Supporting Personalized Learning Experiences within the LOGOS Cross-Media Learning Platform

Polyxeni Arapi[1], Nektarios Moumoutzis[1], Manolis Mylonakis[1], George Stylianakis[1], and George Theodorakis[1]

[1] Technical University of Crete, Laboratory of Distributed Multimedia Information Systems and Applications (TUC/MUSIC), 73100 Chania, Greece
{xenia, nektar, manolis, gstylianakis, gtheodorakis}@ced.tuc.gr

Abstract. LOGOS project provides a Knowledge-on-Demand ubiquitous learning platform that supports the gradual development of courseware and its cross-media delivery as learning experiences in order to satisfy different learning needs. Special attention is given to provide pedagogically-sound personalization services which can be exploited both by Learners directly and courseware authors in order to develop courseware in a semiautomatic way. This paper presents the overall architecture of LOGOS platform and describes in detail the components related with the dynamic creation of pedagogically-sound personalized learning experiences.

Keywords: Learning Objects, Personalization, Learner Profiles, Instructional Design

1 Introduction

The vision of ubiquitous learning (u-learning) is to provide effective personalized learning services to support learning anywhere, anytime exploiting alternative delivery channels and related devices that go beyond the traditional web-based learning approaches. Towards this end, the exploitation of the wealth of available content residing at content archives and digital libraries is very important in order to provide cost-effective production of learning content and give added value to existing content.

Within the LOGOS project, a Knowledge-on-Demand ubiquitous learning platform is being developed in order to bring the u-learning vision into reality. The Platform consists of layered repositories supporting the gradual creation of learning experiences starting from existing content residing at multimedia archives. An Authoring Studio of tools provides all the necessary functionality for learning content creation and supports authoring tasks for certain user roles. Cross-media delivery of learning experiences integrating web-based, mobile and digital TV technologies is handled by special Learning Management System components and publishing services.

LOGOS pays particular attention to the issues related to personalization recognizing that “one size fits all” solutions are no longer enough to satisfy the

Learners' educational needs. Different Learners have different learning styles, educational levels, previous knowledge, technical and other preferences and all these are parameters that affect the learning function outcome. Learners expect from systems a "personal trainer" and not a "classroom" behavior, where their personality and needs are known and taken into account. Moreover, the proliferation of the Internet and the wealth of content in Learning Object Repositories call for flexible solutions where content is not strictly bound with the learning plan but could be retrieved at run-time and ideally from many sources according to the Learner needs. This is called in Adaptive Hypermedia "open corpus". Several research areas are related with the above challenges: Adaptive Hypermedia Systems, Intelligent Tutoring Systems, and Semantic Web [4]. Although each area treats adaptivity of learning experiences from a different point of view, there is a convergence in the research community that pedagogy is important and should be represented in a consistent way. Moreover, the pedagogical model should be reusable and separated from content allowing appropriate learning resources according to the Learner profile to be bound to the training scenario at run-time. LOGOS project provides and implements a framework for the support of pedagogically-sound personalized learning experiences on top of its repositories. These personalization services can be exploited either by Learners (directly) or by courseware authors for the semiautomatic creation of courseware.

The structure of the rest of this paper is: Section 2 presents the architecture of the LOGOS platform. Section 3 describes in detail the components of the LOGOS platform related with the dynamic creation of personalized learning experiences. A review of the related literature is presented in Section 4 and the paper ends with some concluding remarks and future work (Section 5).

2 The architecture of the LOGOS platform

LOGOS platform, depicted in Fig. 1, integrates:

- Appropriate repositories and services for the management of various types of objects:
  - **media objects** coming from external content archives,
  - **digital objects** created on top of media objects that correspond to media objects or parts of them annotated and indexed with administrative and semantic metadata,
  - **learning objects** built on top of digital objects and enriched with educational metadata, and
  - **courseware objects** utilizing the underlying learning objects and corresponding to learning experiences that can be delivered using different delivery devices.
- An **Authoring Studio** that consists of tools for the creation and editing of the above types of objects as well as for the creation of abstract training scenarios (Learning Designs) in order to support the dynamic creation of personalized learning experiences and for the publishing of courseware objects to different delivery platforms.

– **Learning Management System components** for the delivery of courseware to Learners encapsulating functionality to adapt the learning material to individual user needs and context as well as to track user's progress and update the user related information represented in Learner Profiles.

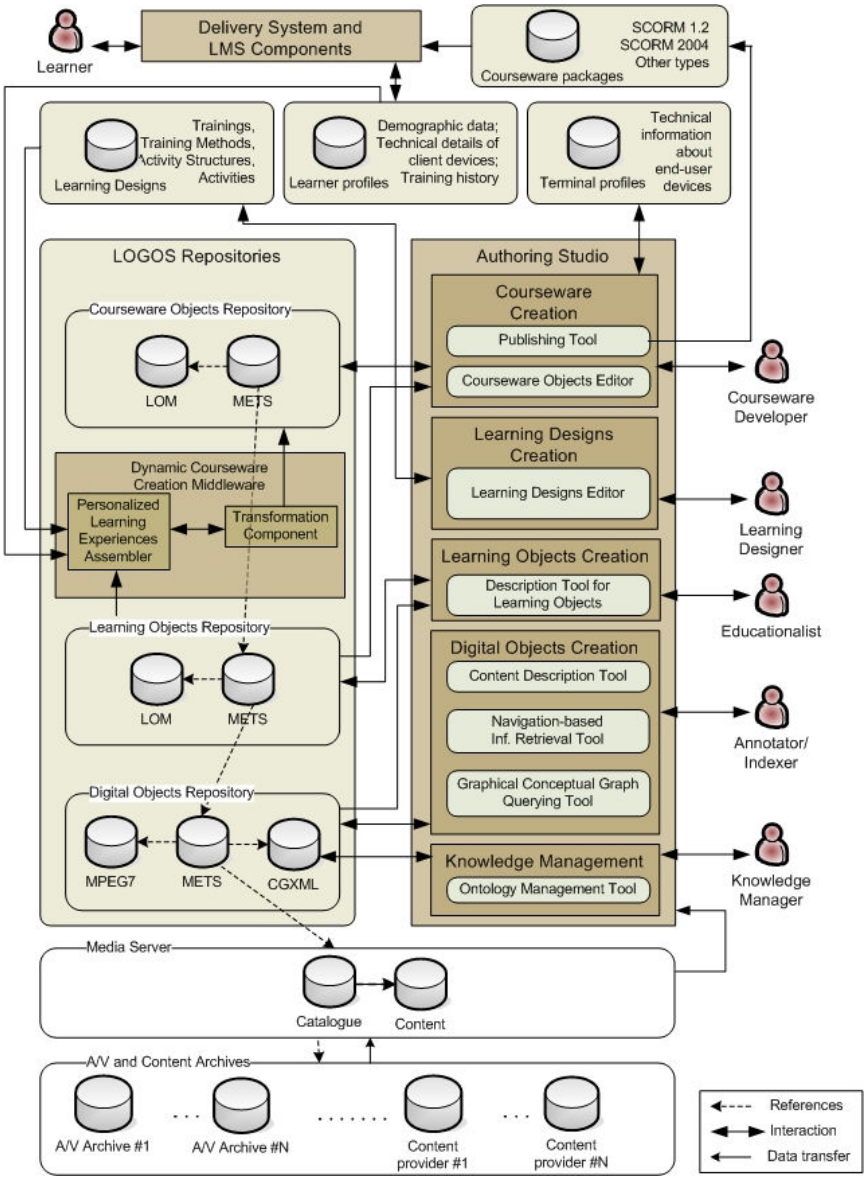

Fig. 1. Overall architecture of the LOGOS platform.

In Fig. 1 the overall architecture of LOGOS platform is depicted focusing mainly on the components that relate to LOGOS Repositories and Authoring Studio and how

they communicate in order to provide the necessary functionality for the authoring of cross-media learning material. The various user roles engaged are also depicted. In the following subsections we give more details about these components in order to provide a clear view of the architecture.

2.1 LOGOS Repositories and Media Server

The authoring process envisioned in LOGOS starts from objects residing at external archives (media objects). They represent interesting material that can be used for the creation of learning material to be delivered to the Learners. Media objects are managed by the Media Server that offers all the necessary functionality to store, search and retrieve them. On top of media objects, other types of objects are created and managed by LOGOS repositories by gradual synthesis of higher level objects and addition of appropriate semantics and metadata. Apart from media objects, there are three basic types of objects managed by the corresponding LOGOS repositories and used by LOGOS Authoring Studio Tools:

- **The Digital Objects Repository manages Digital Objects (DOs),** which are annotated and indexed media objects or parts of media objects that are used to create learning material. DOs are annotated with administrative and semantic information so that they can be readily available to be used as building blocks for the creation of learning material.
- **The Learning Objects Repository manages Learning Objects (LOs),** which are independent and self-standing units of learning content that are predisposed to reuse in multiple instructional contexts. They are collections of DOs enriched with educational metadata. Special types of LOs are assessment objects that contain assessment items.
- **The Courseware Objects Repository manages Courseware Objects (COs),** which are structured objects that can be used to provide individual (personalized) or collective learning experiences in formal or non-formal learning settings. They are hierarchies of LOs including additional educational metadata and possibly sequencing and navigation metadata.

Between the LOs repository and the COs repository there is a **middleware** for the automatic construction of personalized courseware. The middleware is able to exploit information stored in a Learner Profile in order to create specific COs for him/her. In order to do so, the middleware uses abstract training scenarios, called Learning Designs that contain instructional methods of how to teach certain subjects. Essentially, the middleware finds training scenarios that meet the needs of the Learner and then proceeds to fill in the nodes of these scenarios with specific reusable LOs that satisfy the learning goals and preferences of the Learner.

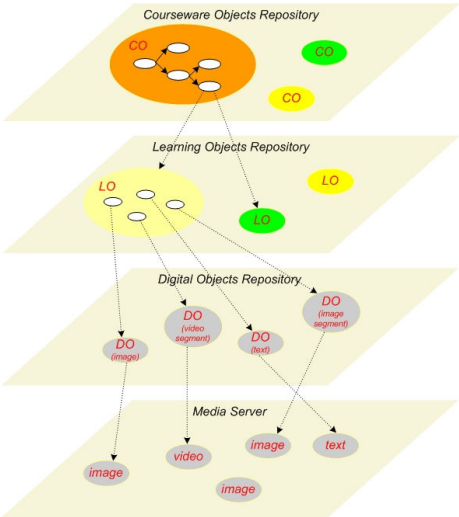

Fig. 2. Relation among COs, LOs, DOs and Media Objects residing in corresponding repositories and the Media Server.

2.1.1 Representation of Digital Objects, Learning Objects and Courseware Objects using METS

In order to support the gradual development of courseware objects a flexible model is needed in order to allow for:

- Integrated description of objects at each level using several appropriate (metadata) schemes to represent the different aspects of objects.
- References to objects residing at lower levels without repeating their information at the current level. Generally, objects residing at a certain level should be able to reference objects at the level underneath. This way reusability of objects at lower levels is supported from the upper levels. Moreover, this flexible representation of objects allows for appropriate adaptation/transformation of objects at run-time in order to support cross-media delivery of learning experiences.

A flexible model that satisfies the above needs is the Metadata Encoding and Transmission Standard (METS) [11]. METS is a widely-accepted standard designed specifically for digital library metadata. METS is a flexible, but tightly structured, container for all metadata necessary to describe, navigate and maintain a digital object: Descriptive, Administrative and Structural metadata. Each type of metadata is described in a separate section, which is linked to its counterparts by internal identifiers. These metadata (any preferred scheme) may be physically stored within the METS file, or in external files referenced from within the METS document. METS has been already used in [1] where an interoperability framework between digital libraries and eLearning applications was introduced.

Fig. 3 illustrates how the features of METS are exploited in order to describe objects at the DO, LO, and CO levels:

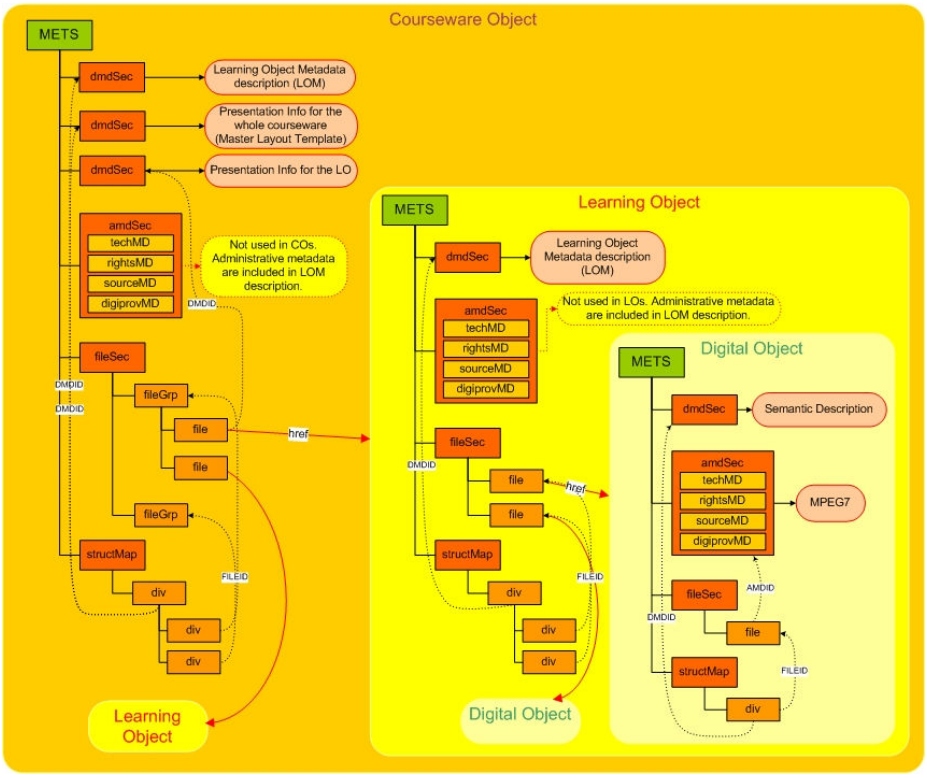

Fig. 3. COs, LOs and DOs and their relations described with METS

- DOs are described both with semantic information and administrative information. The descriptive metadata section (dmdSec) of METS is used to incorporate semantic description expressed with CoGXML (a representation format for conceptual graphs[1]) and the administrative metadata section (amdSec) in order to incorporate administrative metadata expressed with MPEG7 [13]. The fileSec is used to point to the parent media object residing at the Media Server, from which the current DO has evolved.
- LOs are described with IEEE LOM using the dmdSec. Since LOM incorporates in its model entries for administrative information, the amdSec of METS is not separately used in this case for representing administrative metadata. The fileSec consisting of file elements is used to point to the LO's constituent parts (DOs) via identifiers. Assessments are considered as a special type of LOs and are also described with LOM metadata. While LOs containing DOs are built to fulfill specific learning objectives, assessment-type LOs are used to evaluate learning

[1] http://conceptualgraphs.org/

objectives. IMS QTI descriptions are used for the representation of assessments that are referenced from the METS description.

- COs are also described with LOM via the dmdSec of METS. The structMap section is used to represent the CO's structure consisting of a hierarchy of activities (expressed with div) that can take place during the learning process using multiple devices. Each activity (div) points to a group of LOs organized through fileGrp element. Each one of those LOs is considered as an alternative form of educational content supporting the corresponding activity and intended for a particular collection of devices. The file element is used again in order to reference LOs residing at the LO repository via identifiers. A CO as a whole and its constituent LOs could reference using dmdSec elements some presentation info that is exploited at run-time to render the learning material in the target devices.

2.2 Authoring Studio

The Authoring Studio provides authors of learning materials with an extensive set of tools to enable the creation of multi-platform courseware materials that may be distributed to several end-user devices, including PCs, mobile phones or even Interactive Television (iTV). The Studio includes tools for pre-processing of digital textual, graphical, audio and video objects from existing archives, supporting format transformation, segmentation, indexing, annotation, semantic description, etc.

The versatile pre-processing of the available digital objects allows the users - authors of learning materials, lecturers and/or advanced Learners - to achieve a good personalization level of the produced courseware, considering the learning context - Learners aims, stimuli and interests, educational and cultural background, learning place/s, learning styles, course organization, course duration, etc.

The web-based versions of courseware, produced by means of the authoring studio, are formed according to international learning standards (e.g. SCORM) in order to be applicable for different Learning Management Systems (LMS).

The basic components (tools) of the Authoring Studio along with their core functionality are the following:

- **Ontology Management Tool**: This tool provides functionality for the creation of domain ontologies based on the conceptual graph formalism. These ontologies are used to create semantic annotation in digital objects.
- The **Content Description Tool**: This tool provides all the necessary functionality to segment, index and semantically annotate digital objects residing at content archives. It is essentially integrated with the **Graphical Conceptual Graph Querying Tool** to perform searches on the existing digital objects and selects the appropriate ones for additional annotation processes.
- **Navigation-based Information Retrieval Tool**: Apart from the Graphical Conceptual Graph Querying Tool, this tool is going to provide additional functionality for searching for digital objects exploiting an incremental navigation-based approach to the information retrieval process. This approach allows users to formulate very general questions and navigate through structured answers (like a table of contents). This process aims to present a general but structured view of the possible audiovisual material upon a given subject.

- The **Description Tool for Learning Objects**: This tool essentially provides the means to create reusable learning objects from digital objects and describe them with educational metadata.
- The **Courseware Objects Editor**: This tool provides functionality to combine reusable learning objects in order to synthesize courseware for addressing the learning needs of Learners. Courseware synthesis incorporates the selection of reusable learning objects exploiting their educational metadata, structuring them in appropriate hierarchies and specifying alternative navigation paths so that multiple devices could be used for the delivery of created learning experiences.
- The **Publishing Tool**: It provides functionality to publish courseware objects into appropriate formats so that they can be delivered in different devices (PCs, mobiles, digital TV).
- The **Learning Designs Editor**: The Learning Designs Editor is a special tool of the Authoring Studio that allow for the creation of abstract training scenarios that can then be used to support the automatic creation of personalized courseware.

3 Dynamic creation of personalized learning experiences

Personalization is a major concern within the LOGOS project and LOGOS platform, as described in the previous section, incorporates all the necessary components in order to effectively support the creation of personalized learning experiences. In particular, LOGOS follows a pedagogy-driven approach to personalization where pedagogy is clearly separated from the content and encoded in Learning Designs. The Learning Designs Editor is used for the creation of abstract training scenarios based on a certain instructional ontology. The middleware in LOGOS repositories makes use of these abstract training scenarios in order to create personalized learning experiences that match Learner needs and preferences encoded in Learner Profiles. This section described in detail the LOGOS platform components supporting personalization.

3.1 Learning Designs

Learning Designs are abstract training scenarios that are constructed according to an instructional ontology coded in OWL (Figure 3). This ontology has the important characteristic that learning objects are not bound to the training scenarios at design time, as in current eLearning standards and specifications (e.g. IMS Learning Design - IMS LD - and SCORM). Whereas, pedagogy is separated and independent from content achieving this way reusability of Learning Designs or parts of them that can be used from the systems for the construction of “real” personalized learning experiences, where appropriate learning objects according to the Learner Profile are bound to the learning experience at run-time taking into account several parameters of the Learner Profile. This is possible, since the model gives the opportunity to specify in each Activity the learning objects’ requirements, instead of binding the learning

objects themselves. This ontology exploits some elements and ideas from IMS LD and LOM.

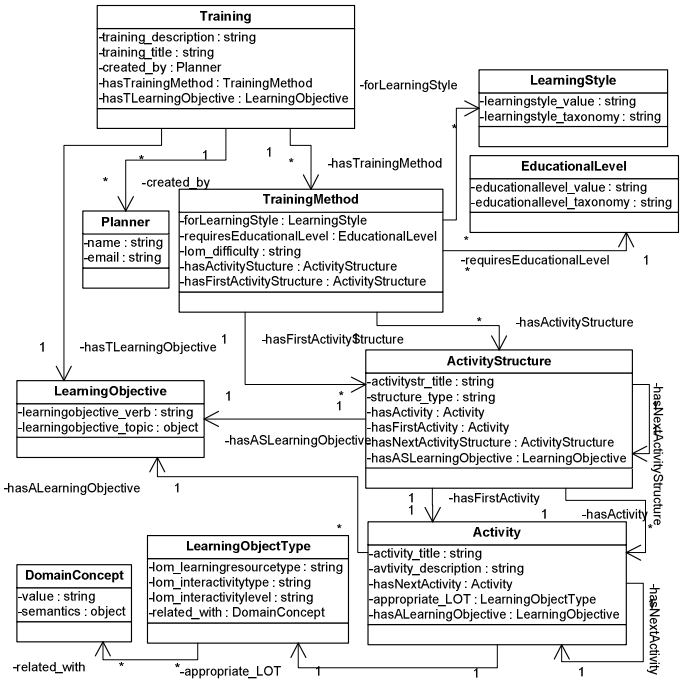

Fig. 4. The instructional ontology used to represent Learning Designs.

A *Training* is a collection of *TrainingMethods* that refer to the different ways the same subject can be taught depending on the *LearningStyle*, the *EducationalLevel* of the Learner and the preferred *Difficulty*. There are several categorizations of *Learning Styles* and *Educational Levels*, thus these elements are flexible so that being able to point to values of different taxonomies. A *TrainingMethod* consists of a hierarchy of reusable *ActivityStructures* built from reusable *Activities*. Each *Training*, *ActivityStructure* and *Activity* has a *LearningObjective*. Usually, simple textual descriptions are used to describe Learning Objectives. However, this approach does not represent a formal way for defining Learning Objectives presenting a technical barrier because textual descriptions are not machine-readable and can not be exploited by automatic personalization processes. To address this shortcoming we define a *LearningObjective* as: (a) a learningobjective_verb, taken from a subset of Bloom's Taxonomy [3]) and (b) a learningobjective_topic that indicates the topic that the Learning Objective is about, referencing a concept or individual of a domain ontology. The *LearningObjectType* is used to describe the desired learning object characteristics without binding specific objects with *Activities* at design time. Via the *related_with* property we can further restrict the preferred learning objects according to their constituent parts (if they are semantically annotated) connecting them with *DomainConcepts* which refer to concepts or individuals from a domain ontology.

3.2 Learner Profiles

In order to support personalization within the framework presented in this paper several parameters in Learner Profiles are considered as important. These elements could be mapped in appropriate elements of the IEEE Personal and Private Information (PAPI) and IMS Learner Information Package (LIP) using extensions. We focus on the Learner's goals and preferences and we illustrate those elements and their relations in a Learner Ontology (Fig. 5). It should be noted that we do not describe here all the parts of a Learner Profile; we just focus on what is important for the dynamic creation of personalized learning experiences.

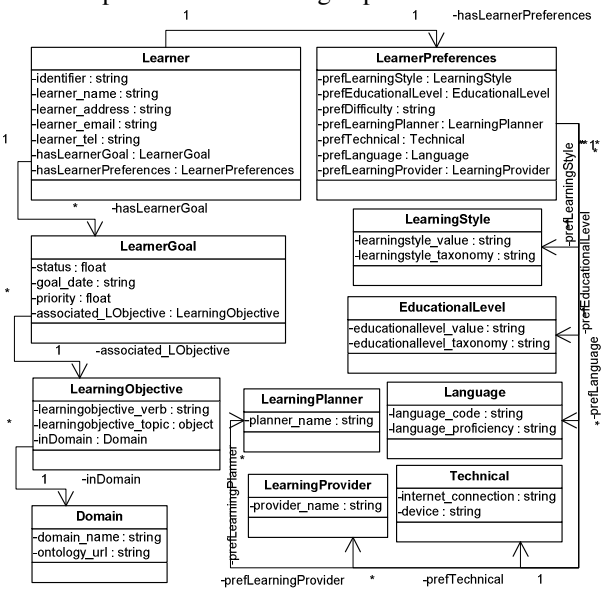

Fig. 5. Learner Model classes related with the dynamic creation of personalized learning experiences.

A LearnerGoal is expressed in terms of LearningObjectives using the structure that was presented above in the instructional ontology. A Learner can have many LearnerGoals. A LearnerGoal has a status property (float in [0, 1]) indicating the satisfaction level of the goal (0 represents no satisfaction, 1 fully satisfied). Using this information one can also infer the previous knowledge of the Learner. The Learner can also define a priority for each LearnerGoal. The Learner can have several types of Preferences: EducationalLevel and LearningStyle matching with the corresponding elements of the instructional ontology, Language, LearningProvider (the author or organization making available the learning objects), LearningPlanner (the person that develops Learning Designs) and Technical preferences.

As it is described in the next subsection, these parameters affect both the construction of an appropriate learning path for a specific Learner according to existing Learning Designs and the selection of learning objects that are thereafter bound at run-time to the learning path to form the resulting learning experience.

3.3 The Personalization Component (middleware)

The Personalization Component takes into account the knowledge provided by the Learner Profiles and the Learning Designs and constructs personalized learning experiences that are delivered next to the Learners. Specifically, the goal is to find an appropriate Learning Design that will be used thereafter to construct a learning experience adapted to the Learner's needs. As already mentioned, learning objects are bound to the learning scenario at run-time.

The procedure of constructing an adaptive learning experience is illustrated in Fig. 6. In each step several parameters of the Learner Profile (given in brackets in Fig. 6) are taken into account:

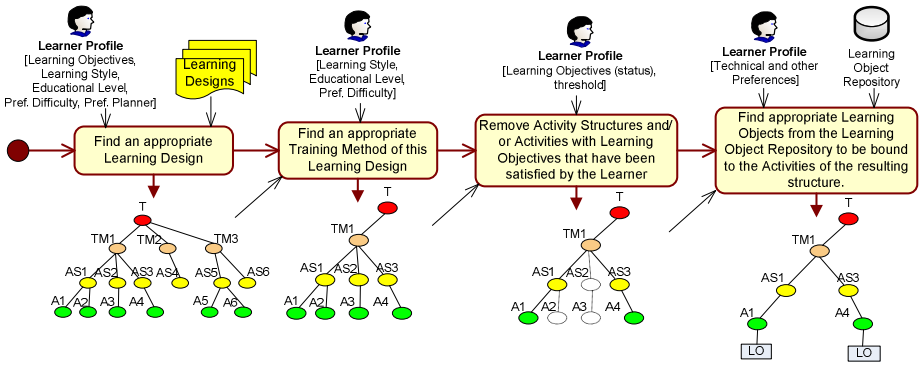

Fig. 6. The procedure of dynamic construction of personalized learning experiences.

1. At the beginning, the component tries to find an appropriate Learning Design (Training in terms of the instructional ontology presented) taking into account the Learner's Learning Objectives, Learning Style, Educational Level, preferred Difficulty, and preferred Planner (optional).
2. When an appropriate Learning Design is found its structure is retrieved (Training(T), Activity Structures (AS), Activities(A)) and an appropriate Training Method of this Learning Design is selected, according to the Learner's Learning Style, Educational Level and preferred Difficulty.
3. The structure of this Training Method is further refined, by removing from it Activity Structures and Activities with Learning Objectives that have been satisfied by the Learner (the Learner can define a threshold value t, so that Learning Objectives with satisfaction value>t are considered as satisfied).
4. Finally, appropriate learning objects are retrieved and bound to each node (Activity) of this structure constructing the learning experience. Here, the Learning Object Type describing the characteristics of appropriate learning objects for each Activity is taken into account along with other Learner's preferences (e.g. content provider, technical preferences). The resulted learning experience is initially stored as a Courseware Object represented in METS format and the Learning Management System components in cooperation with the Publishing Tool are responsible to deliver it in appropriate format to the Learner.

5 Related work

The issues related to the overall process of learning material production especially for cross-media learning experiences, are many and the recent years various research efforts are made to address them from various points of view. The EXPLAIN project [8] builds a web-based Authoring Management System that supports the creation of eLearning content on the whole and allows for the integration of existing authoring tools. It includes project management, content modeling (based on the book paradigm) as well as resource and material management. Although it does not explicitly support authoring of cross-media learning resources, it represents a complementary approach to LOGOS, as presented in this paper that could address many issues related to the cost-effective development of learning material. LOGOS will investigate such issues during its trial & refinement phase. In [12] a document-oriented production process model is presented for the creation of learning contents. The model makes use of markup languages for different domains and supports retargeting of materials produced. A manual-writing metaphor is used based on DocBook markup language for technical documentation for the production of learning contents in Computer Science. External tools are employed for material preparation. LOGOS follows a similar approach but instead of using domain specific markup languages, relies on domain ontologies for the appropriate semantic description of content used to create learning materials. LOGOS layered approach to content creation is more general and facilitates the re-use of materials at different levels supporting also effective personalization in the construction of learning experiences. [9] presents an interesting approach to support t-learning by using and extending SCORM. It builds an MHP-based SCORM Runtime Environment (RTE) able to present SCORM-based learning materials on Interactive Digital TV receivers. No focus is given to learning experiences intended for other types of devices and thus the proposed SCORM extensions suffice for the objective to have a dedicated t-learning platform. LOGOS follows a more general approach and explicitly addresses cross-media related issues. LOGOS is also building extensions to standards (e.g. LOM extensions to support the definition of technical requirements for different devices). However, it does not build special SCORM-based RTEs for different delivery platforms but relies on a general publishing framework.

LOGOS pays particular attention to personalization in order to provide effective learning experiences tailoring to the needs and preferences of individual Learners. It adopts a framework that clearly separates pedagogy from content in order to exploit reusability of abstract training scenarios in various learning situations. In [5] a similar approach is followed to represent pedagogy in order to support run-time resource binding. Our approach differs in that it takes into account the learning style, the educational level and learning goals of the Learners, supporting the representation of different learning paths (Training Methods) for training in a specific subject. In [10], although the need for supporting different training methods for the same subject is recognized, these methods are not connected as in our approach with the learning styles and educational levels of the Learners. Moreover, description of appropriate learning objects characteristics beyond semantics is not supported. An alternative approach is presented in [6] regarding automatic course sequencing. In that work learning paths are not constructed based on pedagogical models, but are extracted

from a directed acyclic graph that is the result of merging the knowledge space (domain model) and the media space (learning objects and their relation) using minimum learning time as an optimization criteria. However, since this approach is highly based on the domain model that does not necessarily imply an instructional model, and also on the relations of learning objects and their aggregation level, there is a risk that the result of the sequencing process may be not always "pedagogically-right" adapted to the Learners' various learning styles. The same authors presented more recently [7] an approach based on competencies on top of a model supporting learning objectives built using domain ontologies. Again, the same shortcomings can be identified related to the lack of reference to learning styles and other pedagogical parameters explicitly used in our case. [2] uses IMS-LD based templates and domain ontologies to contextualize and reuse Learning Objects in different learning experiences but does not explicitly focus on personalization and does not support alternative packaging of courseware as LOGOS does.

6 Conclusions and future work

We have presented the overall architecture of a cross-media learning platform that is being developed in the context of the LOGOS project providing an infrastructure for the exploitation of media residing at content archives. The platform includes multiple layers of repositories managing digital objects, learning objects and courseware objects as well as an Authoring Studio of tools supporting the creation of multi-platform learning content that can support learning experiences through internet, mobile devices and digital TV.

We have also presented in more detail the components of the LOGOS platform that are related with the dynamic creation of personalized learning experiences. These components essentially implement a personalization framework for supporting pedagogically-sound adaptive learning experiences using material in learning object repositories, taking into account the variety of learning needs of the Learners. Since pedagogy plays an important role to achieve this, a model for building abstract training scenarios (Learning Designs) has been also provided and an appropriate tool implemented, which guide the construction of pedagogically sound adaptive learning experiences and allow for the binding of appropriate learning resources at run-time according to the Learner Profiles.

Future work includes testing of the LOGOS platform with real users (both educational content authors and learners) as well as the evaluation of the personalization framework in the context of the experimentation phase of the LOGOS project.

Acknowledgments. The work presented in this paper is funded in the scope of the LOGOS STREP Project (IST-4-027451), named "Knowledge-on-Demand for Ubiquitous Learning".

References

1. Arapi, P., Moumoutzis, N., Christodoulakis, S.: ASIDE: An Architecture for Supporting Interoperability between Digital Libraries and ELearning Applications. Proc. ICALT 2006. Kerkrade, The Netherlands (2006)
2. Azevedo, I., Martins, C., Carrapatoso, E., Carvalho, C., V.: A model for the effective reuse of learning objects in repositories. Proc. ICALT 2006. Kerkrade, The Netherlands (2006) 56
3. Bloom, B., S., Krathwohl, D., R.: Taxonomy of Educational Objectives: The Classification of Educational Goals: Handbook I. Cognitive Domain, Longman, New York (1965)
4. Brusilovsky, P.: Adaptive and intelligent technologies for web-based education. Künstliche Intelligenz, Vol. 4 (1999)
5. Capuano, N., Gaeta, M., Lannone, R., Orciuoli, F.: Learning Design and run-time resource binding in a distributed e-learning environment. Proc. 1st International Kaleidoscope Learning Grid SIG Workshop on Distributed e-Learning Environments. Naples, Italy (2005)
6. Karampiperis, P., Sampson, D.: Adaptive Instructional Planning Using Ontologies. Proc. ICALT 2004. Joensuu, Finland (2004)
7. Karampiperis, P., Sampson, D.: Adaptive Learning Objects Sequencing for Competence-Based Learning. Proc. ICALT 2006. Kerkrade, The Netherlands (2006)
8. Lehmann, L., Aqqal, A., Rensing, C., Chikova, P., Leyking, K., Steinmetz, R.: A Content Modeling Approach as Basis for the Support of the Overall Content Creation Process. Proc. ICALT 2006. Kerkrade, The Netherlands (2006)
9. Lopez-Nores, M., Pazos-Arias, J., J., Garcia-Duque, J., Blanco-Fernandez, Y., Gil-Solla, A.: A Core of Stnadards to Support T-learning. Proc. ICALT 2006. Kerkrade, The Netherlands (2006)
10. Meisel, H., Compatangelo, E., Hörfurter, A.: An ontology-based approach to intelligent Instructional Design support. Proc. KES 2003, Oxford, UK (2003)
11. Metadata Encoding and Transmission Standard (METS) Official Website, http://www.loc.gov/standards/mets/
12. Moreno-Ger, P., Martinez-Ortiz, I., Sierra, J., L., Fernandez-Manjon, B.: A Descriptive Markup Approach to Facilitate the Production of e-Learning Contents. Proc. ICALT 2006. Kerkrade, The Netherlands (2006)
13. MPEG Group, ISO/IEC JTC 1/SC 29/WG 11/N3966, Text of 15938-5 FCD Information Technology – Multimedia Content Description Interface – Part 5 Multimedia Descrip-tion Schemes, Singapore, 2001.