

# Towards a multidimensional model for learning objects and learners

Anis Elferchichi, Noura Ben Alaya, Mohamed Ben Ahmed

► **To cite this version:**

Anis Elferchichi, Noura Ben Alaya, Mohamed Ben Ahmed. Towards a multidimensional model for learning objects and learners. Michael E. Auer. Conference ICL2007, September 26 -28, 2007, 2007, Villach, Austria. Kassel University Press, 13 p., 2007. <hal-00197240>

**HAL Id: hal-00197240**

**<https://telearn.archives-ouvertes.fr/hal-00197240>**

Submitted on 14 Dec 2007

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Towards a multidimensional model for learning objects and learners

Anis Elferchichi<sup>1</sup>, Noura Ben Alaya<sup>2</sup>, Mohamed Ben Ahmed<sup>3</sup>

National School of Computer Science, University of Manouba, Tunisia

**Key words:** *Learning styles, Adaptive learning, learning objects, Aspect oriented programming, Aspect oriented LOs, Aspect oriented learning concept, AC-Genesys.*

## Abstract:

*This paper presents a multi-dimensional learning style model and a system specification to automatically generate adaptive courses using the aspect oriented technique.*

## 1 Introduction

The challenge regarding the application of learning styles in e-learning environments is taking in account, not only the cognitive, affective and social aspects of each learner, but also, the dynamic variation of these aspects for the same learner during a learning activity.

Numerous researches have formed the basis for the development of a number of personalized learning theories and models, based on cognitive, psychological, sociological, and cultural aspects of the learner. Each theory proposes a learning style from a particular point of view and into a particular context. It goes for the term "Learning style" as well, which has no one definition. We define a learning style as the way people construct and co-construct their knowledge and skills.

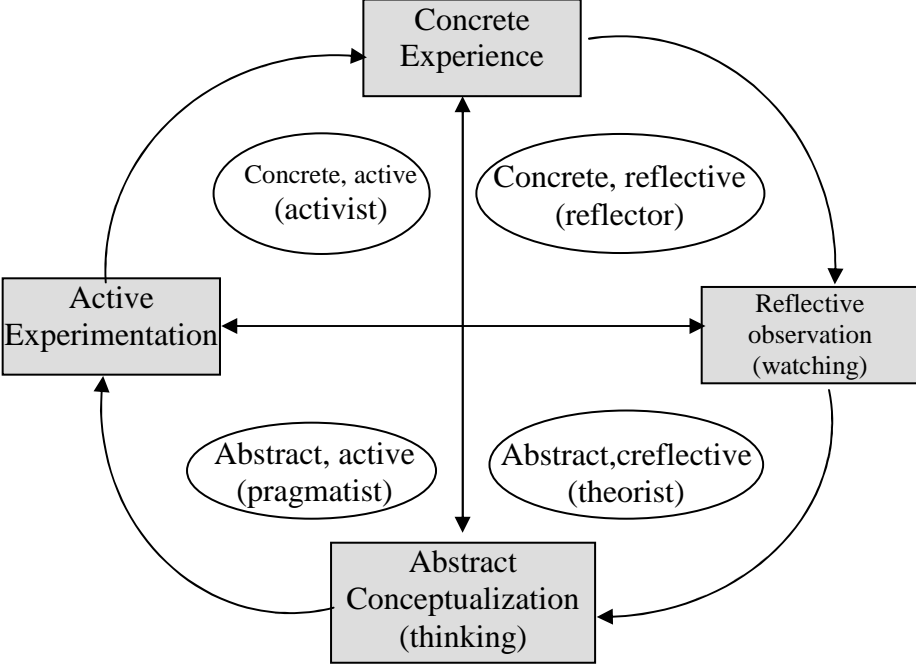
We outline in the beginning of this paper the most well-known learning styles; then, we present a semi-formal design of a system, called AC-Genesys, taking as input the coordinates of a given learner for a wanted material, enabling to figuring out a path crossing the learner objects stored in LOs repository (based on their metadata), and eventually generating personalized course. This generation is driven by a dynamic mediator between the learner profile and the appropriate learning objects. Finally, we present a multi-dimensional model considering, among others, the learner's cognitive aspect, socio-affective aspect, and the material nature.

## 2 Learning Styles

Learning styles is generally defined as a model that classifies students according to where they fit on a number of scales pertaining to the ways they receive and process information [1]. J.W.Keefe describes learning styles as the characteristic cognitive, affective, and psychological behaviours that serve as relatively stable indicators of how learners perceive, interact with, and respond to the learning environment [2].

The following table summarises the most well-known learning styles and their effects on learning based on [1; 3; 4; 5; 6; 7; 8; 9; 10; 11]:

	<b>Factors</b>
<p><b>Dunn and Dunn</b> Learning styles questionnaire/Inventory 1979; 1993</p>	<p>Based on 5 different categories:</p> <ul style="list-style-type: none"> <li>• Environmental factors (sound/noise level, light level, design setting, temperature)</li> <li>• Sociological factors (self/pair/team/authority orientation)</li> <li>• Emotional factors (motivation, persistence, responsibility, structure)</li> <li>• Physiological factors</li> <li>• Psychological factors</li> </ul>
<p><b>Entwistle</b> Approaches and study skills inventory for students ASSIST:1981; 1997</p>	<p>Learning is classified into two main categories:</p> <ul style="list-style-type: none"> <li>• Deep learning: study with the ultimate intention of understanding the subject and integrate the new material with their prior knowledge</li> <li>• Surface learning: seek to reproduce the course material</li> </ul>
<p><b>Felder-Silverman</b> Instrument: Index of learning styles 1988; 2002</p>	<p>Students are classified as:</p> <ul style="list-style-type: none"> <li>• Active (learn by experimentation; working with others) / reflective (think on their own)</li> <li>• Sensing(concrete, practical, oriented toward facts and procedures) / intuitive(conceptual, innovative, oriented toward theories and meanings)</li> <li>• Visual (pictures; graphs; charts) / verbal (written or spoken explanation)</li> <li>• Sequential (incremental steps) / global (accumulate all the facts)</li> </ul>
<p><b>Herrmann</b> Brain Dominance instrument 1986</p>	<p>This method classifies students in terms of their relative preferences for thinking in four different modes based on the task-specialized functioning of the physical brain. the four modes or quadrants in this classification scheme are:</p> <ul style="list-style-type: none"> <li>• Quadrant A (left brain, cerebral): analytical, logical, factual, critical and quantitative</li> <li>• Quadrant B (left brain, limbic): sequential, structured, organized, planned, conservative and detailed</li> <li>• Quadrant C (right brain, limbic): interpersonal, emotional, sensory, kinaesthetic, symbolic and spiritual</li> <li>• Quadrant D (right brain, cerebral): visual, holistic, innovative, conceptual, imaginative, artistic</li> </ul>
<p><b>Honey and Mumford</b> Learning styles questionnaire LSQ:1982</p>	<p>Four main learning styles preferences are identified:</p> <ul style="list-style-type: none"> <li>• Activists: accommodators: open minded, like to be involved in new experiences</li> <li>• Reflectors: divergers: like to collect data and think about it carefully before coming to any conclusions.</li> <li>• Theorists: assimilators: adapt and integrate observations into complex and logically sound theories, think step by step</li> <li>• Pragmatists: convergers: keen to try things out, want concepts linked to their job</li> </ul>
<p><b>Jackson</b> Learning styles profiler LSP:2002</p>	<ul style="list-style-type: none"> <li>• Initiator</li> <li>• Reasoner</li> <li>• Analyst</li> <li>• Implementer</li> </ul>

<p><b>Kolb</b> Learning styles inventory 1970s</p>	 <p><i>Kolb's learning style model</i></p> <p>Kolb defined a 2-dimensional scale to represent learning styles, which leads to 4 extreme cases:</p> <ul style="list-style-type: none"> <li>• Pragmatist(or Converger): abstract/active</li> <li>• Reflector(or Diverger): concrete/reflective</li> <li>• Theorist(or Assimilator): abstract/reflective</li> <li>• Activist(or Accommodator): concrete/active</li> </ul>
<p><b>Myers-Briggs</b> Myers-Briggs Type Indicator MBTI:1962-1985-1998</p>	<p>Students are categorised according to their position on scales based on Jung's theory[12] of psychological types. The types being:</p> <ul style="list-style-type: none"> <li>• Perceiving (work spontaneously) / judging (prefer rigid structure and planning)</li> <li>• Sensing (prefer details) / intuition (prefer abstract concepts)</li> <li>• Thinking (strict logic, impartial) / feeling (decisions are based on social consideration)</li> <li>• Extraversion (thrive in group setting) / introversion (spend time alone)</li> </ul>
<p><b>Riding</b> Cognitive style analysis CSA:1991-1998</p>	<ul style="list-style-type: none"> <li>• holist-analytic: organize information into wholes or parts</li> <li>• Verbaliser-imager: represent information during thinking verbally or in mental pictures</li> </ul>
<p><b>Sternberg</b> Thinking styles</p>	<ul style="list-style-type: none"> <li>• Thirteen thinking styles divided into three functions, four forms, two levels, two scopes and two leanings</li> </ul>
<p><b>Vermunt</b> Inventory of learning styles 1994</p>	<ul style="list-style-type: none"> <li>• Cognitive processing: how students process content</li> <li>• Learning orientation (motivation): why they do it</li> <li>• Affective processes: how they feel about learning</li> <li>• Mental model of learning: how they see learning</li> <li>• Regulation of learning: how they plan and monitor learning</li> </ul>

<b>Witkin</b> 1978	<ul style="list-style-type: none"> <li>• Field dependent: global picture, ignore the details, and approach a task more holistically.</li> <li>• Field independent: discern figures, focus on details, serialistic</li> </ul>
-----------------------	--

*Table 1: Set of the well-known alphabetically ordered learning styles*

As can be noted, typically these learning styles instruments consist of a bipolar scale with a single characteristic at either end. Besides, many of the theories overlap and intersect:

➤ The distinction between field dependent and field independent individuals is similar to that differentiating holist and serialist [13; 14; 5; 15; 16]. That is to say Field dependent typically see the global picture, ignore the details, and approach a task more holistically. Field independent individuals tend to discern figures as being discrete from their background, to focus on details, and to be more serialistic in their approach to learning.

➤ Herman's model has some similarities to Kolb model such as the converger could map approximately over the quadrant A (analytical, logical, factual, critical and quantitative) [6]

➤ Looking closer to the table summarizing learning styles, similarities between Felder-Silverman and Kolb LSI as well as MBTI can be pointed out. Besides, Honey&Mumbord LSQ supports Kolb LSI.

These correlations lead us to pick out the different dimensions of learning styles, to be presented in section 4 through a multidimension model.

### 3 AC-GeneSys System

We present in this section the requirements and then the specification of a system called AC-GeneSys (Adaptive Courses Generation System) allowing the automatic generation of a built-in learning modules and/or courses aggregated from a finite set of learning objects stored in a learning object repository (LOR) taking in account, basically, two factors: the learner profile, preferences, cognitive and affective level; and the learning objects' quality. This would significantly reduce the modules/courses development time, and provides a pretty closer custom of the learning modules.

#### 3.1 AC-GeneSys Requirements

To use the AC-GeneSys, a student accesses the Virtual Learning Environment (VLE) Web site's home and logs into the system. The VLE presents the user, via the Web browser, with a customized desktop proposing, among others, the access to the AC-GeneSys' home page (see figure 1), which would be presented as a Web service.

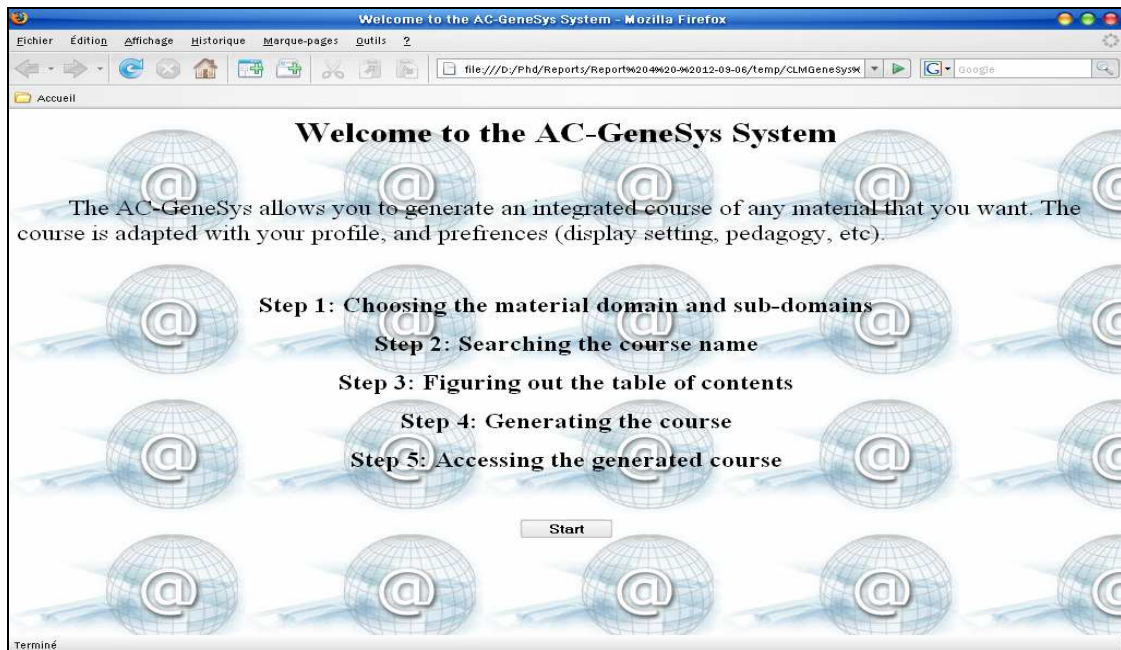


Figure 1: AC-GeneSys' home page

The student has to follow five steps until getting the generated course:

1. *Choosing the material domain and sub-domains*: The repository-supported domains and sub-domains are stored and organized into an XML-file. The student has to get down into the arborescence, across domains' lists (see figure 2), at the end of restricting the material sub-domain, and therefore have better and quicker results.

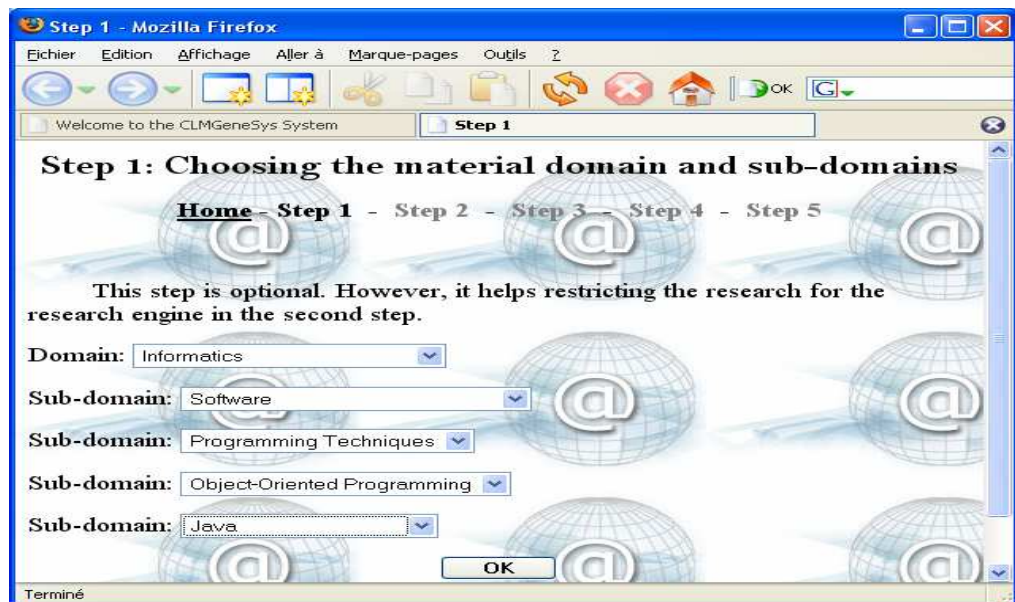


Figure 2: Choosing the material domain and sub-domains

2. *Searching the course name*: The second step's page is a research engine-like one (see figure 3). The student would be asked to type a finite set of words about the wanted course. Once he gets the results, he has only to choose among the resulting course names.



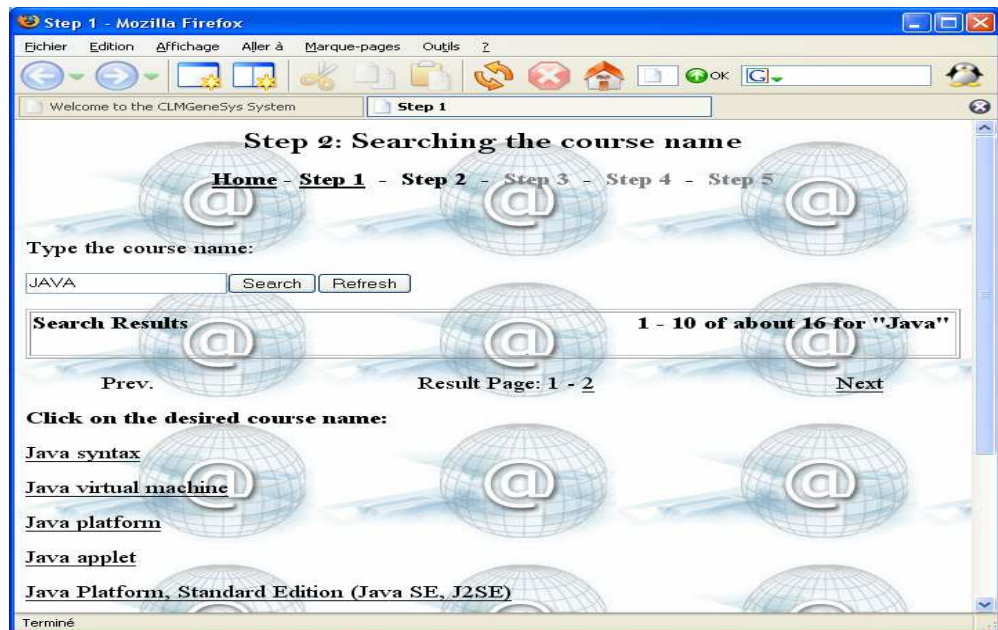


Figure 3: Searching the course name

3. *Figuring out the table of content*: A preliminary table of content would be proposed to the student. This table is susceptible to be modified by the student by removing or adding some elements (see figure 4).

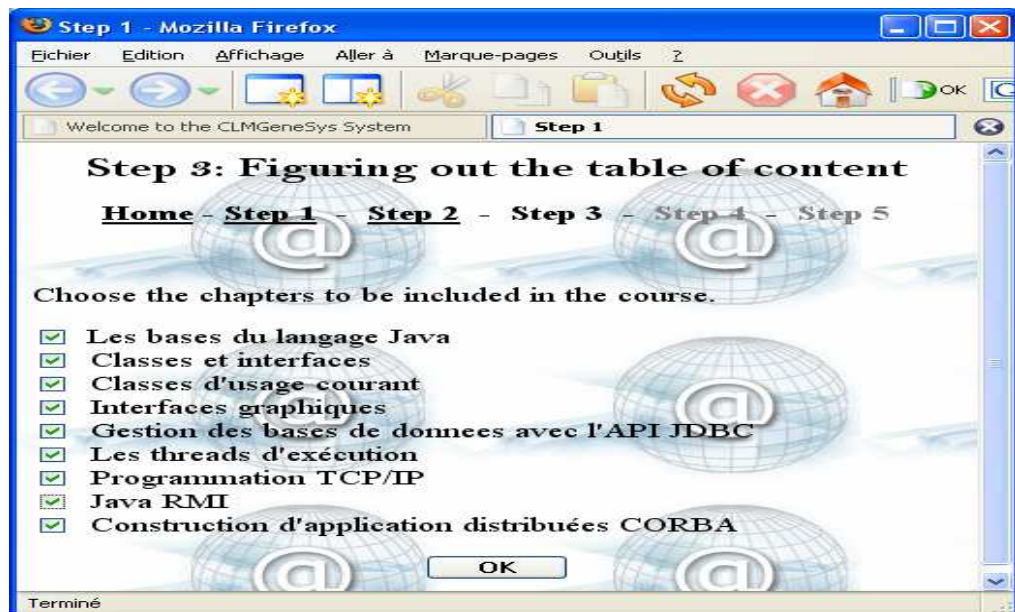


Figure 4: Figuring out the table of content

4. *Generating the course*: Eventually, the student confirms the chosen information and launches the generation process (see figure 5).

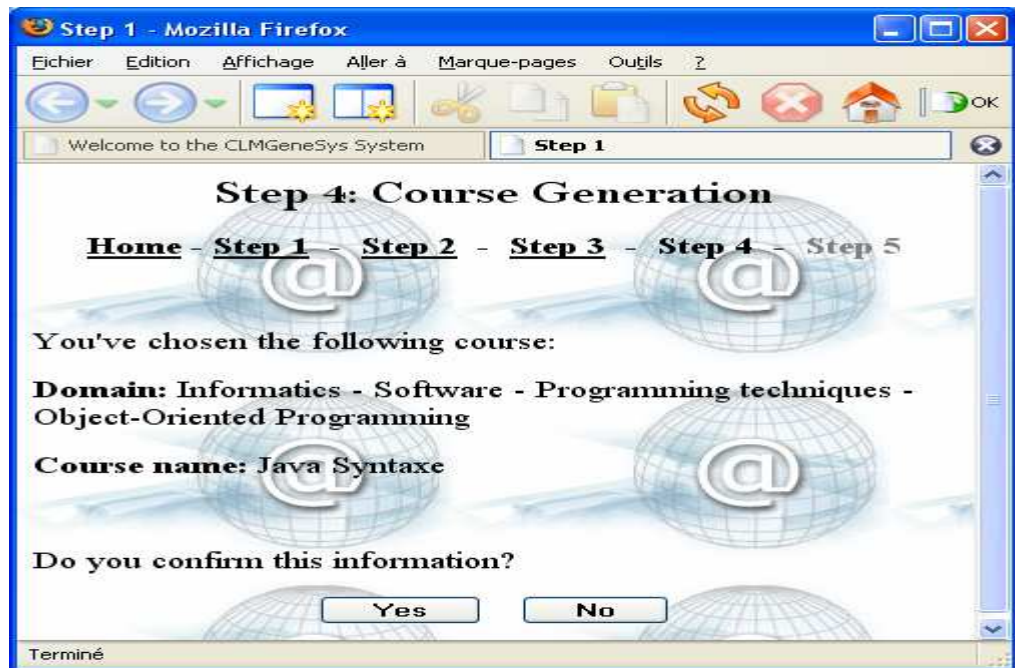


Figure 5: Generating the course

5. Accessing the generated course (see figure 6).

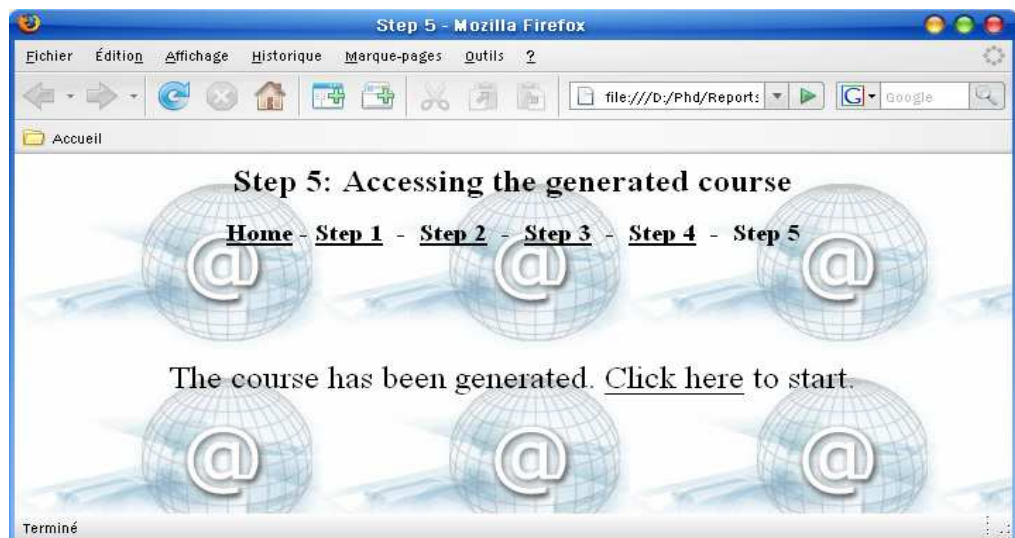


Figure 6: Accessing the generated course

The generated course is aggregated from a finite set of LOs stored in the LMS's repository. These LOs get through two main selections, before the "gluing" process:

□ 1<sup>st</sup> selection level:

- In the first place, the LOs should belong to the material domain chosen and confirmed by the student, and containing exclusively sections from the chosen table of content.
- The LOs should go with the student's display preferences, most adapted pedagogy; and his emotions, and social setting.



□ **2nd selection level:** The “competitor” LOs are ordered according to their rates. Indeed, the LOs should be stars-rated. This rating is a part manually and another automatically:

- Manual rating: The former users are asked to share their opinions about each “used” learning object.
- Automatic rating: The learning object solicitations, the access number, the reusability rate, etc.. are evaluation criteria that would be automatically assigned to the LOs.

### 3.2 AC-Genesys Specification

According to the System requirements, there are two development actions fields. The first one is on the LOs repository, and the second is on the course generator it self.

1. On the repository side, a metadata updating of the LOs is needed. Indeed, to generate a course adapted with the student’s socio-affective level, we need to get information whether a given LO meets these criteria or not.
2. On the generator side, two issues come up: LOs selection and the aggregation process.
  - a. On the student request (step 2), a first research engine would look for suggested courses names (organized into a Domains’ XML-file) grounded on the domain/sub-domains names provided by the student in the first step. Every course name comes with a predefined table of contents which would be proposed to the student in the third step. Once these information are confirmed (step 4), a second research engine would look inside the LOs metadata looking for those belonging to the course chapters. Each one of the selected LOs would be assembled according to its chapter name, and then would be ordered according to its accordance with the student cognitive, emotional, social, and behavioural level. The generated course would be an aggregation of the top selected LOs (see figure 9).
  - b. Regarding the crosscutting concerns acting into the LOs aggregation process (e.g. security, LOs selection; learner cognitive, emotional, social, behavioural aspects, etc), we would adopt the Aspect-oriented programming mixed with the Object-oriented programming. The LOs aggregation would be based on Victor Pankratius works [17], who “showed how crosscutting concerns of Learning objects can be isolated as aspects and how LOs can be generated”.

As a first step, we would identify involved aspects in the system. Then we would define them in the AspectJ meta-language. A prospective update of that meta-language would be necessary if we don’t succeed to express the features of some aspects. Next, we would develop a compiler allowing the defined aspects *weaving*, and generate a Java application of the personalized course according to weaved aspects (see figure 3). Eventually, we would use tools allowing presenting the application as a web service.

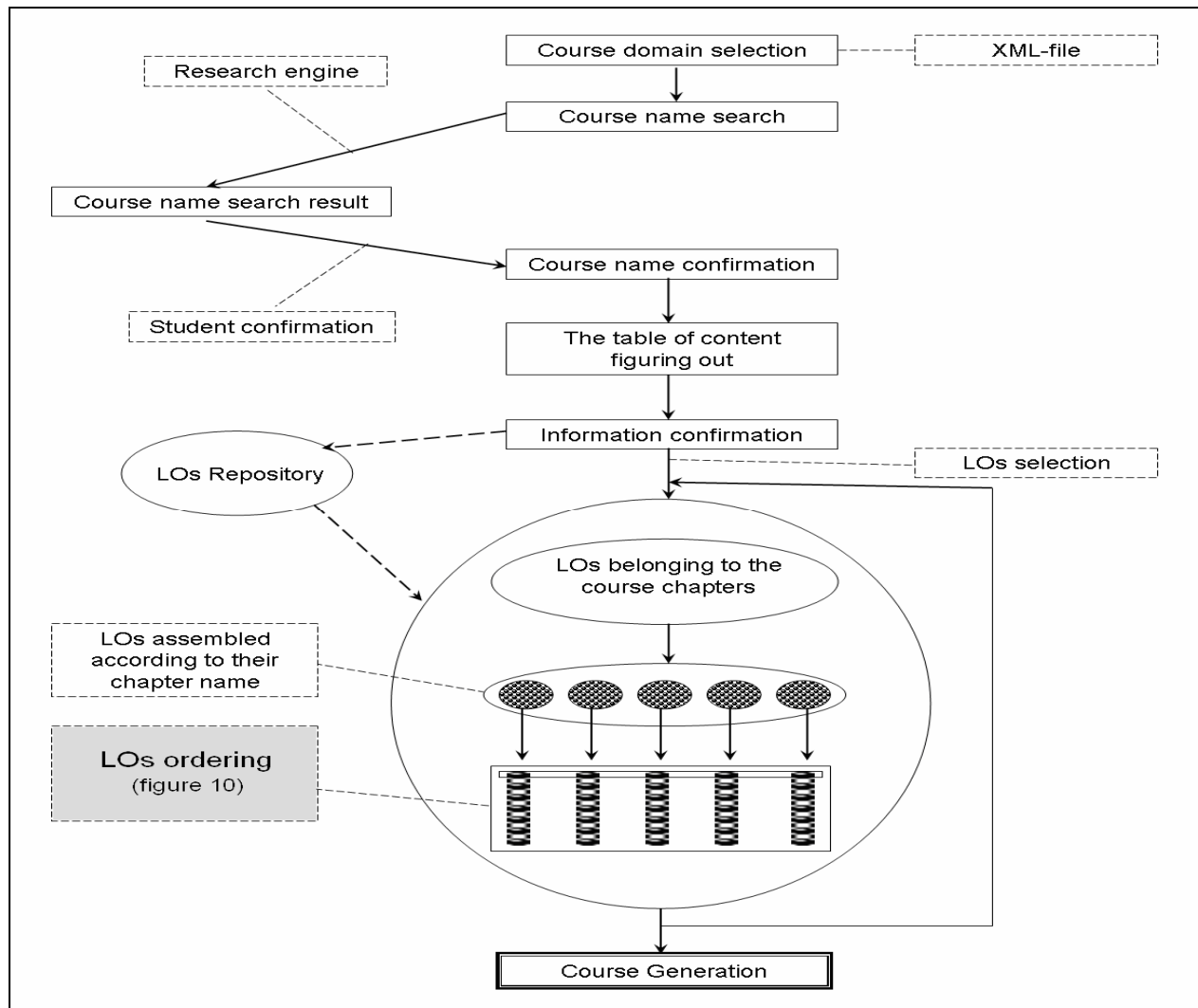


Figure 7: AC-Genesys Specification

## 4 A multidimensional model for learning objects and learners: AOL-Concept

### 4.1 Aspect-Oriented Programming

#### 4.1.1 Definition

Aspect-Oriented Programming (AOP) is a concept aimed to improve the structure of a system, which may be for example defined by a hierarchy of classes using Object-Oriented Programming (OOP) techniques [18].

#### a. Object-Oriented Programming

Object-oriented programming (OOP) has been presented as a technology that can fundamentally aid software engineering, because the underlying object model provides a better fit with real domain problems. But many programming problems have been found where OOP techniques are not sufficient to clearly capture all the important design decisions the program must implement. Instead, it seems that there are some programming problems that fit neither the OOP approach nor the procedural approach it replaces [19].

## b. Crosscutting Concerns

An example of crosscutting concerns is "logging," which is frequently used in distributed applications to aid debugging by tracing method calls. Suppose we do logging at both the beginning and the end of each function body. This will result in crosscutting all classes that have at least one function. Other typical crosscutting concerns include context-sensitive error handling, performance optimization, and design patterns [20].

## c. Aspect-Oriented Programming

Aspect-Oriented Programming is a new technology for separating crosscutting concerns into single units called *aspects*. An aspect is a modular unit of crosscutting implementation. It encapsulates behaviours that affect multiple classes into reusable modules. With AOP, we start by implementing our project using our OO language (for example, Java), and then we deal separately with crosscutting concerns in our code by implementing aspects. Finally, both the code and aspects are combined into a final executable form using an *aspect weaver*. As a result, a single aspect can contribute to the implementation of a number of methods, modules, or objects, increasing both reusability and maintainability of the code. Figure 8 explains the weaving process. We should note that the original code doesn't need to know about any functionality the aspect has added; it needs only to be recompiled without the aspect to regain the original functionality [20].

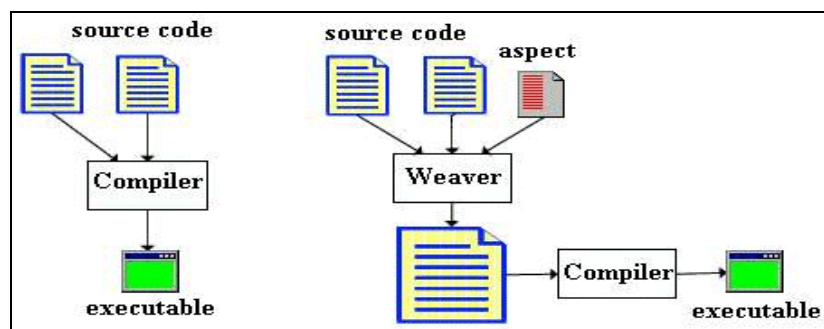


Figure 8: Aspect Weaver [20]

In that way, AOP complements object-oriented programming, not replacing it, by facilitating another type of modularity that pulls together the widespread implementation of a crosscutting concern into a single unit. These units are termed Aspects, hence the name Aspect oriented programming [20].

### 4.1.2 Aspect-Oriented Learning Objects

In [18] (see Appendix II), Victor Pankratius shows how crosscutting concerns of LOs can be isolated as aspects and how LOs can be generated. His examination focuses on XML technology.

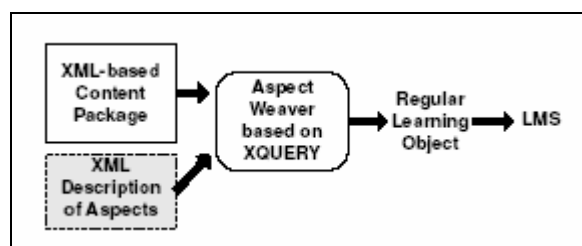


Figure 9: AOP concept for LOs [18]

The conceptual approach is depicted in Figure 9 Using an XML-based aspect language, aspects crosscutting the packages of several LOs are specified separately to achieve better modularization. An aspect weaver which “understands” the aspect language and which is based on XQuery, is used to combine the content packages and the corresponding aspect code with respect to specified join points. The output are regular LOs that can be processed as usual by Learning Management Systems [18].

## 4.2 Aspect-Oriented Learning Concept

According to our point of view, the Aspect-Oriented Programming is the most appropriate programming technique for the foreseen system; indeed, it allows isolating each of the aspects, then *weaving* them using an aspect weaver (AspectJ), and therefore getting the customized course.

The AOL concept is initially described by the figure 10. It would improve the e-learning on several point of views:

- **Learner:** fitted personalization, interactivity, self-evaluation, and progression at his own rhythm, etc.
- **Teacher:** pre-evaluation of the learner is no longer needed, he could depose as many chapters as he wants and it's not mandatory that they should logically adjacent, etc.
- **Developer/Company:** Cost remarkably reduced, development time notably decreased, much easier maintainability, over reusability, no course development staff to foresee (mainly one team for the LOs development, and another for maintainability and updating).
- **Theory:** Conflict solved between reusability and pedagogical effectiveness (indeed, AOL concept guarantees obtaining a high-quality and customized course, and at the same time maximizing LOs reusability), take advantage of the Aspect-oriented programming application in e-learning domain, a newer concept (AOL) easier to get standardized, etc.

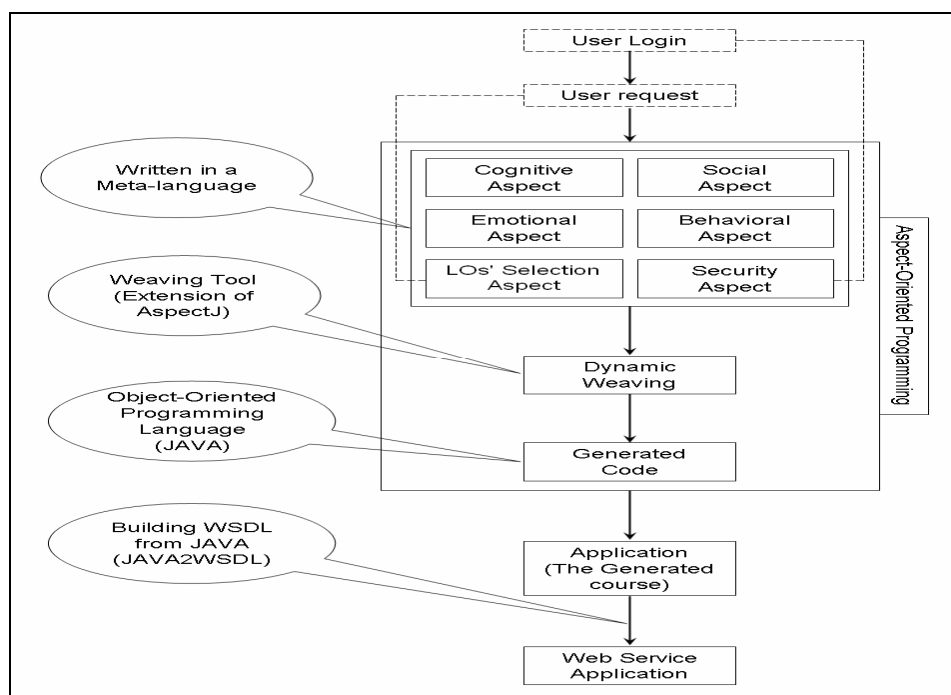


Figure 10: Aspect-Oriented e-Learning

### 4.3 Methodology

The AOL concept is the foundation on which the AC-GeneSys system would be based on. As we specify the AC-GeneSys system, we would define the AOP concept properties, and we would deduce an e-learning methodology based on the Aspect-oriented programming (notation: AspectJ meta-language, process: AC-GeneSys conceptual schema, tool: AC-GeneSys).

## 5 Conclusion

Other than an overview of the most well-known learning styles, this paper proposes a multidimensional model for learner objects and learners based on the Aspect-Oriented technique [19], and implemented by an automatic generation system called AC-GeneSys allowing:

- Reducing from 100 to more than 200 times the development time of an online complex course (foreseen time: from one to two hours. Estimated time according to the E-learning Guild study [21]: until 276 hours), and therefore, reducing notably courses' development cost.
- Providing personalized and well-adapted courses with the learner preferences, cognitive, emotional, social, and behaviourist level. In addition to personalization, this course would be guaranteed by the adoption of the Aspect-Oriented Programming technique allowing handling separately each personalization aspect, and then weaving them together at the end of generating a customized course, adapted with the learner profile.

## References:

- [1] R.M.Felder & al. Learning and teaching styles in Engineering Education. *Journal of engineering education*, 78(7), 674-681, 1988.
- [2] D.Milosevic & al. *Adaptive learning by using SCOs Metadata*. *Interdisciplinary Journal of Knowledge and Learning Objects*, vol 3, 163-174, 2007.
- [3] R.A.Howard & al. *Felder's learning styles, Bloom's taxonomy, and the Kolb learning cycle: tying it all together in the CS2 course*. *Proceedings of the twenty-seventh SIGCE technical symposium on computer science education*. 227-231, 1996.
- [4] M.Laing. *Teaching learning and learning teaching: an introduction to learning styles*. *New Frontiers in education XXXI/4*, 463-475, 2001.
- [5] S.Y.Chen & al. *Cognitive styles and hypermedia navigation: Development of a learning model*. *Journal of the American society for information science and technology*, 53(1). 3-15, 2002.
- [6] A. Cristea & al. *ODL education environments based on adaptivity and adaptability*. *E-Learn'02*, Montreal, Canada, AACE, 232-239, 2002.
- [7] M.T. McKay & al. Cognitive style and recall of text: an EEG analysis. *Learning and individual differences*. 14(1), 1-21, 2003.
- [8] F.Coffield. Learning styles and pedagogy in post-16 learning, *Learning and Skills Research Centre*, 2004.
- [9] C. Karagiannidis & al. *Adaptation rules relating learning styles research and learning objects Metadata*. 3<sup>rd</sup> international conference on adaptive hypermedia and adaptive web-based systems, Netherlands, 1, 136-145, AH2004.
- [10] R.M.Felder & al. Understanding student differences. *Journal of engineering education*., 94(1) 57-72, 2005.



- [11] L.Layman & al. *Personality types, learning styles, and an agile approach to software engineering education*. ACM SIGCSE bulletin, proceedings of the 37<sup>th</sup> SIGCSE'06, vol 38(1), 428-432, 2006.
- [12] F.Fordham. Introduction to Jung's psychology. Available at [www.cgjungpage.org](http://www.cgjungpage.org)
- [13] M.N.Brumbly. *Consistent differences in cognitive styles shown for qualitative biological problem-solving*. British journal of educational psychology, 52, 244-257, 1982.
- [14] B.Ash & al. *Identifying learning styles and matching strategies for teaching and learning*. ERIC Document reproduction service, ED270142.
- [15] G.Magoulas & al. *Adaptive web-based learning: accommodating individual differences through system's adaptation*. British journal of educational technology, Vol 34(4), 511-527, 2003.
- [16] H.Rumetshofer & al. *XML-based adaptation framework for psychological-driven e-learning systems*. Educational technology & society, 6(4), 18-29, 2003.
- [17] Aspect-Oriented Learning Objects. Victor Pankratius. Proceedings of the IASTED'05. Available: [http://www.aifb.uni-karlsruhe.de/BIK/vpa/Pankratius\\_WBE05.pdf](http://www.aifb.uni-karlsruhe.de/BIK/vpa/Pankratius_WBE05.pdf)
- [18] Aspect-Oriented Learning Objects. Victor Pankratius [On-line]. Available: [http://www.aifb.uni-karlsruhe.de/BIK/vpa/Pankratius\\_WBE05.pdf](http://www.aifb.uni-karlsruhe.de/BIK/vpa/Pankratius_WBE05.pdf)
- [19] Aspect-Oriented Programming. Gregor Kiczales, John Lamping, Anurag Mendhekar, Chris Maeda, Cristina Lopes, Jean-Marc Loingtier and John Irwin [On-line]. Available: <http://www.oberon2005.ru/paper/gk1997.pdf>
- [20] Developpez.com Web site [On-line]. Available: <http://www.developpez.com>
- [21] Elearning Guild's website [On-line]. Available: <http://www.elearningguild.com>

## Author(s):

<sup>1</sup> Anis Elferchichi

National School of Computer Science, University of Manouba, Tunisia  
ENSI, Campus Universitaire de la Manouba, 2010, La Manouba, Tunisia  
[anis.elferchichi@isg.rnu.tn](mailto:anis.elferchichi@isg.rnu.tn)

<sup>2</sup> Noura Ben Alaya

The higher institute of management, university of Tunis, Tunisia  
41, Rue de la Liberté, Cité Bouchoucha, 2000, Le Bardo, Tunis, Tunisia  
[noura.benalaya@edunet.tn](mailto:noura.benalaya@edunet.tn)

<sup>3</sup> Mohamed Ben Ahmed

National School of Computer Science, University of Manouba, Tunisia  
ENSI, Campus Universitaire de la Manouba, 2010, La Manouba, Tunisia  
[mohamed.Benahmed@riadi.rnu.tn](mailto:mohamed.Benahmed@riadi.rnu.tn)