An Architecture of eLearning Enterprise Authoring Studio

Ivo Hristov

Institute of Information Technologies
Bulgarian Academy of Sciences, Sofia, Bulgaria
collin@iinf.bas.bg

Abstract: The paper presents an architecture for enterprise development of eLearning content. The authoring studio is intended to work alone or as an integrated part of learning content management system in conjunction with the presented adaptation engine. The content organisation that facilitate development and repurpose of educational content is offered, by definition of granularity, structure, supporting metadata and links between internally stored information objects. Base functionality of building blocks and essential workflows of data are presented. The paper describes structure of objects and the repositories that enable usage of semantic operation at all levels for large and complex educational content development in different domains.

1 Introduction

Creation of content suitable for eLearning has never been an easy task. This is not only due to the natural differences typical for every specific domain, but also because there are wide number of involved professionals – domain experts; animators; graphic designers, developers; educationalists; learning designers and many other. The processes of analysing learning needs, design, develop and test the learning solution becomes more complicated because of structural requirements for learning objects (LO) like [RDS09]:

- Self-contained: references to other resources will decrease reusability; the more pre-requisites it needs, the more difficult will be adapting it to other contexts.
- Modular: a LO must be combinable with other objects to form composite structures as lessons and courses.
- Properly grained: proper size and a proper learning objective for a LO will facilitate reusing it.
- Traceable: a LO should be easily identifiable and traceable through the correct metadata.
- Modifiable: a LO should be modifiable to reformulate it under a given context different to the originally designed.
- Usable: a reusable LO must be easy to use and interactive interface elements it contains should be intuitive.
- Standardised: a reusable learning object must be compliant to a shared specification or standard.

These requirements, and relatively short time for design, development and test of learning content lead to following Authoring Studio (AS) Architecture guidelines:

1) Extensible and Scalable support of wide range of educational content design, development and test for small, medium and large educational projects. The educational content development organisations need to be flexible enough, to take different in size, domain and complexity projects. The Authoring Studio has to be flexible enough to enable every type of educational content development;

2) Background maintenance of automatic compliance with educational standards. Automatic support of educational standards helps to the involved professionals to concentrate themselves on the current educational goals. The automatic standard complaisance offloads the unspecific technical work from authoring staff in order to continuously support educational standards, which tend to be more and more complex and time consuming;

3) Enable customise, reuse, redesign and repurpose of informational objects at every stage of the work:

The goal of the paper is to present an architecture based on the guidelines 1), 2) and 3) that favour the authors to produce new or reuse existing content by effectively searching and annotating internal information objects. The architecture is designed to support semantic annotation and searches, but does not explicitly require usage of semantic services in order to work. The paper is organised as follows: In the “Scope of the system” topic a general overview of the system is presented and a new approach of publishing of learning content is offered. The second topic “Information objects – granularity, creation and storage” discuses the process of creation of information objects, targeting guidelines presented in the introduction. In the last topic “Conclusion and future work” some consideration about future development are presented.

2 Scope of the system

The eLearning Enterprise Authoring Studio components are intended to be used by teams of authors for production of eLearning content in the medium or large enterprise. The general overview of the building blocks and the relations between the main subsystems are shown on Figure 1. The desired architecture is targeting a set of loosely coupled configurable components. The components of the set can be used alone or as an integrated part of the eLearning development environment.

Learning Object Editor (LO Editor) – the component manages chain of information objects starting from digital image or video clip and finishes with composite LO. LOs are presented as chain of internally linked reusable information objects. The name “LO Editor” is an aggregated name for the software editors that manage the dedicated information objects.

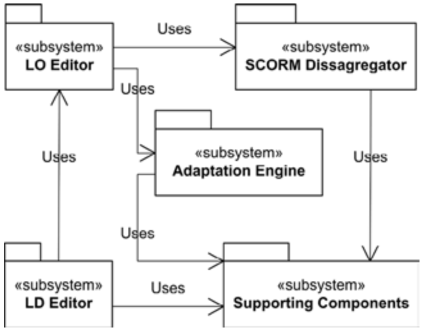

Figure 1: Authoring Studio Components

The Learning design editor (LD Editor), SCORM [AD09] Disaggregation Components (SCORM Disaggregation) and Supporting Component are presented here to illustrate the entire structure of the studio as fully functional educational development environment. These components are partially described only in the context of the management of LOs.

LD Editor is a subsystem used for creation and edition of reusable learning designs. The learning designs are intended to use the created with LO Editor LOs.

SCORM Disaggregation component is used for reverse engineering of already existing SCORM packages. The component inspects structure of externally created SCORM packets, and loads them into Authoring Studio environment for modification.

Supporting Components are used for authenticate, authorise users of the system and to manage users' connection and for other administrative purposes.

Adaptation Engine (AE) – the component is used by the LO Editors for exchanging objects and messages with external Learning Content Management System (LCMS) or other external system. The AE supports as minimum unidirectional configurable publication of eLearning content. Main building blocks of the AE and the workflow of information are shown on Figure 2. AE has to support at least content transformation according to factors like [DH06]:

- capabilities of transport protocol being used to transfer content from specific learning management system (LMS) to the learner;
- specific LMS architecture implementation;
- learner devices being used by the end user and development environment being supported:

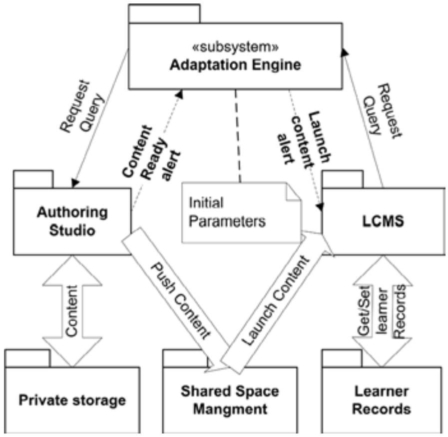

Figure 2: Adaptation Engine Workflows

More interesting is the case, where AE works as mediator between LCMS and Authoring Studio (AS). In this scenario LCMS sends request structured in XML format to the AE, describing the required object. According to the configurable "initial parameters", the AE translates the request in appropriate for the AS format. If an appropriate object is found in the "Private storage", it is pushed to the "Shared Space" equipped together with all related resources. The "content ready" alert is fired and sent back by the AE to LCMS. LCMS can now use the exported object. If the object with the requested characteristics is not found, the request description is stored in the private space of the AS for future development. An alert is sent to the responsible author roles. In a most advanced scenario, the AE can interact dynamically with LCMS and deliver dynamically adopted content according to the learner profile. The organisation of the information object discussed later is designed to support this scenario, but integration of AE with a LCMS will require additional changes in LCMS that are out of scope of the paper.

3 Information objects – granularity, creation and storage

ELearning Enterprise authoring studio has to supports future repurpose and reusability of the created content. Internally the content is organised as relatively small information objects that can be linked for creation of more complex objects. The creation of more complicated objects can be represented as linking independent "islands" of information objects that can be repurposed, changed, reordered, removed or rearranged targeting specific educational goal. The chain of logically linked objects is presented and managed as an educational development project. In general the creation of objects is organised by two cyclic steps: 1.) Store information objects into repository after every significant change over it and 2.) Add appropriate for the level of the object metadata (semantic or not) to describe what is stored. The inserted metadata may vary depending of the level of the object, project type, and many others. If the second step is omitted, the probability of future reuse of the object is limited.

3.1 Objects Granularity

The process of incremental creation of information objects is illustrated on Figure 3.

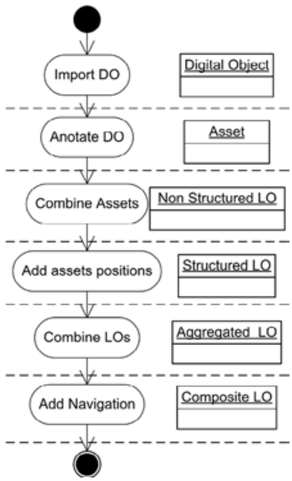

Figure 3: Information Objects Creation workflow

Creation of information object starts with digital object (DO). DOs are images, video clips, sounds, html pages or other digital object, created outside the studio with specific multimedia tools. The DOs are not intended to have any pedagogical meanings. The other objects presented on the Figure 3 have following meaning:

- Asset – annotation of DO stored in XML repository
- Unstructured LO – combination of two or more DOs, enriched with educational metadata and stored in XML repository
- Structured LO – annotated Unstructured LO plus orders, positions, and visibility between the constituent assets
- Aggregated LO – annotated combination of two or more Structured LOs
- Composite LO – annotated combination of two or more Aggregated LOs or other Composite LOs plus a definition of navigation between the constituent LOs.

The organisation of information objects into such a hierarchy enables structuring of learning content with appropriate level of granularity. To find the most appropriate object for given task, the meaningful metadata that describe the object itself have to exist. Fine granularity means that every object is annotated with proper metadata, but the process of inserting the metadata requires additional work. On the other hand, if a learning object is created, but not properly annotated its usage is limited. Balance between the cost of inserting, modifying and supporting the metadata for the given information object and the benefits of using it is essential factor for choosing the appropriate annotation level. The workflows of the information objects need to be implemented by using of integrated set of components. These components have also to ensure that all authors work only on her informational level of professional competency.

3.2 Proposed Architecture

The conceptual approach of information objects creation is shown on Figure 4. The principles of the proposed architecture are extensions of some principles of the architecture used by the LOGOS Project [AMM07]. The architecture includes tree categories of entities: Actors – member of authors' team, responsible for creation of specific type of objects; Software Object Editors – software components used to manage specific type of objects and repositories, used for appropriate storage and retrieval of information objects.

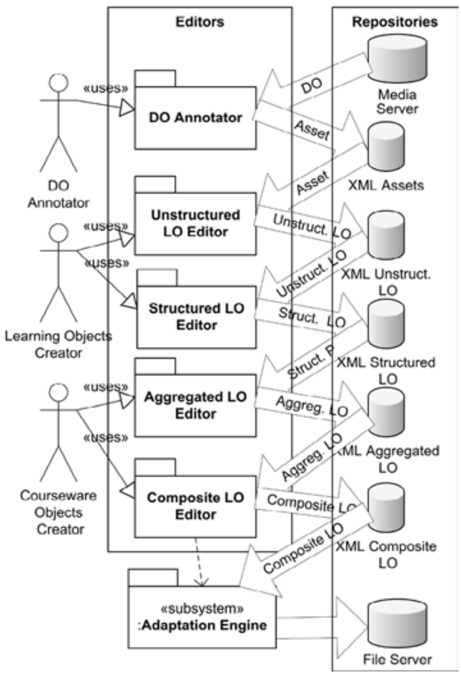

Figure 4: Creation of Information Objects

The groups of authors are defined on the basis of functional requirements for given set of logically related operations. These set of operations specify the levels in the architecture that have the similar professional qualification requirements. Typical actors are: Digital Object Annotator – responsible for annotating DOs and prepare Assets; LO Creator – responsible for creation of the structured and unstructured LOs; CO creator – responsible for the creation of aggregated or composite LOs and export of composite LOs for the external usage. The included Software Objects Editors are:

- DO Annotator – used to by DO Annotator user role to annotate DOs with appropriate metadata and create an asset. The newly created asset has a pointer to the described DO. Many assets can be based on the single DO, according to the annotation. For example many fragments from a single video clip define different assets.
- Unstructured LO Editor is used by the LO Creator in order to combine several assets according to one or more pedagogical goal.
- Structured LO Editor is used to manage presentation of included Unstructured LOs. The Editor includes graphical elements for positioning, conditional showing or hiding of elements, changing of the size and the orientation of referenced assets for underling Unstructured LO. The Structured LOs are visual representation of an Unstructured LO according to some customisation or personalisation requirements. These requirements may include tuning of the content for the screen resolution of the end user device, adapting of the content to be useful for people with disabilities, choosing of the proper size of the images according to the limitations of the transportation channel and etc.
- Aggregated LO Editor – the editor is used by Courseware Object Creator to group one or more Structured LOs, according to some set of requirement - usually educational goal or goals.
- Composite LO Editor – the editor is used by the Courseware Object Creator for inserting navigations between Structured LOs that an Aggregated LO contains. The Composite LO editor differs from the other editors, because it can be used for grouping and also for adding navigation between Composite LO structures in order to build more complicated structures of Composite LOs and/or Aggregated LOs, for example – very detailed courses, personalised course structure or even curriculums. The editor works in conjunction with AE to exports the content in different formats.

In the proposed architecture the repositories are accessible via http protocol and they present their functionality as web services. The base functionality of the interactions between object editors and repositories include following interactions types [IM03]: Search, gather, (alert)/Explore; Request/Deliver and Submit/Store. Interactions between the software editors and the repositories are implemented with the help of predefined SOAP web services. Each editor contains hard-coded (or customisable parameter) the connection points the web services that it consumes. An improvement of this strategy is usage of Universal Description, Discovery and Integration (UDDI) registry. A UDDI registry's functional purpose is the representation of data and metadata about web services. The registry, either for use on a public network or within an organisation's internal infrastructure, offers a standards-based mechanism to classify, catalogue, and manage web services, so that they can be discovered and consumed by other applications. As part of a generalised strategy of indirection among services based applications, UDDI offers several benefits including increasing code reuse and improving enterprise infrastructure management by [MH07]:

- Publishing information about web services and categorisation rules specific to an organisation.
- Finding web services (within an organisation or across organisational boundaries) that meet given criteria.
- Determining the security and transport protocols supported by a given web service and the parameters necessary to invoke the service.
- Providing a means to insulate applications (and providing fail-over and intelligent routing) from failures or changes in invoked services.

The Media Server is used for permanent storage and retrieval of DOs. The content on the Media Server can be a pre-selected and uploaded content from Content Archives. The functionalities of the Media Server presented in [BGH08] give the basis to create a large variety of learning material based on media objects – images, clips sound and many others. The Media Server manages the digital objects and provides a set of tools for creating, editing, formatting, adjusting and removing them. The Media Server supports the change of the video/audio formats in order to allow the presentation of the learning material on different user devices: desktops, mobile phones or digital TV receivers.

File server is used for common storage of packet information, for example SCORM Packages that are result of export of some set of learning object chain.

3.3 Metadata, annotations and storage of objects

All repositories in the Figure 4, except Media Server and File Server store their objects in XML format. Choosing structure of different objects, methods and standards for annotation that best serve the needs for given level are essential factors for effective objects management and usage.

The work of the DO annotator includes activities like description the single video clip or image as whole – semi automatically or by hand; selection and annotation of region or regions of the image or graphic; annotation of a part of video clip - that require definition of the start and end position; enable indexing and etc. For sustainable and yet flexible description of the DOs it is reasonable the usage of public standard for the metadata description. According to [Si01] Moving Picture Experts Group (MPEG 7) the MPEG standard allows different operation. For example on the images can be defined objects, including colour patches or textures. On a given set of video objects can be described object movements, camera motion, or relations between objects.

Taking into account tease characteristics of the MPEG 7 Standard an efficient way of annotating DOs is usage of MPEG 7 descriptors as annotation base for the DO Editor. As a result of working with the DO Annotator in the XML Assets repository is stored XML description of DOs. The Assets repository can contain many description of the some DO according to different context of the usage. Dublin Core [Hi05] is another option for annotation of DOs, but this approach has limited capacity.

Educational aspects of the LOs at all levels are made by enrichment of the objects with Learning Object Metadata (LOM) according to [IE02] standard. The usage of LOM is suitable for all editors of objects (except DO Editor), because all of them are intended to insert some educational values. The LOM enable searches and educational annotation and classifications for different by seize, purpose and granularity LOs.

In general educational aspects of the objects are defined by using LOM, Assets are annotated by the MPEG 7 or Dublin Core, the AE supports export in SCORM format. For the skeleton of the objects on different levels the Metadata Encoding and Transmission Standard (METS) [ME09] is used. METS is successfully applied in different projects [AMC06, AMM07] targeting especially educational aspects on usage of digital libraries. METS is designed specifically for digital library metadata. Each type of metadata is described in a separate section, which is linked to its counterparts by internal identifiers. These metadata can be held physically within the METS file, or in external files and referenced from within the METS document. METS is sufficient solution for solving general aspects of Object development like: Integrated development of objects at different level targeting several appropriate metadata schemas in order to represent different aspects of the objects [AMM07]; referencing objects on the lower level without the need of entirely repeat them; transformation and adaptation of objects at all enabling customisation and repurposing; usage of object as building blocks of hierarchy.

4 Conclusions and Future work

The presented architecture of eLearning authoring studio stores its objects into set of repositories that are accessed via http protocols as XML Web services. In general XML services are the sources of the information objects at all levels. The information objects can have additional semantic annotation that enables future development of the system in order to compose personalised in some aspects learning set. The future research on integrating semantic layer is intended to be developed in the framework of SINUS Project [DA09]. The major approach of the project is based on analysing and exploiting the advantages of semantic web services technology in the automation of learning object discovery, selection and composition within a distributed service architecture seamlessly integrated through ontologies.

The presented architecture is going to be tested for building courseware in the field of Bulgarian iconography. The architecture is intended to use the domain ontology for “Bulgarian Iconographical Objects” [SD09] in order to facilitate searches and automated annotation of objects at different object levels − for supporting content authors and generating customised output according to some properties of learner profile or learning environment.

Acknowledgements

This work is partially funded by Bulgarian NSF under the project D-002-189 SINUS “Semantic Technologies for Web Services and Technology Enhanced Learning”.

References

[AD09] Advanced Distributed Learning *Sharable Content Object Reference Model (SCORM)* ®: 2004 4th Edition, Overview v. 1.0, http://www.adlnet.gov/Technologies /scorm/SCORMSDocuments/2004%204th%20Edition/Documentation.aspx, 2009, (accessed April 2010)

[AMC06] Arapi P., Moumoutzis N., Christodoulakis S.: ASIDE: An Architecture for Supporting Interoperability between Digital Libraries and ELearning Applications, In *Proc. of the ICALT 2006 conference*. Kerkrade, The Netherlands, 2006 pp. 139–140

[AMM07] Arapi P., Moumoutzis N., Mylonakis M., Heodorakis G., Stylianakis G: Supporting Personalized Learning Experiences within the LOGOS Cross-Media Learning Platform. In *Proc. of the Workshop on Cross-Media and Personalized Learning Applications on top of Digital Libraries (LADL2007) in conj. with ECDL 2007 Conference*, September 2007, Budapest, Hungary. pp. 6–9

[BGH08] Bozoki S., Gnandt A., Horváth Zs., Sogrik Gy., Till V.: The Role of the Media Server in the e-Learning Platform, In *Proceedings of the 2nd LOGOS Open Workshop on "Cross-Media and Personalized Learning Applications with Intelligent Content" (LAIC 2008)*, 3 September 2008, Varna, Bulgaria, page 57

[DA09] Dochev D., Agre G.: Towards Semantic Web Enhanced Learning. In: *Proc. of the 1st International Conference on Knowledge Management and Information Sharing (KMIS 2009)*, Funchal, Madeira, Portugal, 6–8 October 2009, pp. 212–217

[DH06] Dochev D., Hristov I.: Mobile Learning Applications – Ubiquitous Characteristics and Technological Solutions, In *Cybernetics and Information Technologies*, Volume 6, No 2, 2006, pages 68–70

[Hi05] Hillmann D.: *Dublin Core Metadata Initiative – Using Dublin Core*, http://dublincore.org/documents/usageguide/, 2005, (accessed April 2010)

[IE02] IEEE *LOM Draft Standard for Learning Object Metadata*, http://ltsc.ieee.org/wg12 /files/LOM_1484_12_1_v1_Final_Draft.pdf, 2002, (accessed April 2010)

[IM03] IMS *Digital Repositories Interoperability Core Functions Information Model*, V1, http://www.imsglobal.org/digitalrepositories/, IMS DRI, 2003, (accessed April 2010)

[SD09] Staykova K., Dochev D.: Ontology "Bulgarian Iconographical Objects" – Creation and Experimental Use, *Cybernetics and Information Technologies*, Vol. 9 (2009), № 1, pp. 25–36

[MH07] Marinchev I., Hristov I.: Managing Learning Objects in large scale courseware authoring studio. In *Proc. of the Workshop on "Cross-Media and Personalized Learning Applications on top of Digital Libraries" LADL 2007*, 20 September 2007, Budapest, Hungary, pp. 6–8

[ME09] *METS: Metadata Encoding and Transmission Schema – An Overview & Tutorial*, http://www.loc.gov/standards/mets/METSOverview.v2.html, 2009, (accessed April 2010)

[RDS09] Rodríguez S., Dodero J., Sanchez-Alonso S.: A Preliminary Analysis of Software Engineering Metrics-based Criteria for the Evaluation of Learning Objects Reusability. In *International Journal on Emerging Technologies in Education (JET)*, Volume 4 (2009), Special Issue: SIIE 2008, pp. 30–34

[Si01] Sikora T.: The MPEG-7 visual standard for content description – An overview, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, Jun. 2001, pp. 696–700