B 11.
Holzinger, A., Wascher, I. & Steinmann, C. (2003) Design and Development of a LO-Editor for the Virtual Medical Campus Graz. In: Bode, A., Desel, J., Rathmayer, S. & Wessner, M. (Eds.) DeLFI 2003, Lecture Notes in Informatics, LNI Volume P-37. Bonn, Köllen Verlag, 440-449. (ISBN 3-88579-366-0).

Design and Development of a LO-Editor for the Virtual Medical Campus Graz

Andreas Holzinger[1], Ingomar Wascher[2], Christian Steinmann[2]

[1]Institute of Medical Informatics, Statistics and Documentation (IMI)
Graz University, Engelgasse 13, A-8010 Graz Austria
[2]HM&S IT Consulting
Pfeifferhofweg 12, A-8045 Graz

andreas.holzinger@uni-graz.at
iw@hms.org
cs@hms.org

Abstract: At the medical faculty of Graz University a new curriculum has been developed for the studies of human medicine. This totally new approach is based on a Module-/Track-model and follows the basic principles of interdisciplinary, topic centered and patient oriented instruction. The Virtual Medical Campus general objective is the realization of an Information System to make the curriculum digitally accessible. The learning objects (LO) used within this campus are developed on the basis of Learning Object Metadata (LOM) for "trans-national education" as a basis for international networking. For easy-to-use manipulation of the metadata, a LOM-Editor is a necessity. Although there are some LOM-Editors available at present, we decided to develop an own Editor, due to two reasons: On the one hand we defined an LO as only being complete if pre-knowledge questions and self-evaluation questions are also included; thus we no longer handle only metadata with our editor but the whole LO. On the other hand, due to the fact that our users are medical teachers and students with low computing experience, we committed ourselves to the method of User Centered Design (UCD). Therefore we developed a totally new editor and speak of an LO-Editor instead of a LOM-Editor. This paper describes the successful development of our LO-Editor from scratch under some external constraints and time restrictions. The development implies some interesting approaches. The LO-Editor is based on the new Microsoft .NET technology but it is not implemented as a web service. It is a server based application split up into two parts, the front end and the middleware. To achieve the most possible platform independence, the front end consists of simple active server pages (ASP) using dynamic HTML (DHTML) and handles the user interaction. The metadata standard is implemented into the middleware by mapping the standard into a class hierarchy. Using the built-in de-serialization and serialization functions of .NET a metadata description (XML-file) can be depicted as an object tree and the object tree may again be saved as a metadata description file. Thus a database is not needed to handle the metadata standard which induces a faster application.

Keywords: LOM-Editor, Metadata, Learning Objects, User-Centered Design

1. Introduction: The Virtual Medical Campus Graz (VMC-Graz)

The Virtual Medical Campus Graz (vmc.uni-graz.at) general objective is not the development of a new learning platform, instead the realization of a tailor-made Information System to make the curriculum digitally accessible and to support the users in their workflows [Ho02b]. The departmental "knowledge" covering the different disciplines is stored in learning objects and can be accessed via teaching and learning module catalogs. The target audience of the Virtual Medical Campus Graz is approximately 4500 students and 600 teachers of the medical faculty. Some content is being developed by domain experts in close cooperation with media specialists. This functional division secures high-quality content, therefore the best possible input and professional media didactic and technical conversion, resulting in the best possible output. However, the 600 teachers mostly use their existing content and create their own learning objects including for example pdf-files, ppt-slides etc. We consider multimedia as being one of many possible elements of an integrated solution [Ho02]. To enable the managing of such objects a compact editor is essential.

Experience from other projects has generally shown that they are mostly technology driven without enough commitment to content, content management and above all metadata-strategies. The crucial point is to provide all users (here the students and teachers) with the possibility to find relevant material quickly. Thus such a project can only be successful when it is fully committed to the execution of metadata activities. It is not just a project but a strategy which raises awareness of the possibilities of these metadata. Subsequently, Learning objects (LO) used within the VMC-Graz are developed according to accepted standards for trans-national education as a basis for international networking in the form of Reusable Learning Objects (RLO). These LOs are stored in the repository and arranged in lectures, themes and modules by the VMC-logic. Over and above the medical faculty's area of activity, the project is a pilot model for the attempts of other similarly based faculties and study directions and an important foundation for the international network in the field of e-Learning. Within the VMC-Graz the "Shareable Content Object Reference Model" (SCORM) in Version 1.1 is used consistently. SCORM is a reference model that defines a Web-based learning "content model" which consists of a set of interrelated technical specifications. In November 1997, the US Department of Defense (DoD) and the White House Office of Science and Technology Policy (OSTP) launched the Advanced Distributed Learning (ADL) initiative, www.adlnet.org [ADL03]. The metadata model of the LOM standard integrated in the SCORM supports the retrieval of learning objects in varying constellations. SCORM denominates the smallest unit which can be administered by an LMS as a Sharable Content Object (SCO). A Sharable Content Object (SCO) represents so-called assets which use the SCORM runtime environment to communicate with different Systems. Such an SCO represents the **lowest level of content granularity** which can be tracked by any system. An SCO should principally be independent of learning context to be reusable in different learning situations. Moreover, several SCOs can be assembled to form learning or exercise units on a super ordinate level. To make a potential reuse practicable, SCOs should be small units. They can be the basis for sharable content repositories which facilitate their exchange. Only an LMS may launch an SCO. An SCO itself is not allowed to launch other SCOs [Do01].

To support the teachers in providing metadata we developed a specially designed LOM-Editor. Due to the fact that we do accept a LO only if it also contains pre-knowledge and self-assessment questions we speak about a LO-Editor rather than a LOM-Editor.

2. What is a Learning Object (LO)

Learning objects (LOs) are historically grounded in the object-oriented paradigm of computer science. Object-orientation basically values the creation of components (called "objects") that can be reused [DN66], [Bo94], [Ci03], defines such a learning object as *"a granular, reusable chunk of information that is media independent."* The term **information chunk** reaches back to Miller (1956) [Mi56]; In his sense a chunk is an **information unit,** which can be perceived at one time by the individual into the short term memory (STM); Chunks are generally information units which can be individually complex and intra-individually very different [Si74].

Generally also the term "media object" is often used and for the purpose of e-Learning such an object is further defined as *"digital media designed and/or used for instructional purposes [SM02]"*. Such objects range from simple text, over video demonstrations up to interactive simulations" (see chapter 6).

According to Wiley (2001), [Wi01], however, the main idea of LOs is to break educational content down into small chunks so that they can be (re)used in various learning environments, in the spirit of object-oriented programming. The Learning Object Metadata Working Group of the IEEE Learning Technology Standards Committee (LTSC) refers to LOs as any entity, digital or non-digital, which can be used, re-used or referenced during technology enhanced learning [Ro02]. Some authors use other abbreviations, e.g. they speak of **E-learning objects (ELOs),** e.g. [MHM02], or some speak of **Reusable Learning Objects (RLOs),** e.g. [Po03].

To our own opinion, LOs are a new way of thinking about learning content and should include at least the following four characteristics (compare also with the center for International Education of the University of Milwaukee [Be03]):

- LOs must be much smaller than traditional learning units, typically ranging from 2 minutes to 15 minutes (absolute maximum 45 Minutes);
- LOs must be selfcontaining: each learning object can be used independently;
- LOs must be tagged with metadata, which contains descriptive information allowing it to be easily found;
- LOs can be aggregated: learning objects can be grouped into larger collections of content, including traditional course structures

3. Instructional design theory and learning objects

3.1 Two Examples Combination and Granularity

Instructional design theories describe methods of instruction and the situations in which those methods should be used, the methods can be broken into simpler component methods, and the methods are probabilistic [Re99]. Instructional design theory, or instructional strategies and criteria for their application play a important role in the application of Learning Objects. There are two vital factors: Combination and Granularity.

Combination. Whilst the Learning Technology Standards Committee (LTSC) promote international discussion around the technology standards necessary to support learning object-based instruction, and many people are talking about the financial opportunities about to come into existence, there is astonishingly little conversation around the instructional design implications of learning objects [Wi01].

Granularity. Discussion of the problem of combining learning objects in terms of "sequencing" leads to another connection between learning objects and instructional design theory. The most difficult problem facing the designers of learning objects is that of "granularity" [Wi01]. How big should a learning object be? The IEEE LTSC leaves room for an entire curriculum to be viewed as a single learning object, but such a large object view diminishes the possibility of learning object reusability. Due to Reusability should considered to be the core learning object notion, this question should be answered cautiously. Luckily, within the VMC-Graz this problem was relatively easy to solve due to the modular and strict logic of the curriculum. Within the VMC-Graz a LO can have any granularity with a maximum didactical duration of a lecture unit which lasts for 45 minutes. In any case the produced LO must fit into this lecture unit! This is in accordance with for example Reigeluth's Elaboration Theory [Re99], [Wi01] synthesized this and other instructional design theories into a learning object-specific instructional design theory, called Learning Object Design and Sequencing Theory.

3.2. A Taxonomy of Learning Objects

Similar to Bloom's famous taxonomy of educational objectives [Bl56], also [Wi01] developed a taxonomy of LOs and differentiates between five learning object types, which we used also within the VMC-Graz:

- **Fundamental LO,** can include for e.g. an image (JPEG, GIF or other, in medical education images play an important role!), a document (DOC, PDF, PPT, etc.), a movie (MPEG, AVI etc.); and any other file, but it can also include a text entry (even only a literature reference to a book);
- **Combined-closed**, e.g. a video with accompanying audio.
- **Combined-open,** e.g. a (external) link to a web page dynamically combining e.g. JPEG and QuickTime files together with textual material "on the fly."

- **Generative-presentation,** for example, a JAVA applet capable of graphically generating a set of staff, clef, and notes, and then positioning them appropriately to present a chord identification problem to a student.
- **Generative-instructional**, for example, an EXECUTE instructional transaction shell [Me99], which both instructs and provides practice for any type of procedure;

The purpose of the taxonomy of [Wi01] is to differentiate possible types of learning objects available for use in instructional design. This taxonomy is not exhaustive in that it includes only LO types that facilitate high degrees of reusability. Types of learning objects which hamper or even prevent reusability (e.g. an entire digital textbook created in a format that prevents any of the individual media from being reused outside of the textbook context), have been purposefully excluded.

4. Example of an existing LOM-Editor

One example of an existing LOM-Editor includes the LOM-Editor from the Technical University of Darmstadt [TU01]. It is available for download for free. In our opinion, however, this Editor has three disadvantages that finally resulted in our decision to develop an own editor: 1) The created metadata (in the accessible and tested version) is not fully compatible with SCORM 1.2.2 or any other known metadata standard (i.e. it is a kind of proprietary standard); 2) Implemented in Java, that means that the latest Java runtime environment has to be installed to function properly (that is an obstacle in a medical faculty); 3) the front end is far from being intuitive and is not really usable for our target group which mainly includes medical people with low computer literacy.

5. Design and Development of the VMC-Graz-LO-Editor

As the VMC-LO-Editor is only one and a relatively small piece of the VMC-project, its development was done in a very short amount of time related to the duration of the whole VMC development.

5.1 Technology

The underlying technology used in this project is the new Microsoft .NET framework. This choice is obvious as the whole application is web based. Microsoft .Net is a major technology change. In addition to the newly introduced C# (pronounced C Sharp) language, it fully supports many other languages as for example visual basic (VB). Visual basic in the latest version (VB.NET) has undergone a major improvement. For example VB.Net now has full OO capability and there is great support for XML technology. These are some reasons why VB.NET was chosen as the programming language.

5.2 Architecture

The LO-Editor is split into two separate parts, the front end and the middleware (see figure 1). The front end is a simple web based user interface, whereas the middleware provides all the necessary functionalities to handle different learning objects, the user interface and the metadata.

The user interface, or the so called front end, is implemented as a collection of active server pages (ASP) to reach the postulated web access capability. To demonstrate the collectivity of the three different parts of the learning object, the user interface uses tabs, one for the actual learning object (the file or text), two for the knowledge questions (computer assisted assessment – CAA) and seven to handle the metadata.

Each of the tabs together with the appending page has a unique color, three completely different colors for the learning object and the two types of questions and a set of graduated colors for the metadata tabs. Our objective was to create a simple and easy to understand interface. Several UCD (user centered design) sessions provided us with the necessary feedback from the user. In these sessions we tested not only the handling but also the color scheme of the LO-Editor. The results have been implemented in the next loops.

The implementation of the middleware took place in three main steps, namely

- Handling of the metadata
- Handling of the actual learning object
- Handling of the computer assisted assessment (CAA)

The first step is the most interesting and innovative detail of the middleware. We intended to depict the whole metadata standard in a simple class structure that would make the LO-Editor very flexible and adaptive to different metadata standards. The tricky part of this approach is the link between the class hierarchy and the metadata in XML format. The solution was very simple, as VB.NET offers excellent support for XML technology. In particular VB.NET offers the so called serialization and de-serialization functions. These functions help to create an object tree from specific XML data and vice versa. Once this main problem was solved, we could add additional functionality to the class structure, such as object sensitive tool tips, to improve the front end functionality.

Compared to the first step, the handling of the actual learning object is quite simple. The LO-Editor accepts links, text and a wide range of files. Handling of a link causes no additional work, for textual input the LO-Editor offers a simple HTML-editor to do some formatting. Whenever a file is selected as a learning object, the size of the file has to be retrieved and stored in the metadata. The last step of the LO-Editor implementation is the computer assisted assessment.

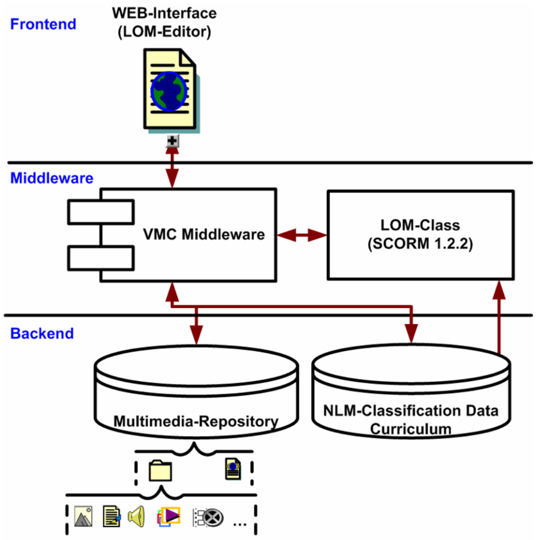

Figure 1: The Architecture of the VMC-Graz LO-Editor

At the moment the interface of the LO-Editor is available only in the German language (see figure 2 and 3), but it is possible to switch between German and English metadata descriptions. The English user interface will be implemented with the next loop together with the improvement suggestions retrieved from the UCD sessions.

Our first implemented prototype (see figure 2) was available on 1st October 2002 and was in full operation until 14th April 2003. During this time we continuously improved the LO-Editor by using a user-centered design process.

5.3 User Centered Design

User Centered Design (UCD) evolved originally in the field of Human-Computer Interaction (HCI) and was first articulated as such in User Centered System Design [ND86]. This term was originally coined by Paul Smolensky to reflect the initials of the University of California, San Diego (UCSD). UCD is not a clearly concrete methodology, rather it is a philosophy. Different versions of UCD are used at different places [VIR02], [Vr02]. However, as a design philosophy [GL85] proposed that there are some common elements for *all* versions of UCD (refer also to [GL83]) which includes early focus on **end users** and **their tasks.** At first, we studied the nature of the work, which the end users were expected to accomplish, on paper and observed which obstacles emerged. Within an iterative design and development process, we then tested the prototypes and first implementation (figure 2) and altered the user interface to fulfill our internal goals and satisfy the end users expectations (figure 3). In this course, UCD is represented as a iterative multi-stage cycle of Understanding, Design, Prototyping, and Evaluation. Experience from other projects was available (e.g. [Ho02a].

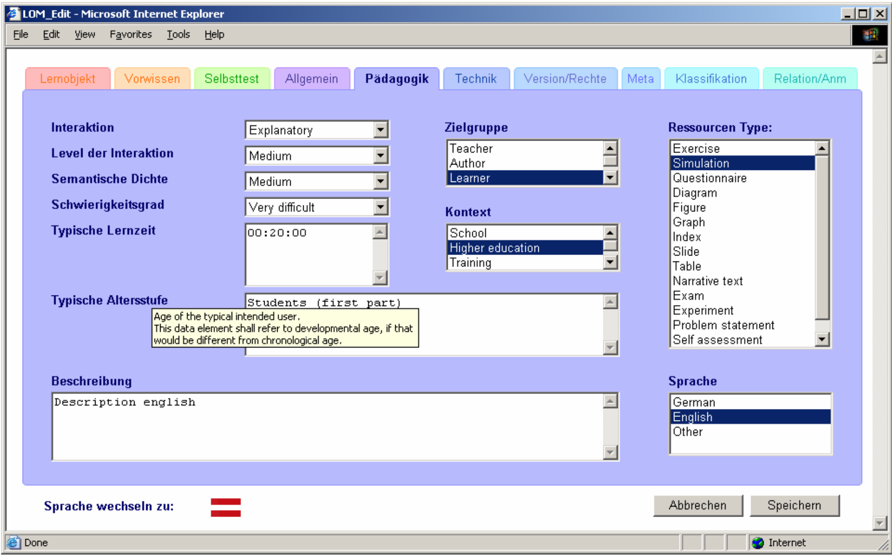

Figure 2 A screenshot from the first implementation.
Educational metadata of the LO with tool-tip text

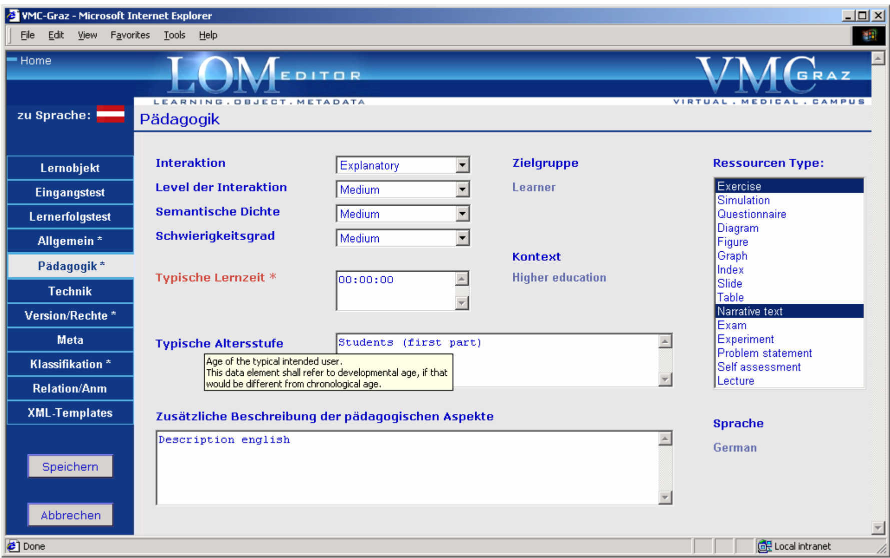

Figure 3: A look at the current version of the LO-Editor after several cycles of reengineering by UCD and with integration in the "look&feel" of the whole VMC-Graz

6. Integration into the VMC-Graz and future outlook

During the first phase of our development of Learning Objects, the initial resistance of the end users was found to complicate the application of metadata. Consequently, we were forced to distinguish between important and less important entries and pre-fill some of these with default values, thereby providing the end users with the possibility of modification. However, the members of the faculty were informed during the VMC-content courses as to the importance of the metadata and about the handling of the LO-Editor. Specifically, we became aware that it is of vital importance to raise the awareness about metadata amongst our faculty members.

For the future we plan to investigate the exchange of learning objects and their reusability within an international context.

7. Conclusion

We found that, in accordance with Torres (2002), [To02] the key principles of all versions of UCD includes the involvement of the end users from the very beginning, the understanding of the end users and their tasks and the setting of measurable goals. The iterative design and development process proved to be ideal and it was fully supported by the two/three layer structure of the LO-Editor. By using NLM we have a data base integration for our metadata. On the one hand, the use of the build-in serialization and de-serialization functions of the .NET framework, together with the hierarchical mapping of the metadata-standard, made it extremely easy to add or adapt necessary functionality for certain parts of the metadata. On the other hand the .NET framework provides various powerful and easy to adapt components to represent the metadata in our web-based application. Finally our LO-Editor has the advantage of being based on the latest international SCORM standard.

References

[ADL03] ADLNet: Advanced Distributed Learning, online at: http://www.adlnet.org, last access: 20030522

[Be03] Beck, R.J.: University of Milwaukee, Center for International Education, Learning Objects, online at: http://www.uwm.edu/Dept/CIE/AOP/LO_what.html, last access: 20030724

[Bl56] Bloom, B.S.: Taxonomy of educational objectives, handbook 1: Cognitive domain. New York: Longmans Green, 1956.

[Bo94] Booch, G.: Object-Oriented Analysis and Design with Applications. Redwood City (CA): Benjamin/Cummings, 1994.

[Ci03] Cisco: E-learning glossary, last access: 20030724, online at: http://www.cisco.com/warp/public/10/wwtraining/elearning/pdf/elearn_glossary.pdf

[DN66] Dahl, O.-J. and K. Nygaard: SIMULA: an ALGOL-based simulation language. Communications of the ACM, 1966, 9(9); S. 671-678.

[Do01] Dodds, P.: Advanced Distributed Learning, Sharable Content Object Reference Model (SCORM) Application Profile WORRKING DRAFT 1.0, March 26, 2003, Version 1.2, Release 2001, http://www.adlnet.org, last access: 20030522

[GL85] Gould, J.D. and C. Lewis: Designing for Usability: Key Principles and What Designers Think. Communications of the ACM, 1985, 28(3); S. 300-311.

[GL83] Gould, J.D. and C. Lewis: Designing for usability; key principles and what designers think. In: Conference on Human Factors and Computing Systems. 1983.

[Ho02] Holzinger, A.: Multimedia Basics, Volume 2: Learning. Cognitive Fundamentals of multimedial Information Systems. New Delhi: Laxmi, 2002. Available in German by Vogel, Würzburg, see: www.basiswissen-multimedia.at

[Ho02a] Holzinger, A.: User-Centered Interface Design for disabled and elderly people: First experiences with designing a patient communication system (PACOSY), In: Lecture Notes in Computer Science. Vol. 2398, Springer, Berlin et al.; S. 34-41.

[Ho02b] Holzinger, A.: The Virtual Medical Campus Graz: A Tailormade Information System, In: Proceedings of Vienna International Working Conference - eLearning and eCulture (books@ocg, Vol. 162), Austrian Computer Society: Vienna; S. 169-177.

[Me99] Merrill, M.D.: Instructional transaction theory (ITT): Instructional design based on knowledge objects, In: Instructional design theories and models: A new paradigm of instructional theory, C.M. Reigeluth, Editor. 1999, Hillsdale: Erlbaum; S. 397-424.

[Mi56] Miller, G.A.: The magical number seven, plus or minus two: Some limits of our capacity for processing information. Psychological Review, 1956, 63; S. 81-97.

[MHM02] Muzio, J.A., T. Heins, and R. Mundell: Experiences with reusable E-learning objects; From theory to practice. The Internet and Higher Education, 2002, 5(1); S. 21-34.

[ND86] Norman, D.A. and S. Draper: User Centered System Design. Hillsdale (NJ): Erlbaum,1986.

[Po03] Polsani, P.R.: Use and Abuse of Reusable Learning Objects. Journal of Digital information, volume 3 issue 4, 2003, 3(4).

[Re99] Reigeluth, C.M.: Instructional design theories and models: A new paradigm of instructional theory. Hillsdale (NJ): Erlbaum, 1999.

[Ro02] Robson, R.: IEEE Learning Technology Standards Committee (LTSC), online at: http://ltsc.ieee.org, last access: 20030724

[Si74] Simon, H.A.: How big is a chunk? Science, 1974, 183; S. 482-488.

[SM02] South, J.B. and D.W. Monson: A university-wide system for creating, capturing, and delivering learning objects, http://reusability.org/read/chapters/south.doc., last access: 20030724

[To02] Torres, R.J.: User Interface Design and Development. Upper Saddle River (NJ): Prentice Hall, 2002.

[TU01] TUDarmstadt: Multibook Project, http://www.multibook.de/lom/, last access: 20030724

[VIR02] Vredenburg, K., S. Isensee, and C. Righi: User Centered Design: an integrated approach. Upper Saddle River (NJ): Prentice Hall, 2002.

[Vr02] Vredenburg, K., et al.: A survey of user-centered design practice. in Conference on Human Factors and Computing Systems. 2002. Minneapolis (MI).

[Wi01] Wiley, D.A.: Connecting learning objects to instructional design theory: a definition, a metaphor, and a taxonomy, In: The instructional use of learning objects, available at http://reusability.org/read/chapters/wiley.doc, last access: 20030724