[HSDK+03-2] Stefan Hoermann, Cornelia Seeberg, Luka Divac-Krnic, Oliver Merkel, Andreas Faatz, Ralf Steinmetz:
Building Structures of Reusable Educational Content Based on LOM. In: *Proceedings of The 15th Conference On Advanced Information Systems Engineering workshop on Semantic Web for Web-based Learning*, June 2003.

Building Structures of Reusable Educational Content Based on LOM

Stefan Hoermann, Cornelia Seeberg, Luka Divac-Krnic,
Oliver Merkel, Andreas Faatz and Ralf Steinmetz

Multimedia Communications – KOM, Department of Electronical Engineering
and Information Technology, Darmstadt University of Technology,
Merckstr. 25, D-64283 Darmstadt, Germany
{hoermann, seeberg, divac-krnic, merkel, afaatz,
steinmetz}@kom.tu-darmstadt.de
http://www.kom.tu-darmstadt.de/

Abstract. This paper shows that in education and training systems LOM can be used to map the structure of courses. The necessary extension of the aggregation level is introduced. The description of the courses created with LOM allow for the reuse at all levels. Rhetoric didactic relations between the learning objects are also stored in the LOM records and support adaptivity. Our approach for the first time shows the suitability of using LOM together with a well defined knowledge base in order to create adaptive and modularized courses. In this paper an example application of the presented approach represents computer-based training.

1 Introduction

Metadata is data about data. We all know this definition and we know the power of metadata in the context of searching for e.g. relevant learning material. But supporting search machines is not the limitation of metadata. Metadata schemata providing several granularity levels allow to create well defined structures of composed learning material. This way, sections, chapters and complete courses can be reused as well as more atomic learning modules.

In the project k-MED [1] the proposed approach is realized. Learning resources of the different medicine domains become modular (knowledge modules) and become worked up with multimedia contents. As a rule, the granularity of the modules is very fine so that the modules contain only one media type. Nevertheless multimedia courses arise from the combination of these modules following the definition in [2].

In this paper related work is discussed in section 2. A technical overview is given in section 3. The following section 4 presents the solution to create, store, and manage courses which consist of modularized multimedia learning resources. Therefore at first data elements are introduced which are needed to explain the solution which has to be presented. Section 5 ends this paper with a conclusion.

2 Related Work

SCORM (Shareable Content Object Reference Model) [3] is developed by the Advanced Distributed Learning Initiative (ADL). It is a data format to structure and store modularized learning resources which are described by meta data within one packed archive file. SCORM represents a reference model for web based learning systems. At the development of SCORM the goals accessibility, interoperability, durability, and reuse of learning resources was followed up. Therefore the SCORM reference model combines already existing specifications and guidelines of more established organizations together and connects them in a larger cohesion. SCORM uses specifications of organizations like IMS Global Learning Consortium, Inc., AICC (Aviation Industry CBT Committee), and IEEE LTSC (Learning Technology Standards Committee).

DocBook [4] is a markup language which has been particularly developed for the creation of software and hardware documentations in computer science. At present the further development of DocBook is driven by the DocBook Technical Committee at OASIS [5]. With an extent of more than 400 different markup tags DocBook is suitable for the preparation of books, articles, and also on-line documents in other application areas than computer documentations. Because of DocBook's objective target it is less suitable to structure modularized multimedia learning resources to learning courses and to support reusability of these learning courses like SCORM. Nevertheless it provides interesting solutions for the markup of high quality on-line and print texts.

A number of further approaches which represent a hybrid approach of SCORM and DocBook because of their objective target has to be mentioned besides SCORM and DocBook. Their objective target is the storage of the learning resources based on XML independently of the output format of the courses. As a rule, HTML and PDF are supported as output formats. Therefore they offer possibilities to create learning courses which consist of modularized learning resources and for the markup of text modules. Regarding these approaches, it can be partly recognized that SCORM and DocBook have served as an example of the implementation.

In an incomplete list the following markup languages have to be mentioned. TeachML, a markup language to markup media independent learning resources and to create course structures in XML. TeachML is developed in the project Targeteam (TArgeted Reuse and GEneration of TEAching Material) [6]. LMML (Learning Material Markup Language), a markup language and course structure language for modularized learning and teaching resources in the Passauer Knowledge Management System (PaKMaS) [7], [8]. The LMML framework is based on a meta modeling approach [9], [10] which describes a generic meta model which is suitable for the description of learning resources. From the LMML meta model application specific instances can be derived. MedicML [11], a medium neutral markup language which is developed as a data interchange format for medical case data.

The above listed solutions offers different possibilities of creating learning courses which consist of modularized learning resources. Partially they support different learning paths in the courses and customization abilities to transform them into different output formats. Neverthless the reuse of existing learning resources and the possibility of the adaptation of the learning courses regarding the learners knowledge is limited.

Solutions like TeachML, LMML, and MedicML consist of two parts. The first part is a markup language to describe formatted text modules, the second part is a markup language to create learning courses which consist of more or less modularized learning resources. For the enhancement of the reuse of already existing modularized learning resources both components of the markup languages TeachML, LMML, and MedicML should be regarded disjointed. So text modules should be described by the corresponding markup languages and should be stored in a knowledge base. The course structures which consist of modularized multimedia learning resources could be created separately with the corresponding markup language. But for the adaptive arrangement of courses as it is described in approaches like [12], [13] another step has to be applied to the course before any presentations of the course can be created. So the course structures by means of TeachML, LMML or MedicML will serve in adaptive systems as an intermediate product at the generation of presentations of the learning courses or as an exchange format for modularized learning resources.

3. Technical Overview

In this section a technical overview of a learning system based on a knowledge base is given to introduce the environment for the approach.

Basically, the system consists of a knowledge base where multimedia resources are stored, and a teaching and learning system which is based on this knowledge base. The knowledge base itself consists of the ConceptSpace and the MediaBrickSpace. In figure 1 you can see clippings of these both areas of the knowledge base. The ConceptSpace is a formal knowledge representation in form of an ontology. It stores the keywords of the domain and semantic relations between these terms. The semantic network is kept consistent by axioms during the construction process by e.g. automatic creation of reversal relations or transitive relations when editing generic and subsumable concept relations. The construction of the ontology is a collaborative process which is carried out according to a modified Delphi method [14]. An approach of semi-automatic enrichment of ontologies can be found at [15]. In the second part of the knowledge base, which is called MediaBrickSpace, learning resources are stored. It is a set of modularized multimedia content in files like e.g. texts, images and videos. So the media bricks represent the modularized multimedia learning resources of the learning system. Every media brick in the MediaBrickSpace is described by meta data to provide mechanisms for finding and reusing of existing media bricks in the knowledge base. For this purpose, the Learning Objects Metadata (LOM) scheme from the IEEE Learning Technology Standards Committee (LTSC) [16] is used. Because of educational meta data in the scheme, LOM is very suitable to describe learning resources. The media bricks and the LOM records which belong to them are represented by rectangles in figure 1. In the MediaBrickSpace themselves relations between the media bricks are also stored. They are concerning the relation of the content of two media bricks. These relations are introduced as rhetoric-didactical relations in [17]. The rhetoric didactical relations are stored in the LOM category Relation with an extended vocabulary of relation types. ConceptSpace and MediaBrickSpace are interconnected by relations between concepts of the ontology and media bricks. A thematic clustering of the media bricks arises from the concepts and relations of the ConceptSpace and the relations between the ConceptSpace and the MediaBrickSpace.

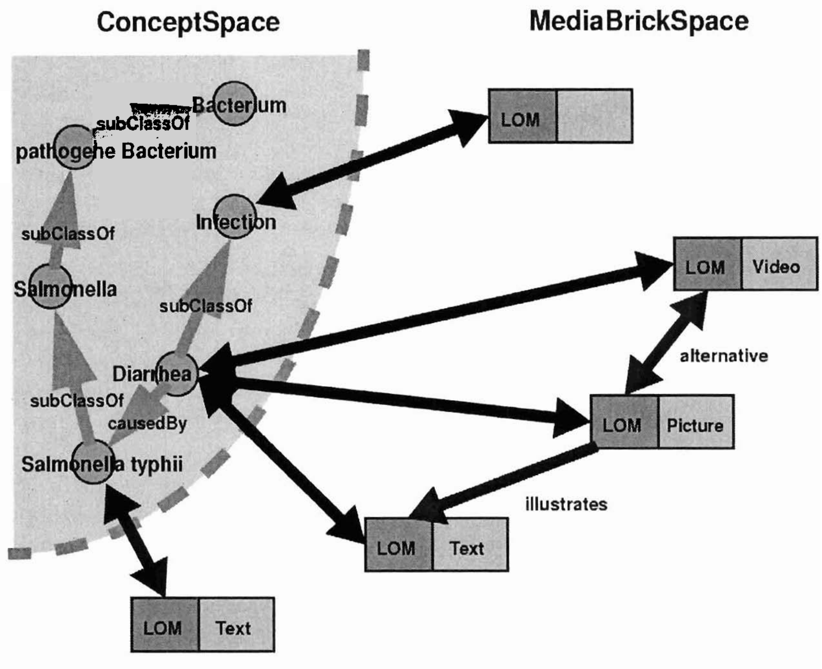

Figure 1 The knowledge base.

ConceptSpace and MediaBrickSpace grow together to a powerful knowledge base which is similar to the Semantic Web [18] concerning the used components. The knowledge base should feature teachers and learners to efficiently find teaching and learning resources. Therefore a number of new tools are needed, since changes in the authoring process arise from the modularization of learning resources.

By the modularization of the learning resources and the consequential use of LOM the possibility to create adaptive learning systems gets an integral component of the authoring process [19]. This requires new tools in the teaching and learning system to store modularized multimedia learning resources, to build courses with them.

4. Creating Course Structures with LOM

In this section the mapping of course structures with LOM will be shown. Therefore the mapping of course structures with LOM records will be presented. Finally an extension of the aggregation levels in the LOM draft is presented to refine the classification of the parts of the course structure.

4.1 Mapping Course Structures on Hierarchies of LOM Records

Three technical requirements for reusable learning material are identified:

Reusability of content on different levels in the course hierarchy (e.g. images as well as chapters and complete courses)

- The usage of standards to guarantee interoperability with other systems
- Separation of content and the way how to present this content

In this paper we concentrate on the first two point. For the third requirement see e.g. [21].

Tree-like structure of courses. The stored courses should have a treelike structure since they represent a known form of the structured representation of on-line and off-line learning resources. In addition, the treelike structuring of the modularized learning resources offers a possibility of the hierarchical structuring of learning resources without abandoning the possibility to linearize the learning resources in the course structure to present them in a linear guided tour to the learner. Links between different sections in the course structure like they are known from the hypertext, can be realized by cross references. Depending of the presentation format this cross references can be embedded as text links or footnotes into the presentations. The treelike structure of the courses also offers the possibility of the creation of a treelike navigation structure for the navigation in the course presented in HTML pages or the creation of a table of content of the course for the course in printed form.

LOM records. For the technical mapping from course structures in data structures, Learning Objects Metadata [16] of the Learning Technology Standards Committee of the IEEE is chosen. It is suitable to describe learning resources of any granularity. Particularly LOM is appropriate to map course structures which consist of fine granular content because all of this fine granular learning resources are described by LOM. So LOM based course structures can be integrated into learning systems with less additional effort. IMS Content Packaging [22] of the IMS Global Learning Consortium represents in conjunction with LOM based course structures a platform and learning system spanning solution which can be applied to exchange content and courses since it is possible to store meta data and media data in the IMS Content Packaging archive.

Objects of the class *MediaBrick* are represented by a LOM data record and physical data. Together they represent the leaves in the course tree which describes the actual content of the course in the form of texts, pictures, videos, animations and simulations. Figure 3 shows such a tree. The LOM record in figure 3 contains an exact meta data description of the physical data which can not be created automatically by processing the data themselves in the most cases. The connection between the LOM record and the physical data of the learning resource is realized by a URI which is part of the LOM record. For this purpose the LOM data field LOM.Technical.Location is used to store the URIs of the learning resources. LOM.Technical.Location has been defined in the LOM- Draft as a list in which several URIs can be stored which reference the same physical data in different places. This redundancy can be used to load data from mirrors in order to perform load sharing or to compensate a server failure. Also it is possible to create more than one LOM record that is assigned to a specific URI. Therefore the LOM records describing the same URI have to have at least different LOM identifier in the data field LOM.MetaMetadata. Both extension possibilities are not represented in figure 2.

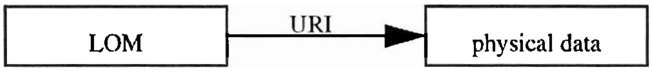

Figure 2 Relationship between LOM and learning resources.

Course structures. Objects of the class *AccumulatedMediaBricks* are also modeled by LOM records. In opposite to objects of the class *MediaBrick* they do not contain any URI, however, that points to physical data. The content of the objects of this class are represented by an accumulation of the content of the subobjects which are referenced by references. So the content of inner vertices is the sum of the content of the child vertices in the course tree. Therefore every object of the class *AccumulatedMediaBricks* contains a list of references to objects of the class *MediaBrick*. In the LOM records this can be realized by the LOM category LOM.Relation. The LOM draft specifies the category LOM.Relation as a multiple instance. Every item in this list consists of the fields LOM.Relation.Kind and LOM.Relation.Resource.Identifier. In these data fields the type of the relation and the identifier of the target LOM record are stored. With the help of this method unidirectional references can be built up between the LOM records. The relations from objects of the class *AccumulatedMediaBricks* in the course tree to their children have the type *HasPart*. The children are referenced by their LOM identifier from the parent vertices. In the reverse direction the children reference their parents by relations of the type *IsPartOf* with the parents LOM identifier. This reverse relation helps to trace back which resource is used by another resource.

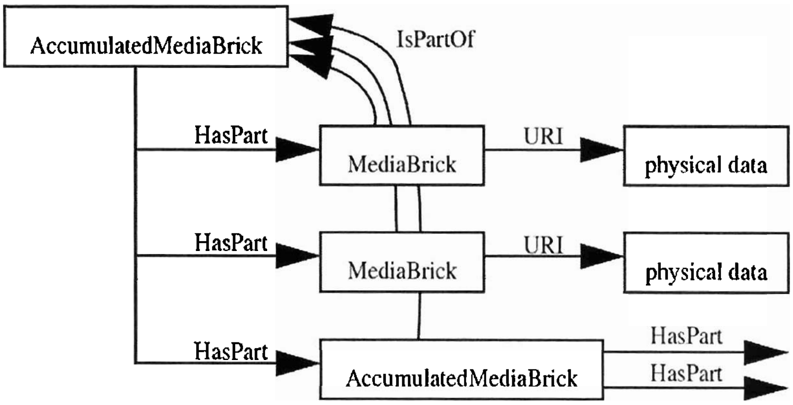

Figure 3 Creating course structures with LOM.

Figure 3 shows an object of the class *AccumulatedMediaBricks* which consists of two objects of the class *MediaBrick* and an object of the class *AccumulatedMediaB-*

ricks. Therefore it references three objects with relations of the type *HasPart* in the scheme described above. Each of the three objects references its parent object of the class *AccumulatedMediaBrick* by relations of the type *IsPartOf* as a reverse relation. Additionally each of the two objects of the class *MediaBrick* references physical data by a URI. The referenced object of the class *AccumulatedMediaBricks* consists of two other objects which are not shown in the figure. They are referenced by two relations of the type *HasPart*. Both relations of the type *IsPartOf* from these objects to their parent object are not shown in the figure.

Cross references to other resources in the course or to other resources outside the course are realized by creating references to objects of the class *MediaBrick* too. Therefore it is only necessary to insert another relation in the list of the relations which are stored in the objects of the class *AccumulatedMediaBricks*. The type of relations which reference other context related areas is *References*. The corresponding type of the reverse relation is *IsReferencedBy*.

The sequence of the relations in the list of relations of objects of the class *AccumulatedMediaBricks* is important to restore the course structure. This applies to relations of the types *HasPart* and *References*. Excepted from this are the relations of the type *IsPartOf* and *IsReferencedBy* which are not used to build up the course structure but can be used to pursue the course structure on a reversed way. By the order of the relations of the type *HasPart* and *References* the sequence of the objects of the class *MediaBrick* and the cross-references are appointed. For this reason the order must at least be guaranteed for the relations of the types *HasPart* and *References* contrary to the recommendations of the LOM draft. This can be achieved by an extension of the LOM scheme.

In practice it turns out that for the distinction of the vertices of the course tree the two classes *MediaBrick* and *AccumulatedMediaBricks* do not suffice. In the next section a set of criteria will be mentioned that are suitable to derive a set of necessary classes to create course structures with LOM records.

4.2 Extension of the Aggregation Levels

The set of the vertices of the course tree can be divided into inner vertices and leaves. Here the leaves form the set of LOM records which describe physical data of learning resources and reference them by URI. The set of inner vertices of the course tree contains LOM records which do not describe directly the physical data. Instead they provide a gradual combination of media bricks depending on their depth in the course tree. Furthermore the set of inner vertices of the course tree can be subdivided into two groups. There are vertices which visually and logically detach their content from content at the same level of the course for the course view; and there are vertices which do not. This may happen with the intention to reach an encapsulation of the content which is expressed visually and logically to the learner. Another criterion to distinguish between modularized learning resources is whether a module is a learning resource which can be presented stand alone to the learner regarding the content.

The criteria mentioned above can be used to derive a set of classes of media bricks. So in order for a LOM record to be assigned to one of this classes, it must be indicated in the LOM record. The data field LOM.General.AggregationLevel is specified for this purpose in the LOM draft. But the specified set of values for this data field is not sufficient to encode criteria mentioned above. For this reason we have defined the following set of aggregation levels for the teaching and learning system:

Atom: LOM records which are assigned to the aggregation level *Atom* are describing physical data that is referenced by a URI. The physical data that is described by a LOM record of this aggregation level forms a unit which can be presented stand alone to the learner regarding the content e.g. by a web browser in combination with plug-ins.

Subatom: Learning resources which are assigned to the aggregation level *Atom* should be presentable stand-alone regarding its content. In many cases this condition does not fit to a learning resource. For this reason the aggregation level *Subatom* is used.

CollectionOfSubatoms: The aggregation level *CollectionOfSubatoms* has been introduced to assign it to LOM records which describe a combination of LOM records of the aggregation level SubAtom to create units which can be presented stand alone to the learner regarding the content. LOM records of the aggregation level *CollectionOfSubatoms* describe the content which is aggregated by LOM records which are referenced by *HasPart* relations.

CollectionOfAtoms: The aggregation level *CollectionOfAtoms* has been introduced for the composition of several units which can be presented stand alone to the learner without detaching their content visually and logically from content at the same level of the course in the learner's view. LOM records of the aggregation level *CollectionOfAtoms* refer the combined content by relations of the type *HasPart* to LOM records which describe this content.

Chapter: The aggregation level *Chapter* has been introduced for the composition of several units with closed sense regarding the content which can be presented stand-alone to the user. The content of the composed units is detached visually and logically from the content at the same level from the learner's point of view.

Course: This aggregation level has been introduced to compose fully featured course units. In the most cases these course units consist of an introduction, the body, and finally a summary.

The impact of the consequent usage of LOM at the creation of courses results in a much better quota of reusing of already existing learning resources at all levels in the course hierarchy. This relates in particular of the erasure, the exchange, the adding of modularized learning resources in the course. But the consequent usage of LOM records at the creation of courses yields also increasing the possibility of creating adaptive courses. This is examined in the next section.

Figure 4 depicts possible *HasPart* relations between course nodes in dependence of their aggregation level. The *HasPart* Relations in this context are transitive. But not all kinds of transitive relations are desire in the course structure. Thus in figure 4 only desired transitive relations are depicted.

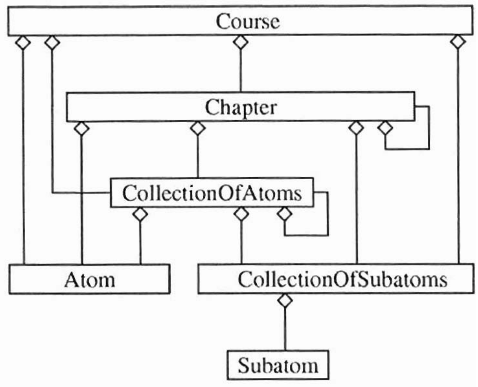

Figure 4 *HasPart* relations between course nodes of different aggregation levels.

5. Conclusion

In this paper we presented our novel approach which for the first times shows the suitability of using LOM (metadata) together with a well defined knowledge base in order to create adaptive and modularized courses. The underlying data format is based on LOM and offers a high degree of re-usability of already existing learning resources of all levels in course hierarchies by the consequent use of LOM. Since it is expected that LOM will advance to the standard for the description of learning resources by meta data in the near future, the proposed approach to create learning courses by LOM meta data records will be suitable as data exchange format or more general as interoperable solution in the future. Therefore at the creation time of the course there is only storing of the raw logical sequence and hierarchy of the modularized multimedia learning resources without any medium dependent informations. So the necessity of using rule based templates to generate presentations of the course arises. Additionally the learning courses which are created with the scheme of the presented approach are adaptable regarding the knowledge of the users.

References

1. k-MED, Knowledge-Based Multimedia Medical Education, http://www.k-med.org, July 2002
2. Ralf Steinmetz, Klara Nahrstedt, Multimedia Fundamentals Volume 1 to Volume 3, Prentice Hall, January 2002
3. Advanced Distributed Learning Initiative, Shareable Content Object Reference Model, January 2001, http://www.adlnet.org
4. Norman Walsh and Leonard Mueller, DocBook: The Definitive Guide, OReilly & Associates, Inc., 1st Edition October 1999, http://www.docbook.org/tdg

5. OASIS, DocBook Technical Committee, www.oasis- open.org/committees/docbook/

6. Gunnar Teege, Targeteam: TArgeted Reuse and GEneration of TEAching Materials, http:// www11.in.tum.de/forschung/projekte/targeteam/

7. Christian Süß,Burkhard Freitag, Passauer Knowledge Management System PaKMaS, in IFIS Report 2001/ 02, Universität Passau, 2001

8. Christian Süß, Rudolf Kammerl, Burkhard Freitag, A Teachware Management Framework for Multiple Teaching Strategies, in Proceedings of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications, Montreal, Quebec, 2000

9. Christian Süß, Adaptive Knowledge Management: A Meta-Modeling Approach and its Binding to XML, in (H.-J. Klein (Ed.), 12. GI-Workshop Grundlagen von Datenbanken, Plön, TR 2005, Christian-Albrechts- Universität Kiel, Germany, 2000

10.Christian Süß, Burkhard Freitag, Peter Brössler, Metamodeling for Web-Based Teachware Management, in P.P Chen, D.W. Embley, J. Kouloumdijan, S.W. Liddle and J.F. Roddick, Proc. Intl. WWWCM99 Workshop on World-Wide Web and Conceptual Modeling in conjunction with ER99, Nov. 15-18 1999, Paris, France, LNCS 1727, Springer Verlag

11.MedicML, XML-Struktur für medizinische Fachinformationen, http://www.medicml.de

12.Alan Bornin, Richard Kuang-Hsu Lin and Kim Marriott, Constraint-based document layout for the Web, Multimedia Systems, Volume 8, Pages 177-189, October 2000, ISSN 0942-4962

13.Veli Hakkoymaz, J. Kraft and G. Ozsoyoglu, Constraint-based automation of multimedia presentation assembly, Multimedia Systems, Volume 7, Pages 500-518, November 1999, ISSN 0942-4962

14.Clyde W. Holsapple and K.D. Joshi, A collaborative approach to ontology design, Communications of the ACM, Volume 45, 2, February 2002

15.Andreas Faatz, Stefan Hörmann, Cornelia Seeberg and Ralf Steinmetz, Conceptual Enrichment of Ontologies by means of a generic and configurable approach, in Proceedings of the ESSLLI 2001 Workshop on Semantic Knowledge Acqusition and Categorisation, August 2001

16.LOM working group, IEEE P1484.12/D6.4, IEEE Learning Technology Standards Committee, Draft Standard for Learning Objects Metadata, http:// ltsc.ieee.org/wg12/index.html

17.Achim Steinacker, Cornelia Seeberg, Stefan Fischer, Ralf Steinmetz, MultiBook: Meta-data for Webbased Learning Systems, in Proceedings of the 2nd International Conference on New Learning Technologies, 1999

18.T. Berners-Lee, J. Hendler, and O. Lassila, The semantic Web, Scientific American 284, 5 (2001), pg. 33-43

19.C.Seeberg, Life Long Learning - Modulare Wissensbasen für elektronische Lernumgebungen, Springer-Verlag, November 2002.

20.S.Weibel, J.Kunze, C.Lagoze, M.Wolf, Dublin Core Metadata for Resource Discovery, http://purl.org/dc, RFC2413, 1998

21.Cornelia Seeberg, Ivica Rimac, Stefan Hörmann, Andreas Faatz, Achim Steinacker, Abed El Saddik and Ralf Steinmetz, MediBook: Realisierung eines generischen Ansatzes für ein internetbasiertes Multimedia-Lernsystem am Beispiel Medizin, in Tagungsband: Treffen der GI-Fachgruppe 1.1.3 Maschinelles Lernen (GMD Report 114), pages 96- 105, 2000

22.IMS Global Learning Consortium, Inc., IMS Content Packaging Information Model, August 2001, http:// www.imsproject.org/content/packaging/cpv1p1p2/ imscp_infov1p1p2.html