:

$$W_j = \sum_{s=1}^z w_{j_s} \quad (20)$$

Finally, suppose that $R_j = \text{Resource}(l_j)$ and:

$$\text{Teaching Styles}(R_j) = \langle \langle T_1, f_1 \rangle, \langle T_2, f_2 \rangle, \langle T_3, f_3 \rangle, \langle T_4, f_4 \rangle \rangle. \quad (21)$$

For each l_j and for each category T_k ($1 \leq k \leq 4$) we increment the h -th bin of $H[T_k]$, where $h = \text{Int}(f_k * 10)$, being $\text{Int}(x)$ the function which returns the integer best approximating the real value x . The increment is obtained by substituting $H[T_k]$ with the new $H'[T_k]$ defined as:

$$H'[T_k](h) = H[T_k](h) + W_j. \quad (22)$$

$$H'[T_k](l) = H[T_k](l), \text{ for all } l \in \{1, 2, \dots, 10\}, l \neq h. \quad (23)$$

Note that W_j can possibly be a negative value (even if this is a seldom event). For this reason $H[T_k]$ is defined in \mathcal{R} (see Equation 15) and not in \mathcal{R}^+ .

When the updating operation has been repeated for all the LOs of the Presentation obtaining the final histograms H' , the system substitutes the old *Categories* with the new:

$$\text{Categories}' = \langle \langle T_1, g'_1 \rangle, \langle T_2, g'_2 \rangle, \langle T_3, g'_3 \rangle, \langle T_4, g'_4 \rangle \rangle, \text{ where:} \quad (24)$$

$$g'_k = (\text{argmax}_{h \in \{1, 2, \dots, 10\}} H'[T_k](h)) / 10 \quad (1 \leq k \leq 4). \quad (25)$$

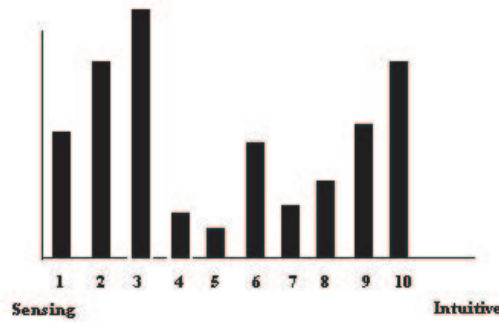


Figure 1: An example of Sensing-Intuitive Histogram

5 Experimental Results

The Diogene's system has been evaluated by a set of 137 learners, belonging to 6 of the 8 different companies and University departments, partners of the project. The evaluation method has been based on on-line questionnaires and interviews of the involved learners as well as the analysis of the system's reports and log information. Figures 2 summarizes the results of the interviews.

About 70 % of the learners felt that their user profile provided good support to define their learning behaviors. They showed highly satisfied with the allocation of courses they achieved by use of their profile definitions (72 %) and the adequacy of the courses that were suggested by the system to their lack of knowledge (78 %). More details are available in (Diogene 2004).

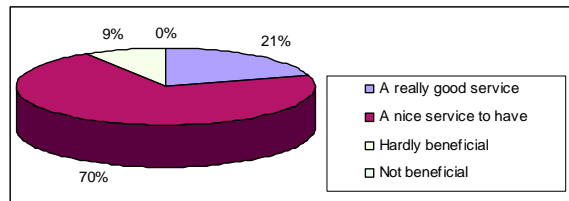


Figure 2: The result of the interviews. The question was: "Diogene is...?"

6 Conclusions

In this paper we have presented a computer based learning system able to automatically generate and personalize complex courses using the system's knowledge on: (1) the student's current skills and preferences and (2) the didactic domain of interest. We have focused on the system's Student Modeling features which contain a representation of the student's knowledge and her/his learning preferences. This latter are mainly based on a pedagogical strategy developed at the end of the '80 years and developed in our system by matching the (previously classified) Learning Objects with the student's profile.

The scientific work presented in this paper is the result of the European Community founded project Diogene (5th Framework Programme, Information Society Technologies, contact number IST-2001-33358, see (Diogene, 2001) for more details), finished in October 2004.

7 References

Blackboard Web Site (1997): <http://www.Blackboard.com/>.

Capuano N., Gaeta M., Micarelli A. and Sangineto E. (2002). An integrated Architecture for Automatic Course Generation. *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT 02)*, Kazan, Russia, pages 322-326.

Capuano N., Gaeta M., Micarelli A. and Sangineto E. (2003). An Intelligent Web Teacher System for Learning Personalization and Semantic Web Compatibility, *Proceedings of the Eleventh International PEG Conference*, St. Petersburg, Russia, 28 June-1 July.

Diogene (2001). A Training Web Broker for ICT Professionals (IST-2001-33358), official Web site: <http://www.diogene.org>.

Diogene (2004). The project's Final Report, available at: <http://www.diogene.org>.

E. Duval et al. (2001). The Ariadne knowledge pool system, *Communication of the ACM* 44 (5), 73-78.

Felder R. M. and L. K. Silverman (1988). Learning and Teaching Styles in Engineering Education, *Engr. Education* 78 (7), 674-681.

Felder and Soloman (1993). The Learning Styles Questionnaire, available at: <http://www.engr.ncsu.edu/learningstyles/ilsweb.html>

IMS (2001). Learning Resource Meta-Data Best Practice and Implementation Guide, 2001, version 1.2.1, Final Specification (<http://www.imsproject.org/specifications.html>).

Lotus LearningSpace Web Site (2004): <http://www.lotus.com/learningspace>

K. C. Seal, Z. H. Przasnyski (2001). Using the World Wide Web for teaching improvement, *Computers and Educations* 36, 33-44.