



HAL
open science

Specification and Prototyping of Personalized Workplace Learning Solutions

Peter Dolog, Paul de Bra, Daniel Burgos, Alexandra Cristea, Stefano Ceri, Vladan Devedžić, Markus Eisenhauer, Geert-Jan Houben, Milos Kravcik, Paul Libbrecht, et al.

► **To cite this version:**

Peter Dolog, Paul de Bra, Daniel Burgos, Alexandra Cristea, Stefano Ceri, et al.. Specification and Prototyping of Personalized Workplace Learning Solutions. 2005. hal-00591533

HAL Id: hal-00591533

<https://hal.science/hal-00591533>

Submitted on 10 May 2011

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.



PROLEARN

European Commission Sixth Framework Project (IST-507310)

Deliverable D1.8

Specification and Prototyping of Personalized Workplace Learning Solutions

Editor Peter Dolog

Work Package WP1

Status document

Date 25/01/2006

The PROLEARN Consortium

1. Universität Hannover, L3S Research Center, Germany
2. Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI), Germany
3. Open University (OU), UK
4. Katholieke Universiteit Leuven (K.U.Leuven) / ARIADNE Foundation, Belgium
5. Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. (FHG), Germany
6. Wirtschaftsuniversität Wien (WUW), Austria
7. Universität für Bodenkultur, Zentrum für Soziale Innovation (CSI), Austria
8. École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
9. Eigenössische Technische Hochschule Zürich (ETHZ), Switzerland
10. Politecnico di Milano (POLIMI), Italy
11. Jožef Stefan Institute (JSI), Slovenia
12. Universidad Politécnica de Madrid (UPM), Spain
13. Kungl. Tekniska Högskolan (KTH), Sweden
14. National Centre for Scientific Research "Demokritos" (NCSR), Greece
15. Institut National des Télécommunications (INT), France
16. Hautes Etudes Commerciales (HEC), France
17. Technische Universiteit Eindhoven (TU/e), Netherlands
18. Rheinisch-Westfälische Technische Hochschule Aachen (RWTH), Germany
19. Helsinki University of Technology (HUT), Finland

Document Control

Title: Specification and Prototyping of Personalized Workplace Learning Solutions

Author/Editor: Peter Dolog

E-mail: dolog@l3s.de

Amendment History

Version	Date	Author/Editor	Description/Comments
0.95	7/12/2005	Peter Dolog	The first composite version created
0.98	16/12/2005	Peter Dolog	Changes based on comments from contributors
1.0	25/01/2006	Peter Dolog	Changes based on reviewer's comments

Contributors

Name	Company
Paul de Bra	TU/e
Daniel Burgos	OUNL
Alexandra Cristea	TU/e
Stefano Ceri	POLIMI
Vladan Devedzic	FON , University of Belgrade, Serbia and Montenegro
Peter Dolog	L3S
Markus Eisenhauer	FHG
Geert-Jan Houben	VUB and TU/e
Miloš Kravcík	FHG
Paul Libbrecht	DFKI
Maristella Matera	POLIMI
Erica Melis	DFKI
Wolfgang NejdI	L3S
Marcus Specht	OUNL
Craig Stewart	QMUL London, UK
David Smits	TU/e
Natasha Stash	TU/e

Legal Notices

The information in this document is subject to change without notice.

The Members of the PROLEARN Consortium make no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Members of the PROLEARN Consortium shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

1 Introduction

The main goal of this document is to survey the existing approaches for the personalization and adaptation of eLearning systems which are proposed by partners from the PROLEARN consortium. This document enables the comparison of various methods and techniques, and facilitates their integration or reuse within the Consortium and beyond. It offers a cohesive reference for other workpackages dealing with work processes and business models; the use on models as means for specification opens the possibilities to integrate the work with other workpackages which also focus on modeling but from other perspectives (e.g. WP4 focusing learning object metadata and querying).

Workplace learning is connected to performance at work and is defined in [Roth et al 1999] as “the integrated use of learning and other interventions for the purpose of improving human performance, and addressing individual and organizational needs. It uses a systematic process of analyzing and responding to individual, group, and organizational performance issues. It creates positive, progressive change within organizations by balancing humanistic and ethical considerations” [RSS1999, p. 121].

The purpose of this document is to survey existing approaches for **specifying and prototyping software solutions and content authoring which introduce personalization in workplace learning**, with a focus on the contributions given by the PROLEARN consortium. The goal is to outline the main challenges for future work.

Specifications are used in many fields for several purposes. In this context, the process of creating specifications can be considered either from the **software engineering** point of view or from the **content authoring** point of view, where the former is concerned with specification of generic tools helping in workplace learning while the latter is concerned with the production of learning content to be presented within the tools. Sometimes, the two activities can be combined.

Both views include two distinct activities: specifying an **overall picture** about the whole system, and then giving **details about the components** that belong to such picture. In software engineering, the former activity produces an architecture definition, while the latter activity produces detailed specifications. In authoring, the former activity is called authoring in the large (focusing on the overall structure of the content and links between content pieces) and the latter is called authoring in the small (focusing directly the content of the pieces).

In the following, we will describe how these components are addressed in specifications of solutions for personalized learning. First, we define the fundamental concepts which should be modelled in order to enable personalization (section 2). Section 3 describes the software components which are normally required for design solutions for learning. The design solutions provided within the PROLEARN consortium are described in section 4. Section 5 focuses on specific adaptation strategies which are adopted in the specification proposals we reviewed. We also provide a discussion and comparative analysis. The paper concludes with summary and further directions.

2 Conceptual Modelling and Specifications

In previous deliverables of WP1, we have described which conceptual structures are relevant for personalized learning. The personalized adaptive learning solutions operate on top of **domain**,

resource and user metadata to provide personalized access to learning. The metadata are utilized for decisions on which learning resources to provide and how to organize them during the presentation. The domain of workplace learning can be characterized by:

- **Procedural knowledge** – e.g., the activities which have to be performed at particular workplace to achieve results expected from the workplace;
- **Propositional knowledge** – e.g., the concepts connected with performance at the workplace;
- **Context** – e.g., situations in which particular concepts and activities make sense.

A similar delineation has been defined by Meyer and Clark while characterizing goals of e-learning content courses [MC2002].

When providing personalized solutions for workplace learning, a developer or workplace learning constructor reason exactly about those kinds of knowledge when developing digital training material for workplace learning. Conceptual modelling helps in understanding and specifying relevant aspects of a solution or a problem. We can consider the each of these kinds of knowledge as a specific view on workplace with its own conceptual model. In personalized solutions, learner characteristics are considered in all of the views.

One of the approaches applied to workplace learning is **constructivism**. Constructivism is based on principle that learners actively construct knowledge by integrating new information and experiences into what they previously understood, including the refinements and reconciliation of the old knowledge to fit into the new situation [Billet 96]. The knowledge construction includes - besides procedural and propositional knowledge - also contexts, usefulness, and motivation of applying constructed knowledge in the way it was constructed. Therefore, a learning situation should also take a context where knowledge should be applied into account. One particular constructivist approach at the workplace is **cognitive apprenticeship** i.e. guiding or helping learners through a problem or situation. As the workplace learning approach is highly contextualized and individualized, personalized software solutions seem natural to support different individual learners and contexts by observing the learners and context features to enable adaptation.

2.1 Procedural Knowledge and Learning Activities

The procedural knowledge character of workplace learning should be reflected in specifications as well. In particular, conceptual structures of activities can be a source for guidance and a different activity path can be suggested according to the role and background of a person or situation he has to follow.

Several approaches are known from the literature where Learning Design was used to model and specify Adaptive Methods of instruction [LD2005, TH2005, RB2005]. Examples in the literature span from adaptive navigation support, adaptive content presentation, to the adaptation of pedagogical strategies, and adaptation of complete learning environments. Furthermore, topics of manifest oriented approaches where the learning design (the instructional logic) can be used in standalone solutions have been discussed. For example, the project aLFanet integrated different learning technology standards for building on Learner Modeling Standards (IMS-LIP), learning object metadata, performance information and test specifications (IMS-QTI), and a description of instructional logics.

As a major insight, LD seems already in its current version to be a highly potent format to describe a wide variety of adaptive instructional methods and offer lots of tools (e.g. RELOAD - <http://www.reload.ac.uk/ldeditor.html>) for creating and deploying adaptive designs in learning environments. Furthermore the industrial uptake of learning design as a powerful and standardized layer to describe and specify instructional logics is very positive. Therefore the main benefits of using LD for describing and exchanging adaptive designs in ProLearn will be:

- A standardized representation basis for adaptive methods currently mostly described in proprietary solutions of different partners.
- A clear and standardized interface for industrial uptake and the chance for building scalable adaptive and personalized learning environments.

Considering Reusability, LD has several advantages, especially talking with instructional designers and trainers in educational departments which are applying learning technology at the workplace.

1) enhanced searches:

- Search using ontology of Learning Designs, with templates and good practice examples.
- Search for Learning Designs that have been used with a particular kind of content.
- With a certain amount of artificial intelligence the repository could also
- Search for Learning Designs with a similar pedagogy
- Search for content that has been used in similar Learning Designs

2) Users may want to retrieve a unit of learning to use in their own learning design. If so, the repository needs to understand the structure of Learning Design in order to provide all the elements which make up that fragment.

3) A Learning Design aware repository can provide information on how learning resources have been used in other units of learning, giving insight into the nature of the resources.

4) The repository could store metadata provided by users, on what units of learning have been used for and how successful they were. This would be valuable to both learners and teachers.

2.2 Propositional Knowledge and Content

The propositional knowledge in adaptive learning environments reflects knowledge about content (a domain model) and a learner (learner or student model). Usually the *network model* is used in order to model a domain (see e.g. [KS2004] or [B2001] for review of adaptive systems where it can be found). This is an advanced form of domain model forming a semantic network. Each concept consists of its description, synonyms, relations with other concepts (the relation types can be specified by the author).

For student modelling, usually an *overlay* model is used, which for each concept stores some data about the student's knowledge on that concept. Another opportunity is a *historic model*, which stores the events related to the student activities and the status of each learning object regarding the student. An *event* is a record in the database with the information that a particular

user performed a certain action with a specific learning object, together with a timestamp when it happened (see e.g. B2001] for review of adaptive systems which use it).

Educational materials are interconnected with the domain model by *multi concept indexing*. Each learning object can relate to many concepts. Concepts can be regarded as input (prerequisite) for a learning object or its output (the learning object contributes to the explanation of the concept).

2.3 Context Knowledge

Context knowledge in this report is understood as a context of work activities (or performance) at workplace. A business process analysis has been used to capture work processes (see e.g. [AH1995]) in enterprises with the aim to understand and potentially change enterprise behaviour to achieve higher profit. The captured processes described in kind of activity models or workflow models can serve as a context for learning. Learning seems an integral part of this understanding and cannot be disconnected from them. Conceptual modelling of business processes and connected workflows are usually based on activity charts. However, the analysis models (specifications) have been used so far just in tools supporting performance of workers. The connections for worker reflections, their improvements and learning as a need or outcome of competency or skill gap are still missing. The forthcoming European projects like TenCompetence, COOPER, or PROLIX seems promising especially in this context of connecting specifications of work processes and activities and learning activities with learner assessment.

3 Software Components for Adaptive Workplace Learning Solutions

Rapid prototyping and reuse of software solutions in different contexts is usually based on frameworks. Those frameworks have usually several components and artefacts in common. Conceptual models as specifications are used to customize the frameworks but keep the common architectures. Model driven approaches operate with generators to target software environments providing rapid prototyping allowing fast feedback from customers.

Agent solutions and solutions based on distributed architectures seem to be appropriate for virtual workplaces. Different agents serving their purpose and/or recommendation functionality [DHNS2004, DHNS2004b] for different types of learners and contexts seem to be suitable for the adaptive, individualized and contextualized workplace nature of workplace learning.

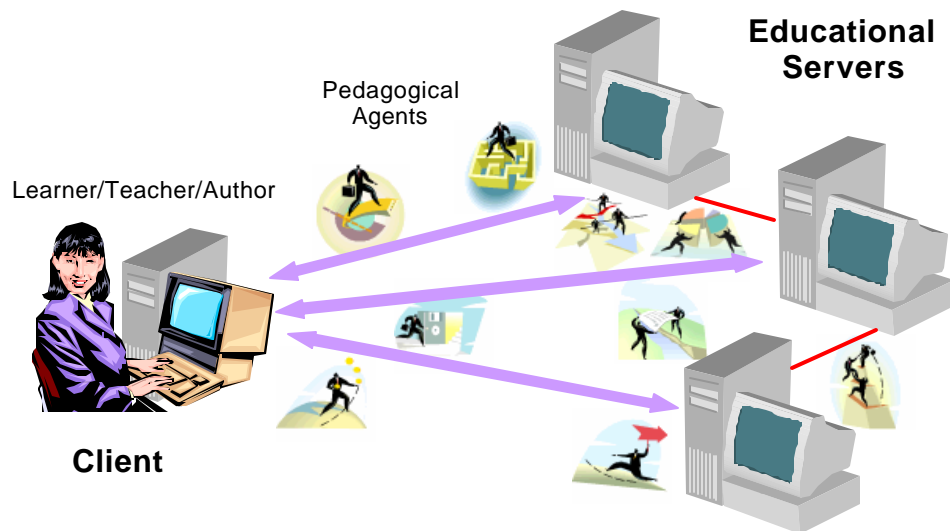


Figure 3.1 - The setting for Semantic Web-based education: the major actors and the technology

Figure 1 shows a likely setting for teaching, learning, collaboration, assessment, and other educational activities in the Semantic Web setting [D2003, D2004]. This setting is a generalization of the virtual classroom architecture. Educational material may be distributed among different *educational servers* - specific Web applications running on physical servers and responsible for management and administration of, as well as access to the material. Learners, teachers, and authors access the educational material from the client side. *Educational content* is any educational material pedagogically organized and structured in such a way that interested learners can use it to get introduced to a knowledge domain, deepen their understanding of that domain, and practice the related problem-solving skills.

The different agents operate on different kinds of knowledge represented in structures described in the previous section. Intelligent *pedagogical agents* provide the necessary infrastructure for knowledge, content, and information flow between the clients and the servers. They are autonomous software entities that support human learning by interacting with students/learners, teachers, and authors, and by collaborating with other similar agents, in the context of interactive learning environments [JRL2000]. On behalf of the learners, pedagogical agent access educational content on the servers by using high-level educational services. An *educational service* is a Web service designed specifically to support a learning, a teaching, or an authoring goal.

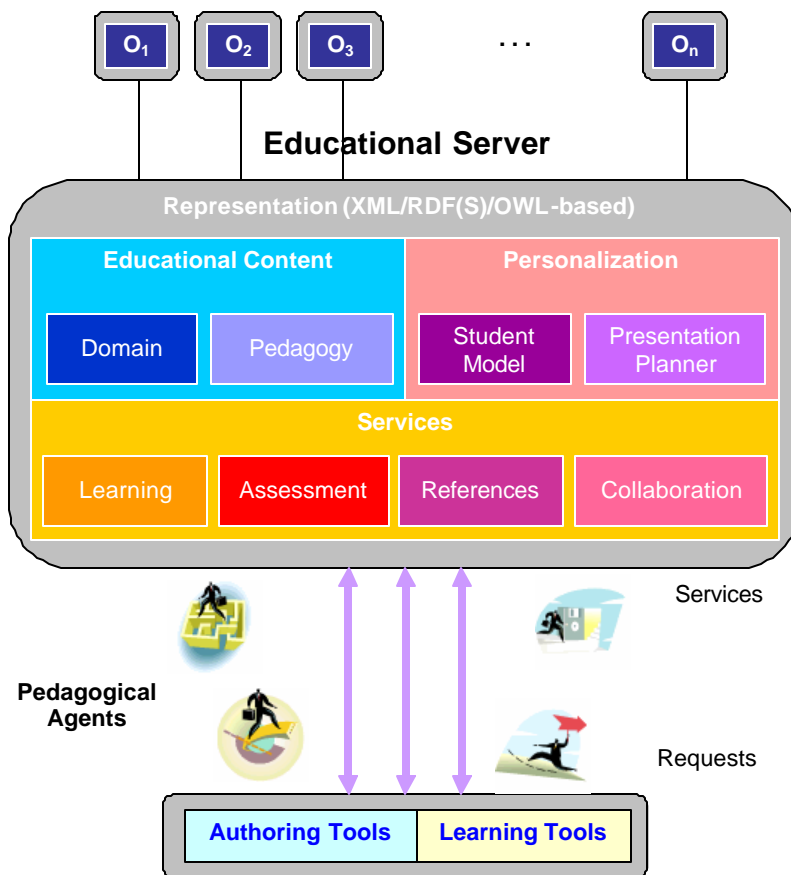


Figure 3.2 - Inside and outside an educational server

Figure 2 shows a general model of educational servers for Semantic Web-based education (see also Figure 1). From the learner perspective, the server appears to act as a powerful extension of the intelligent tutor built into the *learning tools* the learner uses. For the purpose of this discussion, assume that the learning tools include different Web-based ITSs, Adaptive Educational Hypermedia Systems (AEHSs), and similar intelligent learning environments. In combination, the educational server and the intelligent tutor possess enough *domain* and *pedagogical* knowledge to conduct a learning session [D2003, D2004]. The two kinds of knowledge are represented as pieces of educational content, such as LOs, as well as IMS Learning Design descriptions of instructional design and the tutor's heuristics. In fact, it is a generalization of the knowledge model of an intelligent tutoring system (the Expert Module and the Pedagogical Module in the architecture of an intelligent tutoring system).

An educational server is also supposed to possess enough intelligence to arrange for *personalization* of the learning tasks it supports. The server may include a *presentation planner* to help the intelligent tutor select, prepare, and adapt the domain material to show to the learner. The tutor gradually builds the *learner model* (*student model*) during the session, in order to keep track of the student's actions and learning progress, detect and correct his/her errors and misconceptions, and possibly redirect the session accordingly. In the end of the session, the learner model is saved. It is then used along with other information and knowledge to initialize the next session with the same learner.

From the instructor's (teacher's) perspective, an educational server enables ontology-based access to and browsing of constantly updated collection of LOs of different granularities, as well

as access to resources related to selection and customization of teaching strategies. The server also provides different class monitoring and intervention options to support virtual classrooms, as well as direct access to students' models both during and after the learning session. The server itself may provide the tools to support teachers in their activities, or they may use client-side tools to connect to the educational server and use the resources it provides.

From the author's perspective, educational servers extend authoring tools. Through authoring tools, domain authors access different ontologies (see the O_i boxes in Figure 2) stored in ontology libraries managed by educational servers. The authors can develop their own ontologies and publish them in the libraries. They can also reuse and extend ontologies from the libraries for their own authoring purposes. Furthermore, authors can create, store, update, and delete LOs, instructional designs, courses, and the like, all accessible through the server. Likewise, they can access student models stored on educational servers and use them to create more advanced models, with different adaptation and personalization options enabled by AH techniques.

Note that Figure 2 does not imply any specific physical distribution of educational resources. From the perspective of all categories of end users (learners, teachers, and authors), the resources should appear as if they were located on a single server. In other words, each educational server should support seamless integration of resources it provides itself with resources provided by other similar educational servers. It is the task of system developers to make educational servers support this important option for Semantic Web-based education by deploying the latest Semantic Web engineering technologies.

Once again, Figure 2 represents a generalized *model* of an educational server. A number of variations may be used in practical implementations. For example, as an alternative to having a full range of intelligent tutoring functionalities implemented in a Web-based learning environment on the client side, the educational server may provide at least some of them by means of intelligent Web services.

At a first glance, the model from Figure 2 may appear as a mere vision of some faraway future development [AD2004]. However, the model can be easily mapped (instantiated) to different other models and architectures proposed in the literature. For example, a correspondence can be drawn between the educational server model as presented here and the AHA! architecture proposed in [BACH2004] and the UPML model [MDCG2003]. There is also a good deal of overlap between the model from Figure 2 and the models underlying popular LO repositories (such as ARIADNE and MERLOT – see also WP4) and their accompanying tools.

4 Modelling Approaches

Each of the components described in previous section can be described by one or more views focusing propositional, procedural and contextual knowledge about the components. In this section, we will focus on the approaches how to capture the knowledge about content, navigation and personalization provided by the techniques proposed by partners from PROLEARN consortium.

4.1 The UML Guide and WebML

WebML is a high-level model and technology for building server-side Web applications, which is the result of a ten-years research conducted at Politecnico of Milano. The WebML mainly

captures factual knowledge about content and navigation in the content. UML-Guide is a UML-based personalization engine conceived at the L3S laboratory, which dynamically generates personalized user guides by considering user profiles and context-sensitive data stored and managed at the client side. The UML-guides captures a procedural knowledge about navigation or user guidance by means of transitions between learning or navigation states.

An example of WebML hypertext schema for generic eLearning application provided at an educational server is depicted in fig. 3.3.

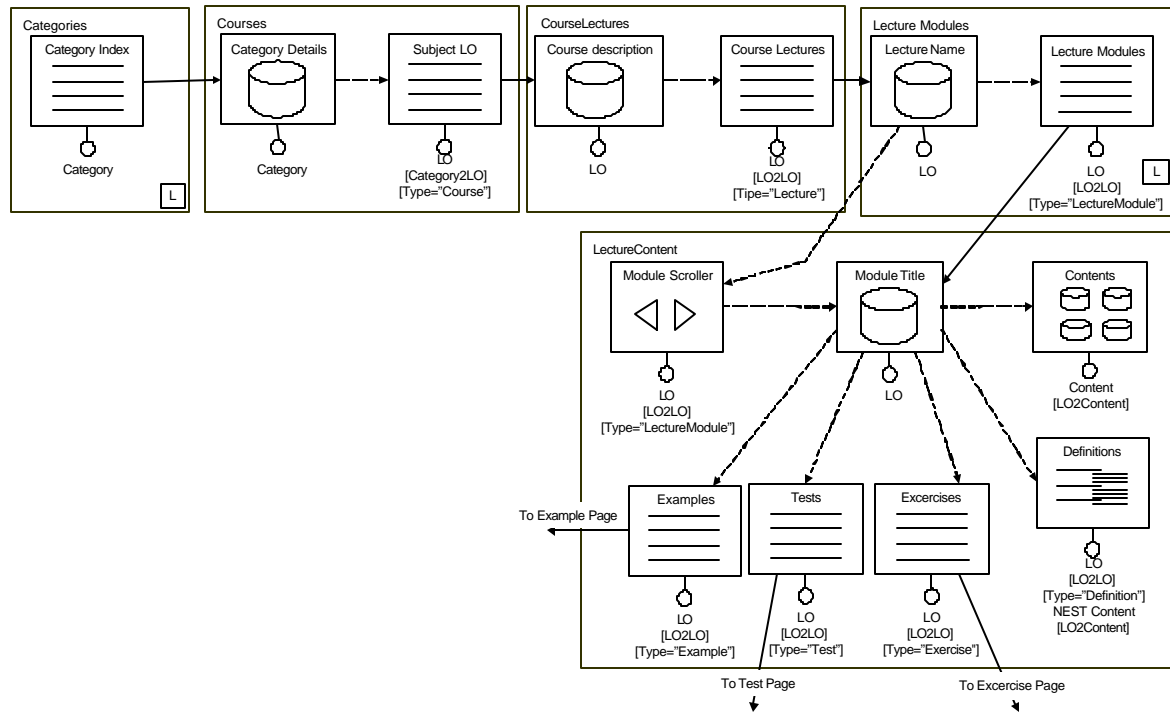


Figure 4.1: WebML hypertext schema for accessing generic eLearning application

Figure 4.1 reports a simplified excerpt of the WebML hypertext schema, which refers to pages for selecting a lecture module, and accessing its contents as well as associated definitions, exercises examples, and tests. The pages are modelled as rectangles. The content of pages is modelled by units of different kind which are inside of the pages. The lecture module selection is operated by means of a navigation chain, in which users progressively select a subject category (*Categories* page), then a course that refers to the selected category (*Courses* page), then a lecture (*CourseLectures* page), and finally the lecture module they are interested in (*LectureModules* page). Pages *Categories* and *Lecture Modules* feature an “L” label, which indicates that they are *landmark* pages. This property represents that the two pages will be reachable from any other page of the hypertext, by means of landmark links, added to the pages when the application code is generated.

Contents of the selected lecture module are shown in page *LectureContent*. In this page, the data unit *Module Title* shows the title and a short description of the learning object, the *Contents* multidata unit shows texts and images that compose the module, while the *Definitions* hierarchical index shows titles of the definitions associated with the module and, nested, the corresponding contents. The index units *Example*, *Tests*, *Excercises* then show the lists of examples, tests and exercises available for the current lecture module. The selection of one entry from such lists leads users in a different page where the corresponding contents

are displayed. As represented by the `Module Scroller` unit, users can browse lecture modules in a *Guided Tour* navigation, which allows moving forward and backward in the (ordered) set of modules available for the currently selected lecture.

State diagrams are used in UML-Guide for modelling the user navigation in an hypertext; each state represents the production of a given information chunk on the device observed by a user, and each state transition represents an event caused by user interaction that leads to the production of a new chunk of information. State diagrams therefore provide an abstraction of hypertext trails, where each trail can be adapted by taking into account the user competencies, background, level of knowledge, preferences and so on [DN2003]. In this way, UML state diagrams are a suitable interface for UML-Guide, whose primary purpose is to build adaptive hypermedia systems. From eLearning point of view, states in the state diagram represent a learning state which is achieved after some actions and interaction of a user. Possible interaction points on a learning path are modelled as transitions which are raised by events. Observations about user actions are modelled as side effect actions and adaptation as conditions on states and transitions.

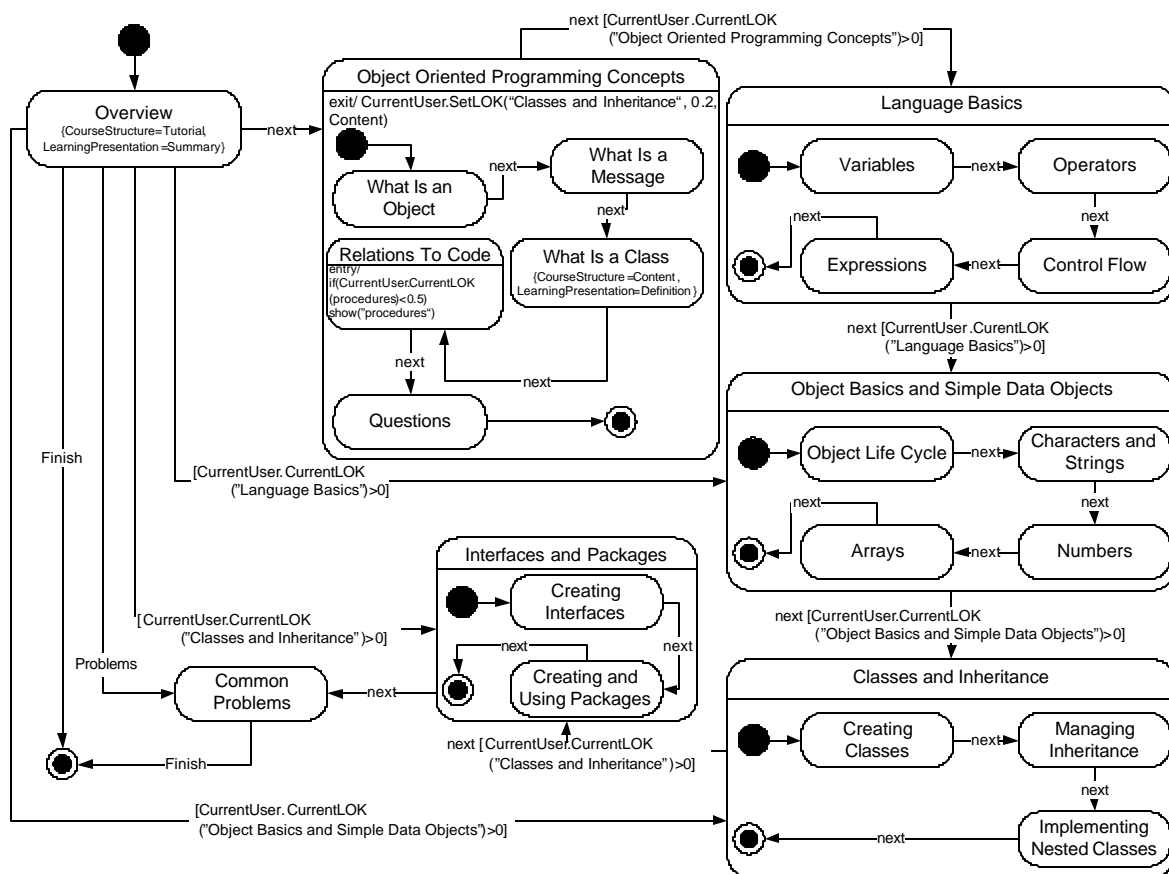


Figure 4.2: A Model of Adaptive Navigation Guide as UML State Diagram for Java Tutorial

The UML-Guide state diagram of Figure 4.2 illustrates a personalized learning environment for teaching object-oriented programming in JAVA. The chosen personalization example focuses on link adaptation. The tutorial starts with an overview of available lectures, as represented by the Overview state, which summarizes the available lectures in the tutorial, as represented by the Summary value in the LearningPresentation tagged value. It also presents the high level tutorial steps (Tutorial value in the CourseStructure tagged value). Links from the overview point not only

to the first section of the tutorial, but also to the other main sections; all these links, except the first one, are associated with guard conditions that check that the user has enough knowledge to jump directly to the respective lectures.

The next step from the Overview is a lecture on the Object Oriented Programming Concepts. This state is accessible without any prerequisite on background knowledge; it is a composite state, containing four steps, represented by four substates: What is an Object, What is a Message, What is a Class, and Relations to Code. The Relations to Code state also shows an entry procedure addressing content level adaptation. The procedure applies to a learning step about building programs; it states that if the current user does not have sufficient knowledge on basic concepts about object-oriented programming procedures, then learning content on procedures will be added.

The next step from the Object Oriented Programming Concepts is the composite state Language Basics. The transition between the two states features a next event and a guard. The guard specifies a link level adaptation rule, saying that the link is recommended when current user level of knowledge is greater than zero. The other learning steps modelled in the state diagram can be interpreted similarly.

4.2 Adaptive Learning Design

The learning designs are activity graphs which can be interpreted adaptively. The activity graphs capture procedural knowledge about learning processes, namely the control flows between the activities in particular learning situations. The learning design prescribe a sequence of activities for a learner and staff roles in learning which are carried out in a particular environments [KP2001]. The adaptation is supported by allowing conditionally constrained branching of control flow in learning activity flows where the conditions are based on user properties.

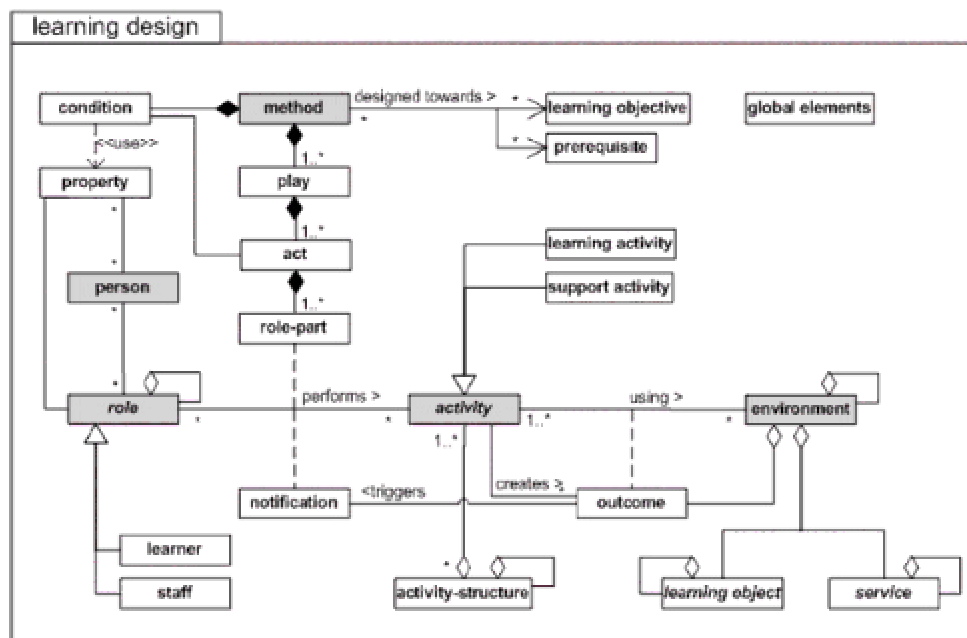


Figure 2.2 - Conceptual model of overall Learning Design (gray coloring is only used to increase the readability). Taken from [LD2003]

Based on [LD2003], a person usually gets a role in the teaching/learning process. His role fulfill certain learning outcomes by performing more or less structured learning and/or support activities within a learning environment. The environment refers to learning objects and services to be used during the performance of the activities. Adaptation process can be embedded into a method used for playing the design where the activities, learning objects, services and roles can be assign dynamically, based on conditions and properties.

The method is based on particular learning objectives (specification of the outcomes for learners), expect certain prerequisites (specification of the entry level for learners) to be fulfilled. The method consists of one or more concurrent play(s); a play consists of one or more sequential act(s) and an act is related to one or more concurrent role-part(s), each role-part associates exactly one role with one activity or activity-structure. The teaching-learning process is modelled in the method on the notion of a theatrical play. A play has acts, and in each act has one or more role-parts. The acts in a play follow each other in a sequence (although more complex sequencing behavior can take place within an act). The role-parts within an act associate each role with an activity. The activity in turn describes what that role is to do and what environment is available to it within the act. In the analogy, the assigned activity is the equivalent of the script for the part that the role plays in the act, although less prescriptive. Where there is more than one role-part within an act, these are run in parallel.

A notification is triggered by an outcome and can make a new activity available for a role to perform. The person getting the notification is not necessarily the same person who creates the outcome. For instance, when one student completes an activity (= an outcome), then another student or the teacher may be notified and set another activity as a consequence. This mechanism can also be used for learning designs where the supply of a consequent activity may be dependent on the kind of outcome of previous activities (adaptive task setting designs).

The explicit roles specified in this language are those of learner and staff roles. Each of these can be specialized into sub-roles, but no vocabulary is put forward for this. It is left open to the learning designer to name the (sub)-roles and specify their activities. For example, in simulations and games different learners can play different roles, each performing different activities in different environments.

Activities can be assembled into activity-structures. An activity-structure aggregates a set of related activities into a single structure, which can be associated to a role in a role-part. A structure can model a sequence or a selection of activities. In a sequence, a role has to complete the different activities in the structure in the order provided. In a selection, a role may select a given number of activities from the set provided in the activity-structure. This can, for instance, be used to model situations where students have to complete two activities, which they may freely select from a collection of e.g., five activities contained in the activity-structure.

Activity-structures can also reference other Activity-structures and reference external Units of Learning, enabling elaborate structures to be defined if required.

An example of IMS LD is a Unit of Learning on Personalization: Quo-Builder 2 (available at <http://hdl.handle.net/1820/428>)

The example is focused on personalization and it allows the set-up of the unit of learning with full personalization of questions, answers, right answers, ranges, points earned, messages of feedback and welcome, title. The questions and related properties are local (loc-property) and keep the same value for all the users in the same run but personal answers and calculations are private and linked to every participant (locpers-property)

The objectives for this Unit of Learning are: a) to work on personalization, adaptive learning and run-time tracking (monitoring); b) to work on locpers-properties and loc-properties, data-type real, initialization; c) to show and hide layers with different content depending on real time results; d) to store and calculate values depending on choices; e) to set-up key activities to follow the learning process depending on the set-up and on the feedback acquired; f) to monitor the progress of the participants in real-time

An excerpt of the IMSMANIFEST.XML file is:

```
...
<imsld:then>
  <imsld:show>
    <imsld:class with -control="false" class="msg-0" />
  </imsld:show>
  <imsld:hide>
    <imsld:learning-activity-ref ref="flow1" />
    <imsld:learning-activity-ref ref="flow2" />
    <imsld:learning-activity-ref ref="flow3" />
    <imsld:class with -control="false" class="msg-1" />
    <imsld:class with -control="false" class="msg-2" />
    <imsld:class with -control="false" class="msg-3" />
  </imsld:hide>
  <imsld:change-property-value>
    <imsld:property-ref ref="all-questions" />
    <imsld:property-value>0</imsld:property-value>
  </imsld:change-property-value>
</imsld:then>
<imsld:else>
  <imsld:if>
    <imsld:and>
      <imsld:is>
        <imsld:property-ref ref="all-questions" />
        <imsld:property-value>1</imsld:property-value>
      </imsld:is>
    <imsld:and>
      <imsld:less-than>
        <imsld:property-ref ref="accuracy" />
        <imsld:property-ref ref="Level2-from" />
      </imsld:less-than>
      <imsld:greater-than>
        <imsld:property-ref ref="accuracy" />
        <imsld:property-ref ref="Level0-to" />
      </imsld:greater-than>
    </imsld:and>
  </imsld:and>
</imsld:if>
<imsld:then>
```

Learning design is supported by authoring tools. Examples of high level editors are the MOT Learning Design Editor (yet to be released), Collage Editor (based on the Reload LD Editor and focused on collaborative patterns), ASK LDT (a graphical editor Level B compliant) and LAMS (not at present IMS LD compliant). The Alfanet LD Editor is an example of a close to the specification editor, and it will soon be joined by the RELOAD LD editor, and the Komposer tool from GTK Press.

Editors can also be general purpose (such as those mentioned above) or specific for a particular pedagogic approach. An example of this kind of approach is the high level EduPlone Learning Sequencer, which provides some simple pedagogic templates which can be applied to learning resources. There is a clear need for more high level tools, both general and specialised, which

will make it easier for non-technicians to become directly involved in authoring Units of Learning. All of the described tools supporting IMS LD level B can make use of general properties to implement personalized learning designs and can support classical adaptive hypermedia approaches of conditional sequencing and adaptive presentation. IMS LD level C furthermore extends current approaches via a general notification mechanism. Mainly coming from a pedagogically driven and collaborative learning background the use of IMS-LD level C furthermore allows for the integration of adaptive self driven learning, adaptive synchronous and asynchronous collaborative learning approaches.

The application which runs a Unit of Learning for a learner is called a player. It coordinates the learners and teachers in their various roles, and tracks their progress, delivering the appropriate learning activities and resources at the right time. It is inevitable with a complex specification like Learning Design that there will be ambiguity and disagreements when deciding how a Unit of Learning should be interpreted by a player. The function of a Reference runtime player is to provide an example for developers which show how any given Unit of Learning should behave when it is run. This is essential if true interoperability is to be achieved. The UNFOLD projects lists several open source and other players the are learning design compatible and can easily be integrated with existing LMS technology.

An IMS Learning Design is a very complex application, and so the development of Coppercore, the Learning Design Engine, a Free and Open Source engine which handles the complex runtime tracking, is a great step forward. Other developers can now add a variety of interfaces to create a range of general and specialised players.

Edubox is an EML player which is used by the OUNL to deliver their online courses, developed together with Perot Systems Netherlands. OUNL has now reached an agreement with Blackboard Inc to incorporate Edubox into their system and to adapt it to handle IMS Learning Design.

4.3 Content Based Adaptive Solutions for Prototyping

Fraunhofer FIT develops Adaptive Learning Environment (ALE) that has been already used by several universities in the area of design and architecture. According to the ALE (the authoring tool integrated in ALE was developed in cooperation with the bureau42 GmbH as author42™) approach [B2003, CM2003, SKPK2001, SKPK2001a, SKKPH2002, SKKPH2002, KS2004] authors create learning objects, structure them (hierarchically and by referential hyperlinks), assign them attributes (pedagogic roles, metadata), and specify concepts as an alternative structure. By means of automatic indexing the learning objects are interconnected with concepts, i.e. the system can find for each concept all its occurrences (including synonyms) in learning objects. Then, a student viewing a learning object or a concept is provided with access to all the related information.

In 2003 ALE has been considered as in that time the most advanced form-based interface among adaptive educational hypermedia systems [B2003]. This kind of interface is more intuitive for authors than that one provided by markup based authoring tools. In the meantime, the ALE authoring interface has been further developed to support template based authoring. This should make the authoring process more intuitive. The system enables reusability of learning objects and content blocks as well as their representation in various multimedia formats. Additionally, it allows separation of the content and the layout for both learning elements and content blocks by means of predefined design templates (in HTML and CSS).

Before a new course is started to be developed the content analysis is usually performed. Then in the template configuration module system administrators can define learning element (paragraph) types and their templates to adapt it to different application domains. Authors can use predefined templates, but these can also be customized and put into the repository so that others can share them. ALE provides an embedded HTML editor operating in the WYSIWYG mode. There is a special (HTML like) language used for tagging of templates. The template system enables definition of the layout as well as restriction of media elements that can be inserted in certain parts of the paragraph (e.g. an image with a specified resolution, a QuickTime movie). The following example shows a simple template for the content block *Definition*:

```
<table border="0" cellspacing="0" cellpadding="10" class="definition">
  <tr>
    <td valign="top" align="left">
      [AuthorTag ContentBlock]
      allowText = "true"
    [/AuthorTag]
  </td>
</tr>
</table>
```

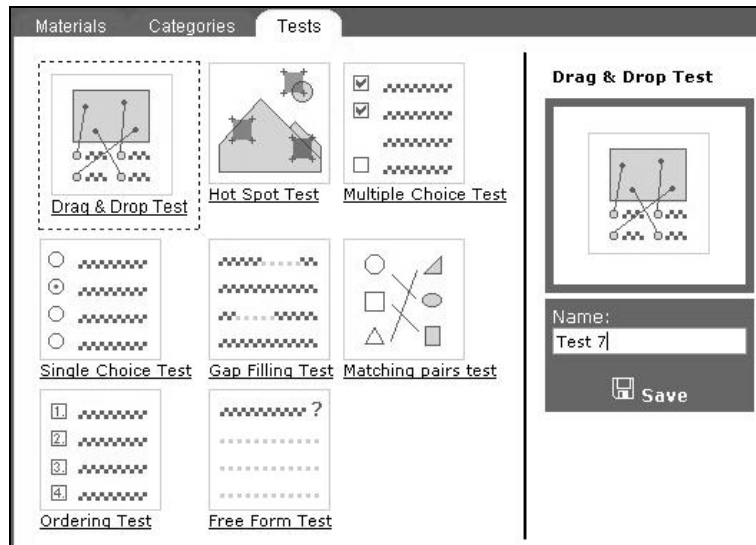


Figure 3.6: Creating a new learning object in ALE.

To create a new learning object the author first chooses its type from a predefined list of templates (Fig. 3.6). A template defines a specific type and structure to keep a certain consistency across the particular installation of the system. Inside a learning element the author creates individual content blocks (text, multimedia, or URL), defines their attributes (options) and specifies the templates (Fig. 3.7). In *Options* the author can specify how individual content blocks are to be integrated into a coherent learning object.



Figure 3.7: Learning element authoring in ALE.

4.4 LAOS based Environments

The LAOS based environment from TU/e focuses direct authoring for personalized workplace learning solutions. It focuses mainly propositional knowledge about content, connected navigation and presentation. The environment offers a support for a) specification, design, model, patterns and framework oriented work; b) authoring systems.

- LAOS framework:** The LAOS model is a general framework for authoring of adaptive hypermedia, based on the AHAM model. Its basic components are: *domain model* (DM), *goal and constraints model* (GM), *user model* (UM), *adaptation model* (AM) and *presentation model* (PM). The framework defines the elements of each (sub-)model based on concept map representations, with the exception of the adaptation model, that is based on the LAG model of adaptivity. The major difference to AHAM (and other models) is a clear separation of primitive *information* (content) - and *presentation-goal related information* (e.g., pedagogical information in educational systems and prerequisites). For more information and recent work, consult [CS2005, C2005, C2005a, PDCSA2005, CSAC2005, SCBA2005, CSB2005, CSBC2005].
- Extraction of authoring patterns:** In order to systematize the authoring process, patterns of authoring have been extracted. For instance, in more recent work, patterns of authoring corresponding to learning style processing have been identified [BCSB2005].
- MOT system:** MOT is a generic authoring system for adaptive hypermedia, based on the LAOS model. MOT implements the domain model (DM) in LAOS in the form of *concept maps*, structured into *hierarchies of concepts*. These concepts can also be connected, beside the hierarchical form, by *relatedness relations*. (In LAOS, any type of relationships is allowed, as long as they represent inherent domain connectivity, and not presentation-related connectivity, such as prerequisites.) The order of the concepts in the hierarchy is

not prescriptive for the final order of presentation. Concepts are represented in MOT via their *attributes*, as required by LAOS. Each of these attributes has a semantic unity.

4.5 ActiveMath

ActiveMath is a complex and rather mature Web-based adaptive learning environment for mathematics. Learning in mathematics is problem oriented and competency based, i.e. learner tries to achieve a competence/skill to deal with a mathematic formulaes. Conceptually, at the specification level, this is similar to the workplace learning which is also competency based and problem oriented.

The ActiveMath has a number of components and interactive learning tools. The basis for handling semantics of learning content is provided by its semantic (mathematical) content markup which is additionally annotated with educational metadata. Several components, tools, and external services can make use of that content markup, for instance, a course generator, a semantic search engine, and user input evaluation services.

Adaptivity in ActiveMath is provided on the basis of the knowledge encoded in the metadata and supports self-directed learning through the mathematical content and exercises.

4.5.1 Components of ActiveMath

The ActiveMath is based on the goals of the Semantic Web in various ways. It addresses the broad accessibility and interaction of Web-resources and Web-services and strives at the reuse and interoperability of content resources and of services. ActiveMath' open architecture integrates a number of components, tools and services. The components are modularly implemented and use different types of knowledge such as content, exercises, knowledge about the learner, knowledge about the context, pedagogical or tutorial strategies, etc.

Currently, the functionalities of ActiveMath's components and services include:

- a tutorial component consisting of an adaptive course generator and a reactive suggestion component (tutorial component)
- semantic retrieval of learning objects (mediator)
- updating beliefs about the learner (learner model)
- playing and evaluating interactive exercises (exercise subsystem [GGM2005])
- search for content and semantics as well as dependencies (semantic search)
- efficient rendering and delivery of content (presentation subsystem [JLWM2004])
- evaluation of mathematical expressions in user input (computer algebra systems)
- interactive concept map tool [MKH2005]
- learner management
- input of (semantic) mathematical expressions (mathematical input editor).

4.5.2 Knowledge Representation on ActiveMath

One objective of using a generic semantic markup language is to keep the encoded content reusable and interoperable by other, even non-educational, mathematical applications.

ActiveMath uses the semantic XML-markup language for mathematical documents, OMDoc [K2001] , for its content encoding. OMDoc has evolved as an extension of the OpenMath (European) standard for mathematical symbols and expressions [BCCDG2004]. ActiveMath uses these representations for the following reasons: they provide a semantics for mathematical expressions and a standardized structure of the content. Especially OpenMath is used for a variety of mathematical services and thus, all applications can take advantage of each other's advances in tools and implementations.

ActiveMath's content is represented as typed items called "learning objects" annotated with metadata. That is, the content is rather fine-grained in structure. The semantic information includes types, relations, and other mathematical and educational metadata. The type indicates a characterization of the items as collection, theory, concept or satellite items: an OpenMath symbol defines a mathematical concept abstractly; a theory assembles concepts and it can import other theories; concepts (definitions, algorithms, and assertions/theorems) are the main items of mathematical contents, whereas satellites (exercises, examples, explanations, introductions) are additional items of the content which are related to one or several concepts. All items are accessible via a unique identifier. The content (including the OpenMath symbols) induces ontology.

For ActiveMath, we have extended the primarily mathematical OMDoc to serve educational purposes. For this, the metadata characterize not only mathematical and organizational properties and relations, and intellectual property rights properties (by Dublin Core), but also educational annotations. These metadata include properties such as difficulty, learning-context, and field. They are compatible with LOM and extend it.

Moreover, items can be linked by relations. The DOMAIN_PREREQUISITE relation expresses a mathematical dependency between concepts, ie, it describes which concepts are mathematically necessary in order to define the current one. The FOR relation links satellite items to concepts and definitions to symbols. For instance, an example or exercise may illustrate or train a related definition. The against relation describes an item as a counter example or misconception of the related concept. The is_a relation serves to represent mathematical hierarchies and special cases, eg, a "function" is a specific "relation".

For a more detailed coverage of ActiveMath' knowledge representation see [MA2003]. The service architecture handling this knowledge is described in [MGHLUW2005].

4.5.3 Content Authoring Tools in ActiveMath Framework

Content in ActiveMath is written using a much enhanced XML-editor. This choice has historical reasons and we acknowledge the need for a more visual tools for novice authors. The paradigm of source-and-build editing has however several other implications and advantages which we promote:

- edition, being based on a well-defined syntax can be hinted for and verified
- the knowledge structure is specified in a single file which can evolve rapidly with the needs and changes of the learning environment
- this specification enables easy input of metadata: the support of the XML -editor allows such an input as easy as a form-based input
- insertion of the relations in the metadata uses the learning environment: the author can insert a reference by dragging links referring to content-items from the browser directly

into the editor which will make sure the appropriate XML-element is created with the shortest relative reference.

- the source encoding has been polished to maintain the highest possible readability. Such artifacts as hidden attribute values in the Document Type Definition, managed set of referenceable items, or the usage of a special linear syntax for mathematical formulae make the source document relatively easy to read.
- we have strived for short write-and-test-cycles allowing an author to view changes in quite instantly.
- feedback on the errors at the source is provided, as is done in most compiler-based projects, has turned out to be a very important feature to raise the trust of the author.

The adaptivity of ActiveMath makes feedback on errors even more important: although the visual rendering of a small piece of content is easily checked for within the learning environment during test-cycles, other parts of the content can only be checked for in isolated places of the learning environment. An example of such is given by the relations between items: each test-cycle makes sure all relations in store are verified and otherwise provides feedback indicating, in a click, where the error should be.

Currently, the validation of the course generation or the learner model updates can only be done by really using ActiveMath: during the projects several prototype-users were defined which were the basis for the verification of the course-generation.

5 Adaptation Strategies based on Specifications

Adaptation algorithms use specifications as input for performing the adaptation. The adaptation based on specifications deduces additional models or knowledge relevant for recommendation, hiding, or composition of learning environment features to particular individual learner. This section is devoted to adaptation strategies which use conceptual models from previous section.

5.1 Model Driven Customization for Workplace Learning

Adaptive behaviour is dedicated to client side agent in integrated approach of the UML-Guide and WebML [CDMN04, CDMN05]. The main aim was to exploit the potentiality of the two research works and to investigate possible integrations. The ultimate goal has been the building of an integrated framework, providing courseware companies with a systematic method - and implementation support, for fast prototyping and development of personalized vertical e-learning applications.

While educational server-side solutions are dominant, yet bringing some intelligence to the client for performing user-specific functions may be highly beneficial in some cases. Client-side solutions can reveal as being more dynamic, more adaptive, and protective for sensitive user data. They may be very effective for “remembering” the local context or being aware of the local peculiarities of individual user interaction which is very relevant for virtual solutions for workplace learning. Also, a clear separation of concerns between the client and the server may lead to interesting business opportunities and models. These characteristics are well suited for workplace learning solutions offered by elearning vendors as they provide simple customization approach to suit elearning for particular context and further provide flexibility by dynamic adaptation to suit individual learner needs.

Based on such premises, [CDMN04, CDMN05] proposes the combination of WebML and UML-Guide, for the engineering of workplace oriented e-learning systems.

To be able to add client side adaptive guidance to a problem presented in the generic eLearning application as depicted in fig. 3.4, additional model is needed.

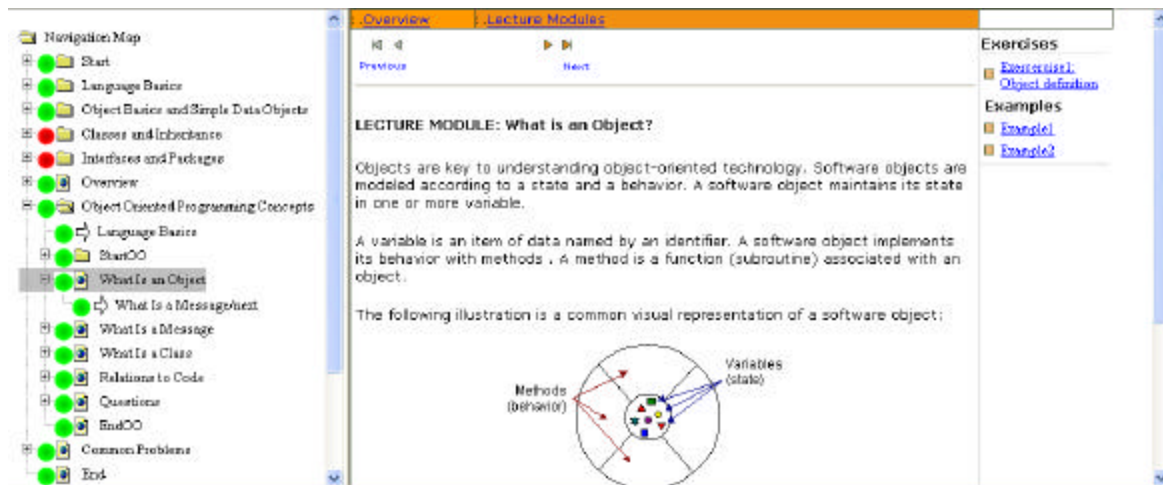


Figure 3.4: A generated application with adaptive client side agent, which operates on the left frame and educational server generated right frame for a generic problem based activity

The left frame adaptive guide is generated from a state diagram of the UML-Guide where adaptation is determined according to a knowledge about a user solving particular problem in an educational activity. The idea behind a state machine view is that it models educational states of a learner and decides according to his progress with activities for particular transitions in his educational path. An excerpt of such a model used to generate the left frame of fig. 3.4 is depicted in fig. 3.5.

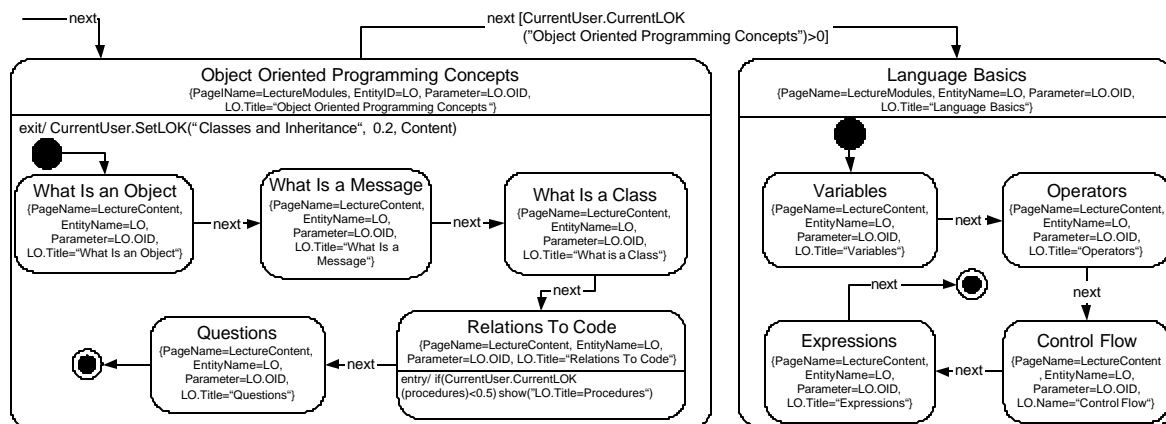


Figure 3.5 Excerpt of UML-Guide state diagram extended with tagged values representing WebML concepts.

Figure 3.5 depicts an excerpt of state diagram extended with tagged values for WebML concepts, needed for computing WebML links. This work must be performed by UMLGuide designers, typically in the course of the transformations required for “implementing” UML-Guides starting from their high-level descriptions (as illustrated in Figure 3). For instance, Object Oriented Programming Concepts is a lecture. The corresponding page name is LectureModules from WebML hypertext model. The entity used to store lectures in the

WebML data model is LO. The title used as an attribute to identify the lecture is the same as the state name. Entry and exit actions are transformed if they send parameters in WebML links, as it is in the case of `Relations To Code` (where the parameter of the `show` method is replaced by specific WebML parameter `&LOM.Title=Procedures`). It is worth noting that, although in our example tagged values for page and entity names are constant values, in more complex cases they can be specified as well as parametric selectors, so as to retrieve their values from the WebML XML specification based on specific conditions. Also, more tagged values can be needed, for example for identifying content units IDs, in situations where the selection of pages is based upon the content units they include. Queries for retrieving OID's of the WebML concepts and content are submitted through a specifically designed interface to the WebML run-time components. The interface consists of the two functions `GetWebMLConcept(Type, Name)` and `GetWebMLRecordOID(Entity, Attribute, Value)`.

In the integrated framework, the WebML method and its development support environment is used for generating the server-side "backbone" of a generic "vertical e-learning system", collecting a large number of Learning Objects (LOs). The UML-Guide is used for specifying and building company-specific e-learning curricula that guide users in the fruition of LOs for reaching some learning goals.

The proposed approach capitalizes on the use of two systems that both start from high-level abstractions, and are both capable of automatic deployment of the implementations:

- The WebML method is based on the use of high-level concepts, such as the notions of entity and relationship to denote content, and of page, unit, and link to denote hypertexts. These abstractions are automatically turned into implementation artefacts by means of WebRatio (<http://www.webratio.com>), a tool for the automatic deployment of Web applications.
- UML-Guide is based on the use of UML state diagrams, whose nodes and arcs - representing states and transitions - are turned into XML specifications. A client-side translator, written in XSL, turns such specifications into a user interface facilitating the adaptive use of the application.

A typical elearning scenario, which can take benefit of the proposed integration, is the one where a courseware company develops and distributes a vertical application for elearning, running on the company's server, specified and developed through WebML. The vertical incorporates learning situations in the format of guided experiences, exercises, tests, definitions and examples for computer science, arranged according to the ACM categories, and learning paths with checkpoints for the learner. Thus, such a vertical has generic learning experience accessible through navigation mechanisms enabling a generic user to access such experience through predefined navigation paths. The vertical makes available to users generic navigation mechanisms, based on hypertexts, such as guided tours or indexed accesses to pages based on broad categories.

The vertical is used by Small-Medium Enterprises (SMEs) wishing to build personalized e-learning curricula, to be used by their employees for focused training activities. We assume that each SME has a clear instruction goal (for example, teaching its employees how to integrate Java programming into Oracle 9i), and that it can use UML-Guide to specify it in UML; we assume that UML state diagrams, together with a vocabulary listing of all the learning experiences available in the vertical, may be an easy-to-use interface for the SME designer. UML-Guide specifications select the concepts and activities to be covered in the learning paths, as well as the workflow driving the student in the learning process. We also assume that each SME has a clear view of its employees' competencies, and thus is able to constrain possibilities in the learning paths by adaptation rules based on such competencies. These rules enable

adaptive experience and situation selection from the WebML vertical and also enable to adaptively indicate, show, and hide links in the learning path, and adaptively customize their targets.

Based on the conducted experiments [CDMN04, CDMN05]., following observations have been collected:

- The users of the composite system is provided with the standard, WebML-generated interface of the vertical, populated by content spawning a large body of knowledge; but the focused learners is also provided with a guide, available on an interface that can be opened “aside” the main one, and that points to pages and contents published by the WebML-generated interface, according to a specific learning objective and user experience.
- The use of high-level WebML abstractions in the context of UML-Guide enables the specification of a powerful client-side personalization engine. The resulting application generator can be considered an “adaptive hypermedia generator” in full strength, whose potential expressive power goes well beyond the experiment we have performed.
- The tools prove to be highly complementary and easily integrated, as it is sufficient to reuse concepts of WebML inside UML-Guide to provide concept interoperability and the URL generation technique of the WebML runtime inside the UML-Guide XSL code to provide systems interoperability.

5.2 Authoring and Adaptation Strategies

Adaptation strategies can be classified to three categories: adaptive selection of media items, adaptive ordering of media items, and adaptive tools for navigation support.

Selection of media items is based on different learner preferences. This can also be extended to different learning styles. For instance, the same information (or the same concept) can be presented in various ways, by using alternative media types [B2001] – audio, video, image, text, etc. Depending on the learner’s style a certain item (or group of items) may be included into the final presentation. E.g., in LS terms, we can say that *verbalizers* [RB2000], who prefer textual information, may be presented with text and possibly spoken audio; whilst *imagers*, who prefer pictorial information, can be shown images, diagrams, graphs, charts or other items about the same concept [RR1995]. The selection process can be applied not only to media items, but also to other types of items.

The order in which information items are processed can be based on learner needs. E.g., some learners prefer to learn things by *doing something actively* first whilst others prefer to *collect data first* (corresponding to the *active* and *reflective* learning styles, respectively). Moreover, some learners tend to learn through a linear, step-by-step, logical and systematic process, whilst others want to see the big picture before they tackle the details (corresponding, respectively, to *sequential* and *global* LS [FS2000]).

Depending on the learner preferences, different learning tools can be provided. In terms of catering for learning styles, for example, *field-dependent* [WMGC1977] learners can be provided with a concept map, graphic path indicator, advanced organizer, etc. in order to help them organize the structure of the knowledge domain. Alternatively, *field-independent* learners might be provided with a control option showing a menu from which they can choose to proceed with the application in any order [TPG2002].

5.3 LAG-XLS: An XML Learning Style Adaptation Language

LAG-XLS [SCB2005] bases *selection* and *ordering* of concepts on the attributes and values of their sub-concepts¹, as follows. The names of the attributes and their values indicate how these sub-concepts represent the parent concept. For instance, if the media attribute is audio, the sub-concept will represent an audio version of the concept. Another goal was that of expressing monitoring strategies. To achieve this, the adaptation language for AHA! contains elements specifying user model updates.

The resulting LAG-XLS language, corresponding to various strategies (extracted from what was previously implemented in 'adaptation assembly' form only, but also from literature review are presented in Figure 4.1. The meaning of the DTD *elements* and *attributes* is explained below.

```
<?xml version="1.0" encoding="UTF-8"?>
<ELEMENT strategy (description, if*)>
<!ATTLIST strategy name CDATA #REQUIRED>
<ELEMENT description (#PCDATA)>
<ELEMENT if (condition, then)>
<ELEMENT condition (#PCDATA)>
<ELEMENT then (action*, select?, setDefault*, sort?)>
<ELEMENT action (UMvariable, expression)>
<!ATTLIST action attributeName CDATA #REQUIRED>
<ELEMENT UMvariable (#PCDATA)>
<ELEMENT expression (#PCDATA)>
<ELEMENT select (showContent*, showContentDefault*, showLink*)>
<!ATTLIST select attributeName CDATA #REQUIRED>
<ELEMENT setDefault (expression)>
```

Figure 4.1 LAG-XLS DTD

- *strategy* – is the root element of a file corresponding to a strategy, attribute *name* – the name of the strategy;
- *description* – is the strategy meaning; e.g., the corresponding learner model for which this strategy has been created;
- *if* – a statement to specify if-then-else rules (currently we have only *if* statements within the *strategy* element, however we are thinking about applying other statements as well, like *for*, *while*, etc., as in LAG);

¹ In AHA! there can be different types of concepts, e.g., *abstract page* or *object* (fragment) concepts. Abstract concepts do not have a resource associated with it. Page concept can have one or more associated resources. Fragment concepts should be included into pages; they can have multiple resources, however they represent alternative versions of a *part* of a page. These resources are well-formed documents, to be scanned by the AHA! engine for other recursively included objects. Therefore they do not have a header and cannot be viewed separately.

- *condition* – appears within an *if* statement; a Boolean expression which can contain user-related information; e.g., about the user’s LS;
- *then* – defines actions to be performed when the *condition* is satisfied;
- *else* – an element defining an alternative set of actions;
- The following elements are used to define *how* the adaptation is performed:
- *select* – selecting a concept representation from a number of existing ones to be included into the final presentation;
- *sort* – sequencing different concept representations depending on the user’s learning style, reordering them from most to least relevant.

The “select” and “sort” elements have an attribute “attributeName”. The value is provided by the author depending on the aspects of the concepts he wants to include or reorder in the final presentation. For example, we have a concept which has several children representing it via different types of media. All the children concepts have an attribute “media”. The value of this attribute for different concepts can be “audio”, “video”, “text”, “image”, etc. In the final presentation for various strategies (links to) media items can be explicitly included or not; similarly, (links to) media items can be ordered in different ways:

- *showLink* – showing a link to the concept representation;
- *showContent* – showing the content of the concept representation;
- *showDefaultContent* – showing a default content specified by the author when no other representation is found for a particular concept;
- *action* – specifies how the user model is updated; attribute *UMupdate* shows whether it is an absolute or relative update;
- *UMvariable* – indicates which user model variable should be updated, namely which attribute of which concept;
- *expression* – is the value used for user model update.

5.4 FOSP Method

FOSP method [K2004] is aiming at collaborative authoring of adaptive educational hypermedia, which can be generalized also for other application areas. It addresses the objective to simplify the authoring process and make it more efficient. Its main idea is to separate the partial results produced by different authors in such a way that they can be reused. This concerns also adaptation strategies that specify how the domain model and the context model attributes should be processed to present the content to the user accordingly. An instruction designer specifies adaptation as sets of content object preferences for different contexts. We have identified a pattern in the adaptation process that consists of four operations – Filter, Order, Select, and Present.

To illustrate our method let us consider the following first. When a teacher wants to teach a learner certain new knowledge or skill, he usually first decides what types of learning resources are suitable for the particular user, e.g. for one learner it can be a definition and an example, for another a demonstration and an exercise. Then he should order the resources, i.e. decide whether to start with the definition or the example. Each learning resource can have

alternative representations, so the teacher has to select the most suitable one – narrative explanation, image, animation, video, etc. This illustrates the basic reasoning behind our method, which takes into account also different presentation opportunities of various devices. Note that we are proposing a technique to specify an adaptation strategy, not an adaptation strategy itself. Specification of adaptation strategies is a task for instructional designers.

In the FOSP method, an adaptation strategy maps the domain model (learning objects with attributes and metadata) and the context model (including the learner model with learning styles and preferences) onto the course presentation for the learner. To be more concrete we define the following sets:

- *Role* – the pedagogical role of the object (e.g. definition, example, demonstration)
- *Style* – the learner’s learning style (e.g. intuitive – sensitive, active – reflective)
- *Media* – the media type (e.g. text, image, audio, video, animation)
- *Context* – the usage context (e.g. multimedia desktop, mobile device)

The proposed adaptation strategy is based on these functions:

- *Weight*: $\text{Role} \times \text{Style} \rightarrow \text{Integer}$
- *Sequence*: $\text{Role} \times \text{Style} \rightarrow \text{Integer}$
- *Alternative*: $\text{Media} \times \text{Style} \rightarrow \text{Integer}$
- *Threshold*: $\text{Style} \rightarrow \text{Integer}$
- *Granularity*: $\text{Context} \rightarrow \text{Integer}$

The *Weight* function represents the relevancy of the pedagogical role for the learning style. The *Sequence* function defines the order for the presentation of the role for the learning style. (Note the difference between these two: an introduction does not have to have the highest relevancy, but when selected it should be the first. The selected components do not have to be ordered according to their relevancy.) The *Alternative* function expresses the relevancy of the media type for the learning style. The *Threshold* function sets the threshold for the object display based on the learning style. The *Granularity* function specifies the maximal number of objects presented at once for the context. The proposed adaptation strategy consists of four operations:

- *Filter*: for the current object it selects just those components that have their *Weight* greater than *Threshold*
- *Order*: this sorts the selected components according to the *Sequence* value
- *Select*: from the alternative components it chooses that one with the highest *Alternative* value
- *Present*: it displays the components taking into account the *Granularity* value

So to define a pedagogic strategy for a certain learning style the instruction designer needs to specify the functional values of *Weight*, *Sequence*, *Alternative*, *Threshold*, and *Granularity* for different types of learning objects (i.e. content objects). But it is not necessary to define all the values. If no value is specified a default one will be applied: 0 for *Weight*, the minimum value for *Threshold* and the maximum one for *Granularity*. The basic operations *Filter*, *Order*, *Select* and *Present* are interpreted by the system. According to their first letters we call this method

FOSP. To implement this method in practice it is crucial to choose suitable visualization of the functional values and an intuitive user interface to control them.

5.5 Adaptivity in ActiveMath

The adaptivity in ActiveMath is based on the learner model which builds on the conceptual items of the content. The learner model is permanently updated using information such as reading of items and exercise results. Based on this learner model the following adaptivity features are provided in ActiveMath:

- Using preferences, the presentation is given an appearance or theme
- The presentation format can be chosen depending on browser's capabilities and learner's choices (currently among HTML, XHTML+MathML, SVG, or PDF). Each presentation code is generated from the semantic encoding. Facilities of each language are invoked to best render it (for example tooltips are output in screen-oriented formats whereas the high-quality layout of LaTeX is used for the print format).
- The learner mostly accesses content through the familiar metaphor of books which are presented as a sequence of pages with a table-of-contents. The links to pages and sections of this table-of-contents are presented with visual mastery hints indicating which page refers to concepts with low, medium, or high user model values. This can be seen as elementary link annotation adaptivity although no navigation restriction is applied.
- Books can be dynamically generated in ActiveMath: they are generated according to the goals of a learner and to a pedagogical scenario. Course generation makes use of the items types, metadata values and relations as well as the user model values so as to select the concepts, content-items, and activities that should be included in a book to learn for this goal. The course generation can be understood as a form of adaptive navigation support. It can be complemented by post-generation user actions to allow modifications to the learner's book considered to be appropriate for her learning (see [U2003]).

5.6 Discussion

We have performed a survey of requirements for professional learning (Appendix A). To summarize, learning needs should be available in a suitable form everywhere and at the workplace it should be seamlessly integrated into the work processes. Learning objectives should involve the whole spectrum from high level skills to concrete pieces of knowledge. E-learning and blended learning are highly demanded by users, taking into account various pedagogical strategies according to the particular objectives and context. Finding a suitable business model for professional learning is a crucial issue, which implies lack of learning resources, as well as their quality, accessibility, flexibility, reusability, and interoperability of learning solutions. Personalization and adaptation of learning is generally considered as highly important because learning has to be individualized to become more effective and efficient.

We have posed 3 questions when comparing the approaches mentioned above in the context of requirements for professional learning. Learner assessment is important in highly contextual workplace learning so we have investigated what the reviewed approaches can deduce about a learner. In particular, all the approaches are based on certain learner performance model but some of them can deal with learner skills and competencies while some just with learner

knowledge items. Another question is which conceptual model has been followed and how it relates to workplace learning performance (activities, learning states, or content). This is also closely related to the adaptation level and how adaptable the solution is. Another important question is which conceptual structures are provided explicitly and which are generated on the fly. Following table summarizes the answers to our questions.

Approach/Characteristic	Knowledge about a learner	Conceptual Model and adaptation level	Presentation Model
IMS LD	Competence and learner knowledge	Learning activities	Explicit – method and environment
WebML/UML-Guide	Competence and learner knowledge	Learning states	Explicit – WebML Hypertext schema and UML navigation guide
ALE	Learner knowledge	Learning objects	Explicit – templates
LAOS	Learner knowledge	Learning goals	Explicit – goals
ActiveMath	Competence and learner knowledge	Learning concepts	Generated – based on planner and book paradigm

The main difference is in the conceptual model employed. As the requirements for professional learning solutions stated, integration with the work processes is required. As the work and business processes are modelled usually by activity graphs, the IMS LD approach is very closely related. Learning states are close as well as at the high level the states recorded in learner performance may trigger the abilities to perform certain work activities. On the other hand, the learning can be designed as guidance through outcomes of particular activities which can be seen as the states of work/learning states on the path. The other three approaches which are based on learning content or concepts can be used as supplemental additional solutions when a content based approach to learning is needed. The conceptual model employed also closely relates which kind of knowledge can be deduced about a learner. In IMS Learning Design, the competencies and skills of a learner can be deduced from properly designed learning objectives while learner's knowledge can be deduced from both, learning objectives of learning activities and knowledge outcomes and subjects of learning objects used within learning environment. Learning states of the UML guide trigger performance records by utilizing references to WebML learning object units and exercises followed. The generality of the conceptual approach in the integrated solution of WebML/UML-Guide allows for design and provision of any kind of virtual learning approach combining content based approaches with skill oriented collaborative learning (which will be especially studied in forthcoming FP6 IST Cooper project). In ActiveMath, the fine-grained competency model for math is employed and connected to mathematical exercises and mathematical conceptual objects. On the other hand, content orientation of ALE and LAOS allows just for recording learner performance in terms of his propositional knowledge reproduced from content he has been visited.

Technically, adaptation approach employed in the reviewed approaches is similar at the high level. It is usually based on conditions which constrain planned learning scenario modelled differently in different approaches. At the low level however, this creates differences in how the adaptation conditions are assigned, whether they are separated or not, whether they are assigned to presentation specifications or they are considered together with content especially when the presentation is generated on the fly.

The major characteristic of LAOS is a clear separation of primitive information (content) - and presentation-goal related information (e.g., pedagogical information in educational systems and prerequisites). For instance, as prerequisites are not hard-wired in the domain model, elements of the domain can therefore be used in different settings and orders than initially intended. In this way LAOS facilitates a high degree of information reuse, by separating information chunks from specific context. This separation is expressed by having two different models, instead of one: a domain model (DM) and a goal and constraints model (GM). The separation can be understood easily if we use the encyclopaedia metaphor: DM represents the book(s) on which the presentation (e.g., a PowerPoint™ presentation represented by the GM) is built. From one book (or DM) one can construct several presentations (here, GMs), depending on the goal. A presentation doesn't contain a whole book, just some (constrained) part of it. Furthermore, a presentation can contain information from several books. The separation therefore gives a high degree of flexibility, based on the DM - GM multi-multi dependency. Another important difference to AHAM is given by the notion of 'concept' used in the domain model. In LAOS concepts have different representations defined via attributes, and are restricted to represent a semantic unity (unlike in AHAM). This is enforced by allowing only self-contained attributes (without direct or indirect dependencies). This setting allows attributes to be flexibly re-ordered. Links are therefore external and can be dynamic.

ActiveMath differentiates itself from many adaptive hypermedia systems by the fact that it encodes the adaptation sources as knowledge written ontologically inside each concepts and items whereas many adaptive hypermedia systems use explicit rules to encode the adaptivity. The limit between conceptual and content items in ActiveMath is somewhat shallow. The authoring processes are, at least currently, the same. This lack of separation makes authoring of "concept content pieces" possible. Examples of such in mathematics (the predilection domain of ActiveMath) include theorems or axioms which are simultaneously conceptual and presented-content items. The separation is visible, however, for knowledge items such as mathematical symbols which are the purely conceptual items on the back of definitions. The learner-model of ActiveMath is based on conceptual items and the relations between them. ActiveMath also differentiates itself also by the granularity of media items: the presentation of ActiveMath items can only support presentation adaptivity (choice of the medium, theme, or notation). ActiveMath's knowledge is, however, built by the metadata although most adaptive hypermedia select content at levels of a resource or bigger, mostly a "page", ActiveMath selects content at the paragraph level. Each of these paragraphs, having pedagogical and mathematical annotations, are the building blocks of the content-selection mechanism. The latter is based on pedagogically sound learning scenarios implemented as a set of HTN-planner operators.

ActiveMath's meta-navigation structures are minimal. Whereas IMS-LD or WebML takes them to high skies, ActiveMath presents the paradigm of a book as main navigation feature, allowing direct dictionary-like browser as accessory. This allows a learner to keep a simple idea of the structure of the content to be learned.

Such table-of-contents can be authored, naturally. But these table-of-contents are also generated and adapted by the content-selection mechanism and can be edited by the so-called assembly-tool.

Considering the level where the adaptivity is specified most system define rules and conditions that take into account user properties for adaptive presentation, sequencing, and interaction. While adaptive LD offers a standard mechanism based on an IMS draft standard, LAOS works on a proprietary basis. In both cases the logic is quite technical to be specified.

The approaches could be considered complementary when combining adaptivity on different levels, while WebML targets a range from selection and sequencing of logical entities in AHA! the adaptivity can be specified with each learning object in detail. On the same level as AHA! IMS-LD works while it splits more between contents and navigational logic. Similar to WebML the ALE System specifies adaptive selection and sequencing rules on an entity level as it states which logical dependencies for sequencing should be taken into account. Compared to WebML ALE takes into account metadata of the learning objects and a learning style model as also a knowledge model to present content. Content adaptivity is focused by AHA! the ALE system and Active Math. Taking into account dynamic curricula generation ActiveMath is probably the most advanced approach, while on the other hand still working with simple prerequisite relations between learning objects.

The FOSP method is based on the experience that authoring of adaptive educational applications is easier if the procedural and declarative knowledge are separated. To support collaborative authoring through reusability of partial results it is also beneficial to separate the procedural knowledge related to instruction, adaptation, and presentation. So the main idea here is to separate different types of knowledge and let them interact.

6 Conclusions

We have concentrated on the authoring and engineering point of view in the specification of adaptive learning environments. The adaptivity based on a user profile is naturally suited for workplace learning, due to its learner and context centric notion which is apparent in adaptive personalized systems for learning. The adaptive systems have not been sufficiently studied in the context of workplace activities. Presented approaches should be studied in integration with business process specification approaches. IMS Learning design seems a good basis for that, as one specification approach which is activity/learning workflow centered. However, it requires integration with business process analysis and specification approaches used in optimization of enterprise performance.

It seems that both engineering and authoring views on specification live together and complement each other. While the engineering focus is more on technology, the authoring focus is more on provided content and pieces of knowledge.

However, one of the main issues in development of advanced technology learning environments is a gap between pedagogues and technicians. An example of the project which tries to deal with this problem is WINDS [KSO2004]. The project has attempted to overcome this gap. Teachers of design and architecture specified their pedagogical requirements to be considered by software developers in the implementation of a unique adaptive learning environment. As the

result, authors without programming skills could create adaptive educational hypermedia courses.

We have pointed to several methods and tools which support specification of adaptation strategies in adaptive hypermedia applications. The key idea of the methods is to simplify the complex authoring process. Collaborative authoring is supported by sharing of partial results between various authors that participate in the development of adaptive hypermedia. This approach is compliant with the established standards and recommendations, including the AHAM reference model. Specification of adaptation strategies by separating the content, declarative and procedural knowledge in adaptive courses is quite natural and similar approaches have been applied in related areas.

7 References

- [AD2004] L. Aroyo and D. Dicheva: The new challenges for e-Learning: The Educational Semantic Web, *Educational Technology and Society* 7(4):59-69.
- [AH1995] W.M.P. van der Aalst and K.M. van Hee: Framework for Business Process Redesign. Proceedings of the Fourth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE 95), IEEE Computer Society Press, Berkeley Springs, 36—45, 1995.
- [B2003] P. Brusilovsky, Developing Adaptive Educational Hypermedia Systems: from Design Models to Authoring Tools, *Authoring Tools for Advanced Technology Learning Environments* (Kluwer, 2003).
- [B2001] P. Brusilovsky, Adaptive hypermedia. User Modeling and User Adapted Interaction, 11(1/2), (2001), 87-110.
- [B1996] S. Billet. Towards a Model of Workplace Learning: The Learning Curriculum. *Studis in continuing education* 18, no. 1 (1996): 43-58
- [BACH2004] P. De Bra, L. Aroyo, and V. Chepegin: The next big thing: adaptive Web-based systems, *Journal of Digital Information* 5(1) (November 18, 2005); <http://jodi.ecs.soton.ac.uk/Articles/v05/i01/DeBra/>.
- [BCCDG2004] S. Buswell, O. Caprotti, D. Carlisle, M. Dewar, M. Gaëtano, and M. Kohlhase: The OpenMath Standard, version 2.0, The OpenMath Society, June 2004, Available at www.openmath.org.
- [BCSB2005] E. Brown, A. I. Cristea, C. Stewart, and T. Brailsford, 2005, "Patterns in Authoring of Adaptive Educational Hypermedia: A Taxonomy of Learning Styles", International Peer-Reviewed On-line Journal "Education Technology and Society", Special Issue on Authoring of Adaptive Educational Hypermedia, July 2005 (Volume 8, Issue 3).
- [C2005] A. I. Cristea: Authoring of Adaptive Hypermedia; Adaptive Hypermedia and Learning Environments. book chapter to appear in "Advances in Web-based Education: Personalized Learning Environments", Sherry Y. Chen and Dr. George D. Magoulas. IDEA Publishing group.
- [C2005a] A. I. Cristea: Editorial: Authoring of Adaptive Hypermedia; appeared in Special Issue on Authoring of Adaptive Hypermedia", International Peer-Reviewed On-line Journal "Education Technology and Society", Special Issue on Authoring of Adaptive Educational Hypermedia, July 2005 (Volume 8, Issue 3).

- [CDMN04] S. Ceri, P. Dolog, M. Matera, W. Nejd: Model-Driven Design of Web Applications with Client-Side Adaptation. Proc. of ICWE 2004, LNCS 3140, pp. 201-214, Springer Verlag, 2004.
- [CDMN05] S. Ceri, P. Dolog, M. Matera, W. Nejd. Adding Client-Side Adaptation to the Conceptual Design of e-Learning Web Applications. *Journal of Web Engineering*, Vol. 4(1), January 2005, pp 21-37.
- [CM2003] A. I. Cristea, A. de Mooij, LAOS: Layered WWW AHS Authoring Model and their corresponding Algebraic Operators, *Proc. of WWW'03*, Budapest, 2003.
- [CISCO2003] *Reusable Learning Object Strategy: Designing and Developing Learning Objects for Multiple Learning Approaches* (Cisco Systems, 2003).
- [CS2005] A. I. Cristea and C. Stewart: Automatic Authoring of Adaptive Educational Hypermedia, book chapter to appear in "Web-Based Intelligent e-Learning Systems: Technologies and Applications", Zongmin Ma. IDEA Publishing group.
- [CSB2005] A. I. Cristea, D. Smits and P. De Bra: Writing MOT, Reading AHA! - converting between an authoring and a delivery system for adaptive educational hypermedia. A3EH Workshop, AIED'05 July, 2005, Amsterdam, The Netherlands.
- [CSAC2005] A. I. Cristea, C. Stewart, H. Ashman and P. Cristea, 2005, "Evaluation of Adaptive Hypermedia Systems' Conversion", HT'05, 6-9 September 2005, Salzburg, Austria, ACM.
- [CSBC2005] A. I. Cristea, C. Stewart, T. Brailsford and P. Cristea, 2005, "Evaluation of Interoperability of Adaptive Hypermedia Systems: testing the MOT to WHURLE conversion in a classroom setting", A3EH Workshop, AIED'05 July, 2005, Amsterdam, The Netherlands.
- [D2003] V. Devedžic: Key issues in next-generation Web-based education, *IEEE Transactions on Systems, Man, and Cybernetics, Part C – Applications and Reviews* **33**(3):339-349, 2003.
- [D2004] V. Devedžic: Education and The Semantic Web, *International Journal of Artificial Intelligence in Education* **14**:39-65, 2004.
- [DHNS2004] P. Dolog, N. Henze, W. Nejd, and M. Sintek: Personalization in Distributed eLearning Environments. In Proc. of WWW2004 - The Thirteen International World Wide Web Conference, New York, USA, 2004. ACM.
- [DHNS2004b] P. Dolog, N. Henze, W. Nejd, and M. Sintek: The Personal Reader: Personalizing and Enriching Learning Resources using Semantic Web Technologies. In Proc. of AH2004 - International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, August, 2004, Eindhoven, The Netherlands. Springer Verlag.
- [DN2003] P. Dolog and W. Nejd: Using UML and XML for Generating Adaptive Navigation Sequences in Web-Based Systems. In Proc. of «UML» 2003 - Sixth International Conference on the Unified Modeling Language: Modeling Languages and Applications, October 2003, San Francisco, USA, Springer Verlag, LNCS 2863
- [FS2000] R.M. Felder and B.A. Soloman. Learning styles and strategies. 2002, Available at: <http://www2.ncsu.edu/unity/lockers/users/f/felder/public/ILSdir/styles.html>
- [GGM2005] G. Gogvadze, A. González-Palomo, and E. Melis: Interactivity of Exercises in ActiveMath, Proceedings of International Conference on Computers in Education (ICCE 2005. In Print.

- [JRL2000] W. L. Johnson and J. Rickel, and J. C. Lester: Animated pedagogical agents: face-to-face interaction in interactive learning environments, *International Journal of Artificial Intelligence in Education* **11**:47-78, 2000.
- [K2001] M. Kohlhasse: OMDoc: Towards an internet standard for mathematical knowledge. In Lozano, E. R. (Ed) Proceedings of Artificial Intelligence and Symbolic Computation, AISC'2000, LNAI. Springer Verlag.
- [KP2001] Rob Koper: Modelling units of study from a pedagogical perspective: the pedagogical metamodel behind EML. 2001. Available at: <http://eml.ou.nl/introduction/docs/ped-metamodel.pdf>
- [KS2004] M. Kravcik and M. Specht: Authoring Adaptive Courses – ALE Approach. *Advanced Technologies for Learning*. 2004
- [KSO2004] M. Kravcik, M. Specht, and R. Oppermann, Evaluation of WINDS Authoring Environment, *Proc. of Adaptive Hypermedia '04 Conf.*, Eindhoven, 2004.
- [LD2003] IMS Global Consortium: IMS Learning Design Specification v. 1.0. 2003. Available at: http://www.imsglobal.org/learningdesign/ldv1p0/imslld_infov1p0.html
- [LD2005] Learning Design: A Handbook on Modelling and Delivering Networked Education and Training. R. Koper and C. Tattersall (Eds.) 2005, Approx 412 p., Hardcover Springer Verlag ISBN: 3-540-22814-4
- [LOM2002] LOM: *IEEE Standard for Learning Object Metadata*. IEEE (2002)
- [MA2003] E. Melis, J. Büdenbender, E. Andrès, A. Frischauf, G. Gogouadze, P. Libbrecht, M. Pollet, and C. Ullrich: Knowledge Representation and Management in ActiveMath. *Annals of Mathematics and Artificial Intelligence*. 38(1-3), 47-64, 2003.
- [MDCG2003] E. Motta, J. Domingue, L. Cabral, and M. Gaspari: IRS-II: a framework and infrastructure for Semantic Web services, in: *Proceedings of The Second International Semantic Web Conference*, Florida, 2003; <http://www.cs.unibo.it/~gaspari/www/iswc03.pdf>.
- [MGHLUW2005] E. Melis, G. Gogouadze, M. Homik, P. Libbrecht, C. Ullrich, S. Winterstein, Semantic-Aware Components and Services of ActiveMath, *British Journal of Educational Technology*, 2005.
- [MKH2005] E. Melis, P. Kärger, and M. Homi: Interactive Concept Mapping in ActiveMath (iCMap), pages 247-258 in Proceedings of Delfi 2005: 3. Deutsche eLearning Fachtagung Informatik, J. Haake, U. Lucke, D. Tavangarian (eds), GI e.V., 2005, LNI 66, ISBN 3-88579-395-4.
- [PDCSA2005] G. Power, H. C. Davis, A. I. Cristea, C. Stewart, and H. Ashman: Goal Oriented Personalisation with SCORM. Proceedings of 5th IEEE International Conference on Advanced Learning Technologies (ICALT 2005), Kaohsiung, Taiwan.
- [RB2005] P. Rosmalen and P. Baticario: Using Learning Design to Support Design and Runtime Adaptation. In R. Koper and C. Tattersall (Eds.), *Learning Design: A Handbook on Modelling and Delivering Networked Education and Training*. Springer Verlag, p.291-301, 2005.
- [RB2000] R.J. Riding and C. F. Buckle. Learning styles and training performance. Sheffield: Training Agency, 1990.
- [RR1995] R. J. Riding and S. Rayner. The information superhighway and individualized learning. *Educational Psychology* (1995), 15 (4), 365-378

- [RSS1999] W. Rothwell, E. Sanders, and J. Soper: *ASTD models for workplace learning and performance*. Alexandria, VA: The American Society for Training and Development. 1999.
- [SCBA2005] Stewart, A.I. Cristea, T. Brailsford and H. Ashman: 'Authoring once, Delivering many': Creating reusable Adaptive Courseware. WBE'05 February, 2005, Grindelwald, Switzerland.
- [SCB2005] N. Stach, A.I. Cristea and P. De Bra: Explicit Intelligence in Adaptive Hypermedia: Generic Adaptation Languages for Learning Preferences and Styles. International Workshop on Combining Intelligent and Adaptive Hypermedia Methods/Techniques in Web-Based Education Systems, HT'05, 6-9 September 2005, Salzburg, Austria, ACM.
- [SKPK2001] M. Specht, M. Kravcik, L. Pesin, R. Klemke: Integrated Authoring Environment for Web Based Courses in WINDS, *Proc. of the ICL 2001 Workshop*, Villach, 2001.
- [SKPK2001a] M. Specht, M. Kravcik, L. Pesin, R. Klemke: Authoring Adaptive Educational Hypermedia in WINDS, *Online-Proc. the ABIS 2001 Workshop (Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen)*, Dortmund, 2001.
- [SKKPH2002] M. Specht, M. Kravcik, R. Klemke, L. Pesin, R. Hüttenhain: Adaptive Learning Environment for Teaching and Learning in WINDS, *Proc. of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems*, Malaga, 2002.
- [SKKPH2002] M. Specht, M. Kravcik, R. Klemke, L. Pesin, R. Hüttenhain: Adaptive Learning Environment in WINDS, *Proc. of the ED-MEDIA 2002 Conference*, Denver, 2002.
- [SKKP2002] M. Specht, M. Kravcik, R. Klemke, L. Pesin: Information Brokering for the Adaptive Learning Environment, *Proc. of the e-Learn 2002 Conference*, Montreal, 2002.
- [TH2005] B. Towle and M. Halm: Designing Adaptive Learning environments with Learning Design. In R. Koper and C. Tattersall (Eds.), *Learning Design: A Handbook on Modelling and Delivering Networked Education and Training*. Springer Verlag, p.215-226, 2005.
- [TPG2002] E. Triantafillou, A. Pomportsis and E. Georgiadou. AES-CS: Adaptive Educational System base on cognitive styles. In AH2002 Workshop (Malaga, Spain), 10-20.
- [WMGC1977] H. A Witkin, C. A. Moore, D. R. Goodenough, and P. W. Cox. Field-dependent and field-independent cognitive styles and their educational implications, *Review of Educational Research*, 47(1), 1977, 1-64.
- [U2003] C. Ullrich. An instructional component for dynamic course generation and delivery. In R. Tolksdorf and R. Eckstein, editors, *Proc. of Berliner XML Tage 2003*, pages 467–473, 2003.
- [ULWM2004] C. Ullrich, P. Libbrecht, S. Winterstein, and M. Mühlenbrock: A Flexible and Efficient Presentation-Architecture for Adaptive Hypermedia: Description and Technical Evaluation. Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies. 21-25, 2004.
- [VD2004] K. Verbert and E. Duval: Towards a Global Component Architecture for Learning Objects: A Comparative Analysis of Learning Object Content Models. Proceedings of the ED-MEDIA 2004 Conference. AACE (2004)

Appendix A. Survey of Requirements of Professional Learning

In the time interval between October 26 and December 2, 2005 we have received 42 responses from 17 countries to our questionnaire. Most of the respondents were from Europe (one third of all from Germany), 4 came from America and one from Africa. Some respondents entered more than one country as localization of their organization. Almost all the respondents were older than 25 and younger than 65 years, and in these limits younger people were superior in number. One half of all respondents worked in academic institutions, 7 in research organizations and private companies respectively. One half of the respondents' organizations had the main activity in education and training, more than one third in the ICT area. The organizations were predominantly larger, with more than 100 employees, almost one third of them with more than 1000 people. Two thirds of the respondents were professors and researchers, 7 were occupied by management. Most of the people ranked their ICT expertise as high. Their educational background was predominantly postgraduate and postdoctoral.

1. In which settings, environments and situations you want to perform learning?

To perform learning people want to use both traditional live methods (lectures, seminars and labs) and e-learning facilities (such as learning management systems, video conferences, simulations, games) to explore technical, social, cultural, and natural environments. It should be possible everywhere – at work, at home, but also when travelling or waiting, during breaks, with ubiquitous access, both planned and on demand. Learning has to be personalized and self responsible, considering both theoretical knowledge and practical skills. Various educational methods need to be supported, including individual study, collaborative learning, assignments and assessments. A supporting environment and suitable atmosphere should be created – pleasant, relaxed, open, and quiet, according to the situational demands.

At the workplace learning should be seamlessly integrated into the daily work and without severely interruptions of the work processes, enabling collaboration in expert groups. At home learning should add to the existing skills and fulfil leisure time. Learning should happen both in formal and informal learning situations, supported by distributed virtual environments. It has to be independent of time and place, suitable to each own working schedule, including weekends or one day a month but very intensive.

One respondent has replied: "I would like to learn at my working place, without considerable time spent on participating at lectures, exercises etc. The optimal instruction at the working place will concern learning to solve specific problems I am encountering here. The information about the solutions should be easily available, and preferably searchable using standard tools without involving third persons (which would take too much time)."

2. What are the main objectives (like development of certain learner's skills and competencies) you want to achieve by learning?

By learning respondents want to achieve peak professional performance of individuals and teams, to support staff, and enable self responsible learning for self development. Knowledge workers should be able to get aware of newest trends in their field and to fill their knowledge gaps acquiring new competences in a rapidly evolving context. Skills like critical thinking, creativity, team competences, ability to acquire new knowledge, problem analysis, application of theory into practice, ability to act in new environments, language, communication and presentation skills, conflict management, research, administrative, financial, business and

management skills, as well as computer proficiency are requested. New skills should comply with the job demands and add to the current competencies for fun, e.g. languages. Major objectives include learner's ability to be integrated in the industry and to manage new technologies. Learning should contribute to the engineers' skills in installing, operating, and servicing the companies' products.

Respondents want to find efficient and cheap methods of learning content supply, which do not involve third parties as well as efficient and simple collection of data from distance learners. They would like to support their personal development, to enrich their self consciousness, to improve their professional standing, to stay up-to-date in their areas of interest, and to look up details of methods they intend to apply. Understanding and managing artefacts in dynamic working and living environments is of great importance.

3. Which delivery methods you want to use? (Multiple selections allowed)

Classroom learning (face-to-face)	9
E-learning (computer-mediated learning)	24
Blended learning (combination of both)	23
Other	5

Most of the respondents want to use e-learning and blended learning, but classroom learning is occasionally demanded as well. Other methods are not so much requested – they include on demand learning for specific work tasks, one day intensive face-to-face (or video conferencing), practice-based learning, projects, literature, and conferences.

4. Which pedagogical strategies you want to apply? (Multiple selections allowed)

Direct instruction (lecture, drill and practice, demonstration)	17
Indirect instruction (problem solving, case study, concept mapping)	22
Experiential learning (simulation, game, experiment, field trip)	22
Independent study (computer assisted instruction, report, homework, project)	19
Interactive learning (debate, role playing, brainstorming, cooperative learning)	25
Other	4

Although interactive learning is slightly more preferred by our respondents than direct instruction or independent study, each of the five different pedagogical strategies finds its place, depending on the particular objectives and contexts. Also other strategies have been mentioned, e.g. formal learning.

5. Which software systems are you using to support learning and how are you satisfied with them? (Minimum 1, Maximum 5)

Content management systems	21	3.1
Learning management systems	23	3.1
Corporate training systems	8	2.1

Enterprise resource planning	7	2.7
Other	7	

To support learning most of the respondents use learning and content management systems while their satisfaction with them is average. Some people consider Google and the Internet itself as systems supporting learning, but they are even less satisfied with these solutions. On the other hand, there were also individuals that were very satisfied with their systems: namely with collaboration and workspace software, process engine, MOO, and specialized software (e.g. for simulation).

6. Which categories of employees are involved in your learning offerings? (Multiple selections allowed)

Management	12
Sales force	5
Administrative staff	7
Engineers	15
Specialized workers	11
Non-specialized workers	1
Other	10

From several various categories of employees most often engineers, management, and specialized workers were involved in learning offerings.

7. Who decides about the learning plan of an employee? (Multiple selections allowed)

Employee himself/herself	21
His/her manager	15
Human resource manager	6
Other	4

The employee himself or herself decides usually about his or her learning plan, in some cases the manager participates in this decision as well.

8. What are the main issues you encounter in learning deployment?

At the organizational level the change takes a long time, there is a lack of support from managers and the organization. A “paradigm shift” in the mind of decision makers has to happen. Organisational costs of learning (e.g. training time, training infrastructure, travel costs) is high as well as development time for training on new products as training development SMEs are not available. Companies indicate the sectors that need more technological education. It is not easy to adjust teaching to the necessities of the industry and of the society. Integration of e-learning and blended learning concept into teaching according to the Bologna-context (curricular level) is difficult.

There is a lack of learning resources, their flexibility and reusability for various situations. Quality and accessibility of useful materials is not appropriate. Students prefer paper materials for

learning. They do not want spend time in front of a computer. E-learning is not always user friendly, uses legacy applications, and lacks pedagogical concepts. Its organisation is not enough reflecting and supporting learning processes of semi-administered students. There is also a problem of efficient examining (so that no one cheats). Lack of interoperability and integration between learning systems and knowledge management solutions is another issue.

Personalization of learning resources to the learner's needs, preferences, skills, knowledge, and context is needed. It is not enough specialized, learners need to get the training quickly to answer a precise requirement that has arisen. Language is sometimes a barrier as most references are in English. Most students lack some skills like logical thinking because many are used to reciting. There are motivation differences in learner identities that should be supported in various learner cultures. Motivation and discipline are issues for the learner in self-administered and semi-administered settings.

9. Do you consider personalization and adaptation of learning materials (experiences) to the needs of the user and to the current context as important factors in your learning solutions? Why?

A high majority of respondents considers personalization and adaptation of learning as important and crucial factors, just one does not, one partially for lack of available resources, and one only if these solutions support collaborative learning, because otherwise learning is reduced to passive training. The main reasons for positive responses are that learning should be individualized to become more effective and efficient, personalization is the key element of the learning process, and specific problems need specific solutions, as students differ greatly in their background and capabilities especially in the field of computers. Learning materials are typically too general to cover a very wide range of purposes, so personalization can be the most important added value that elearning can offer compared to classical learning – to optimise education, to adjust to various working conditions and needs, because students (academic and corporate) have different goals, interests, motivation levels, learning skills and endurance. Two responses follow:

“Yes, that is exactly what we can support and the only right way to integrate learning and working. It fits with the way 'Digital Natives' are learning today.”

“Personalization and adaptation of learning material is important, but not feasible at the moment, due to the lack of resources (designers). At this moment there are not enough resources to develop standard-form (not personalized) material of good quality. Personalized content is expensive in development. And it is not clear what to personalize (focus on material or on navigation). We would like to personalize electronic material we are using, by introducing modification of content by students (adding bookmarks, marking text, adding comments). But existing tools do not offer such functionalities, either are very expensive, or do not work with common document standards (pdf, doc, html). For obvious reasons personalization includes integration with databases and server applications. We do not know any good solutions offering the personalization of public documents described above.”