

Decision Support Models for Composing and Navigating through e-Learning Objects

Gerhard F. Knolmayer
University of Bern, Institute of Information Systems
knolmayer@iwi.unibe.ch

Abstract

Libraries of learning objects may serve as basis for deriving course offerings that are customized to the needs of different learning communities or even individuals. Several ways of organizing this course composition process are discussed.

Course composition needs a clear understanding of the dependencies between the learning objects. Therefore we discuss the metadata for object relationships proposed in different standardization projects and especially those suggested in the Dublin Core Metadata Initiative.

Based on these metadata we construct adjacency matrices and graphs. We show how Gozinto-type computations can be used to determine direct and indirect prerequisites for certain learning objects.

The metadata may also be used to define integer programming models which can be applied to support the instructor in formulating his specifications for selecting objects or which allow a computer agent to automatically select learning objects. Such decision models could also be helpful for a learner navigating through a library of learning objects. We also sketch a graph-based procedure for manual or automatic sequencing of the learning objects.

1. Introduction

Traditional and electronic courses present a certain domain of knowledge from a specific viewpoint. Instructors offer distinct courses on the same domain of knowledge to learning communities that differ in various properties. Usually the learners are supported by some documentation of the course content. Often one textbook does not reflect all topics considered as relevant and textbooks typically cover topics that are not treated in the course. Therefore, instructors often provide individually compiled notes for their courses; they are composed with respect to the

- objectives of the course
- background of the learners
- number of hours devoted to the course, and
- preferences of the instructor.

Although courses from the same knowledge domain typically have many elements in common, it is rare to find two courses that use identical notes. When instructors gain access to learning materials, they often break them down into their constituent parts and afterwards re-assemble them in ways that support their individual instructional goals better [29]. Thus, many e-Learning concepts suggest to provide a collection of learning objects and to support the instructor in his selection and, maybe, also in his enhancement of learning objects. The relevant objects have to be brought into an appropriate sequence. The process described is called course composition, course customization or content packaging.

In Section 2 we give an overview on the concept of e-Learning objects (ELOs), discuss their properties, and summarize the standardization efforts with respect to their metadata. In Section 3 we describe roles in and responsibilities for selecting ELOs. We also discuss enhancements by the instructor's team and the navigation of learners through the set of ELOs as alternatives to pure selection procedures of the instructor. In Section 4 we show how the Gozinto method, which is widely applied in ERP systems for bill of materials explosion and materials requirements planning, can be used to determine feasible selections of ELOs. We develop integer programming models in which the instructor defines benefits and costs of selecting ELOs and the constraints existing in the educational environment; the model determines the optimal selection of ELOs under the given constraints. We also discuss variants of the model to allow individual enhancements of objects and treating prerequisite conditions as soft constraints. The sequencing technique suggested can be applied by the instructor but also if the learner is autonomously navigating through the content. Section 6 summarizes the results and presents an outlook on related issues.

2. e-Learning Objects

2.1. Definition

Since about 1999 the term "learning objects" is widely used in the e-Learning community; in 2002-09 Google provided almost 76,000 hits for this search string. There is no unique definition of learning objects. The Learning Object Metadata (LOM) Working Group of IEEE LTSC defines a learning object as any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning. Examples of technology-supported learning applications include computer-based training systems, interactive learning environments, intelligent computer-aided instruction systems, distance learning systems, web-based learning systems and collaborative learning environments [19]. The Internet Learning Solutions Group of Cisco Systems [5, p. 6] defines a learning object as a granular, reusable chunk of information that is media independent. Learning objects are also defined as elements of computer-based instruction grounded in the object-oriented paradigm of computer science, allowing instructional designers to build small (relative to the size of an entire course) instructional components that can be reused a number of times in different learning contexts. They are generally understood to be digital entities deliverable over the Internet, meaning that any number of people can access and use them simultaneously [37, p. 3].

Another source explains the notion "learning objects" by describing them as follows:

- Self-contained - each learning object can be consumed independently
- Reusable - a single learning object may potentially be used in multiple contexts for multiple purposes on multiple campuses
- Can be aggregated - learning objects can be grouped into larger collections, allowing for their inclusion within a traditional course structure
- Are tagged with metadata - every learning object has descriptive information allowing it to be easily found by a search, facilitating the use by others
- Allow for learning that is
 - just enough - if you need only part of a course, you can use the learning objects you need
 - just in time - learning objects are searchable, you can instantly find and take the content you need
 - just for you - learning objects allow for easy customization of courses for a whole organization or even for each individual [27].

Metadata is often regarded as a critical component of learning objects and therefore mentioned in their definition [7; 21].

Other terms are used very similar or even synonymously to learning objects. Among others, terms like educational objects, content objects, training components, nuggets, and chunks [5, p. 6], "media object" [31], "lecture object" [36], "knowledge object" [24], "educational objects" [12], "educational software components" [30], "courseware unit" [14], "learning unit" [4], "learning atoms" [3], "atomic information units" [23], "assignable units" [17], and "sharable content objects" [2] are used.

2.2. Properties

Properties of learning objects often invoked when working on their specifications and standards include

- accessibility
- discoverability
- durability
- interoperability
- reusability
- extensibility
- affordability, and
- manageability [31].

However, the central benefit of learning objects upon which most institutions focus is their potential reusability [3; 6; 15; 16; 28]. In Cisco's view, Reusable Learning Objects (RLOs) are based on 7 ± 2 Reusable Information Objects (RIOs), each consisting of 7 ± 2 building blocks, an overview, and a summary. Each RIO should consist of content items, practice items, and assessment items [6]. The main idea is to allow authors to publish the same input to multiple delivery media, i.e., author once, deliver many. SCORE (System for Courseware Reuse) [33], an e-Learning project at the University of Karlsruhe, is named with respect to this aspect of ELOs.

2.3. Object Size

There is a substantial discussion about the appropriate size of ELOs [25; 31; 37; 38]. For texts, an ELO could be 1 sentence, 1 paragraph, 1 idea, 1 topic, 1 section, 1 chapter, 1 lesson, or 1 course. It may be even more difficult to determine the optimal ELO size if acoustic or video streams are provided.

A small, content-specific, semantically self-contained ELO has a good chance of reusability. It has been recommended that a typical screen should be constructed from 2 or more ELOs [25]. However, a very large number of ELOs will result if a low granularity is chosen; for instance, a physical science online course consisting of 34 lessons and approximately 350 web pages contains over 1300 objects, ranging from simulations to charts and diagrams [31]. The effort for metadata documentation of

ELOs is partially size-independent and increases therefore with a rising number of small ELOs. Furthermore, small ELOs lead to many interfaces in the composed material and some of the interfaces may need smoothing to avoid content disruption.

Learning objects may be aggregated in a hierarchy. A glossary provided by the Commission on Technology & Adult Learning [7] mentions that learning objects can be grouped into larger assemblies and be nested within each other to form an infinite variety and size. In its Reusable Learning Object Strategy Cisco Systems [6, p. 8] mentions as sample hierarchy

Curriculum -> Unit -> Module -> Lesson (RLO)
-> Section (RIO).

2.4. Metadata

Generally, metadata are data about data, in our context about e-Learning objects. The types of metadata that can be associated with an Online WWW document include

- details of the document's author, publisher, and publication date
- details of the ownership of any associated intellectual property rights
- details of any ratings assigned to the data to allow protection against harmful content facilities to be applied
- searchable keywords that can be used to classify the document
- codes used to classify the document's contents with respect to a standardized classification scheme (e.g. Universal Decimal Classification)
- details of the type of data found in the document, and the relationships between different data components [9].

There are several activities to standardize Learning Object Metadata (LOM). An important progenitor to this standardization efforts is the Dublin Core Metadata Initiative which is dedicated to promoting the adoption of interoperable metadata standards and to developing specialized metadata vocabularies for describing resources. This initiative specified many metadata attributes; among others it defined 12 qualifiers for the relation between resources [10]:

- IsPartOf: is part of
- HasPart: has part
- IsVersionOf: is version of
- HasVersion: has version
- IsFormatOf: is format of
- HasFormat: has format
- References: references
- IsReferencedBy: is referenced by
- IsBasedOn: is based on
- IsBasisFor: is basis for

- Requires: requires
- IsRequiredBy: is required by

Today, the main source are the LOM defined by the Learning Technology Standards Committee (LTSC) of IEEE. The committee was formed in 1996 and is developing standards for a variety of aspects of learning technology divided into five groups; one of them is about Data and Metadata. This group is proposing standards that focus on the minimal set of attributes needed to allow the Learning Objects to be managed, located, and evaluated. The standards will distinguish obligatory and optional fields and accommodate the ability for locally extending the basic fields and entity types. Purposes of the LOM project are, among others, to

- enable learners or instructors to search, evaluate, acquire, and utilize learning objects
- enable the sharing and exchange of learning objects across any technology supported learning systems
- enable the development of learning objects in units that can be combined and decomposed in meaningful ways
- enable computer agents to automatically and dynamically compose personalized lessons for an individual learner, and to
- compliment the direct work on standards that are focused on enabling multiple Learning Objects to work together within a open distributed learning environment [18].

Where applicable, LOM may include pedagogical attributes such as

- Teaching or interaction style
- Grade level
- Mastery level, and
- Prerequisites.

The Draft Standard for LOM, Version D6.4 [19] allows, among others, to define the

- difficulty
- typical learning time
- relation between this learning object and other learning objects, if any. To define multiple relationships, there may be multiple instances of this category. If there is more than one target learning object, then each target shall have a new relationship instance. The vocabulary references the 12 qualifiers of the Dublin Core Metadata Initiative, cited above.

Several other organizations cooperate with IEEE LTSC LOM. The Instructional Management System (IMS) Global Learning Consortium project is a coalition of corporate, academic, and government partners with the vision of creating a comprehensive open architecture and infrastructure for learning technologies. The IMS Learn-

ing Resource Metadata Information Model Version 1.2.1 defines metadata with respect to relationships as "Features of the resource in relationship to other learning objects"; the vocabulary is again referencing the Dublin Core. IMS permits 100 as the smallest maximum of an unordered list [20]. Differences between the metadata specifications of IEEE LTSC LOM, IMS, and the European Ariadne project are compiled in a GESTALT document [13].

The Advanced Distributed Learning Initiative, a program of the US Department of Defense and the White House Office of Science and Technology to develop guidelines for large-scale development and implementation of distributed learning, focuses on the Shareable Courseware Object Reference Model (SCORM) which is integrating several learning standards. ADL references IEEE LTSC LOM (and, thus, indirectly the Dublin Core Metadata) with respect to the relation attribute as best practice vocabulary [2, pp. 2-75].

Of course, the optimal amount of metadata depends on the learning environment. Duncan et al. [11] argue that due to the effort associated with defining metadata, a goal should be to generate as many as possible automatically. For instance, one could classify ELOs with respect to their similarity of texts by assuming that they are related to each other if some similarity threshold value is exceeded.

Some implementations apply a restricted set of relationships. For instance, the Department of Management Information Systems of the Poznan University of Economics uses Topic Maps for describing courseware. Among the association types defined are

- Courseware_prerequisite
- Courseware_successor, and
- Courseware_related [1].

3. Course Composition

The term "software composition" is used in connection with building object-oriented software systems. Course composition may be defined as selecting ELOs from a digital library, eventually enhancing the resulting collection individually, and sequencing the objects in a way which is appropriate for the targeted learning community or individuals. In some cases the instructor will not accept an ELO and will be interested in modifying it, in replacing it with another one, or in smoothing the interfaces between some learning objects by adding sequence-dependent content.

Several groups of persons or computer programs may be responsible for course composition:

- The instructor composes the material, based on an ELO library. However, this work may not be accepted by many instructors if they regard the necessary technical knowledge beyond their

capabilities and maybe also as below their job profiles.

- Support staff of the instructor composes the material, based on an ELO library and on specifications obtained from the instructor.
- Support staff of the library composes the material, based on the ELO library and on specifications obtained from the instructor.
- The authors of the ELO library define several or even many course variants and compose appropriate material for all variants; for instance, in the "Wissenswerkstatt Rechensysteme" project [23] three levels of intensity result in three sizes for each module defined as
 - Basic Variant (equivalent to a lecture of 2 hours per week)
 - Advanced Variant (equivalent to a lecture of 4 hours per week and to exercises of 2 hours per week)
 - Expert Variant (equivalent to a lecture of 8 hours per week and to exercises of 4 hours per week).
- The authors of the ELOs or their support staff compose the material, based on the ELO library and on specification obtained from the instructor.
- The learners define the content they are interested in simply by accessing certain parts of the ELO library.
- If several sources of ELOs are available, e.g., via the Web, aggregators who are responsible for gathering metadata from remote ELO libraries and expose it to support the course composition process may be necessary.

One of the purposes of the IEEE LTSC LOM project is to "enable computer agents to automatically and dynamically compose personalized lessons for an individual learner" [18]. According to this concept, a computer agent receives a more or less precise description of the course environment. Based on this specification and previously agreed composition rules it may select ELOs and sequence them. The agent is highly dependent on the ELOs' metadata: When labeled with metadata, an e-Learning system can mix and match learning objects to create individualized learning experiences [8]. It has also been suggested to implement an adaptive competency-based approach by matching object metadata with individual competency gaps [22]. However, no well-defined rules exist how the ELOs should be selected, which of them should be enhanced, and how they should be sequenced to make instructional sense. In the following section we formulate decision support models which should contribute to solving these problems as well for manual as for automatic course composition.

4. Decision Support Models for Course Composition

4.1. A Gozinto-type Model

Some matrices and graphs are closely related to each other. An example is a quadratic adjacency matrix which defines whether a certain object i , represented as well in row i as in column i of the adjacency matrix A , is related to another object j or not. For such an adjacency matrix we would obviously end up with $A = A^T$, i.e., the matrix A equals its transpose. This redundancy is avoided by defining the matrix elements as

$$a_{ij} = \begin{cases} 1 & \text{if object } i \text{ is a direct prerequisite for object } j \\ 0 & \text{if object } i \text{ is not a direct prerequisite for object } j \end{cases}$$

As discussed above, the instructor should be able to retrieve this type of relationships from the metadata section of the ELOs. An example for an adjacency matrix is given in Table 1.

Graphically one can visualize relationships by arrows that connect the objects i and j if and only if object i is a direct prerequisite for object j (Figure 1).

The adjacency matrix of Table 1 can be transformed into the graph shown in Figure 2.

i / j	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	1	1	1						1	1				
2		0	1	1	1		1				1				1
3			0			1		1		1					1
4				0	1				1				1		1
5					0				1	1					
6						0		1			1			1	
7							0	1							1
8								0	1				1		
9									0	1				1	
10										0					1
11											0				
12		1			1							0			1
13													0	1	
14														0	
15															0

Table 1: Adjacency matrix, showing direct prerequisites for 15 Learning Objects



Figure 1: Learning Object i is a prerequisite for learning object j .

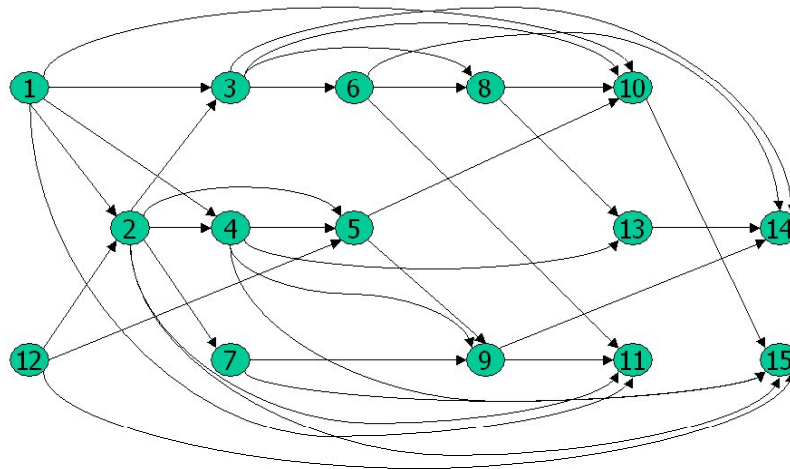


Figure 2: Graphical representations of direct prerequisites for 15 learning objects in an adjacency graph.

Typically not only direct prerequisites but also indirect ones exist; these may not always be easy recognized when selecting ELOs. Often the course composition will be under some time pressure (“just-in-time customization”) and the relationships between the objects may not be considered extremely carefully in the selection process. Therefore it would support the composition process if information as well about direct as about indirect depen-

dencies between objects were available. These relationships can be documented in a matrix T with

$$t_{ij} = \begin{cases} > 0 & \text{if object } i \text{ is a direct or indirect prerequisite for object } j \\ = 0 & \text{if object } i \text{ is neither a direct nor an indirect prerequisite for object } j \end{cases}$$

The adjacency matrix A is similar to an assembly parts matrix P in Material Requirements Planning. However, the elements of P can be any number of parts needed and their values are therefore not restricted to binary values. The arrows of the corresponding Gozinto graph show the number of units of a lower-level part that go directly into the upper-level part. From the assembly parts matrix P the total requirements matrix R can be derived [35] as $R = (I-P)^{-1}$ where I represents the identity matrix.

We can apply this procedure also to the relationships between ELOs by computing

$$T = (I-A)^{-1}$$

Matrix T shows the total number of direct and indirect prerequisite relationships between two learning objects [26]. Larger values of t_{ij} imply that there exist more paths between i and j , representing a larger number of dependencies of object j from object i . Table 2 shows the matrix T for the example given above. The numerical value of the matrix elements gives the number of paths between two nodes in Figure 2. For instance, $t_{15} = 3$ shows that there exist three paths between objects 1 and 5:

- Path 1 – 2 – 5
- Path 1 – 4 – 5
- Path 1 – 2 – 4 – 5

$(I-A)^{-1}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	2	2	3	2	1	4	6	10	10		6	16	14
2		1	1	1	2	1	1	2	4	5	6		3	9	8
3			1			1		2		3	1		2	4	3
4				1	1				2	1	2		1	3	2
5					1				1	1	1			1	1
6						1		1		1	1		1	2	1
7							1		1		1			1	1
8								1		1			1	1	1
9									1		1			1	
10										1					1
11											1				
12			1	1	1	3	1	1	2	5	6	7	1	3	10
13													1	1	
14														1	
15															1

Table 2: Matrix T showing the total number of direct and indirect prerequisite relationships between two objects

Matrix T allows the course composer to identify easily all objects that must be selected if certain target objects should be included. If, in the example above, an instructor wants to present the target objects 5 and 13 to her learners, her demand vector d becomes

$$d' = (0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0)$$

We may compute the vector $s = Td$ and note that each object j for which $s_j > 0$ holds has to be selected. In the

example we obtain $s' = Td = (9\ 5\ 2\ 2\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 6\ 1\ 0\ 0)$. Thus, if the target objects 5 and 13 should be presented, all ELOs except 7, 9, 10, 11, 14, and 15 must also be selected.

In computing T it is necessary to invert the matrix $(I-A)$. Under certain mathematical conditions it is impossible to invert a matrix. In our case this may happen if for any pair of indices i and j $a_{ij} = a_{ji} = 1$ holds. In this case a mutual dependency between two objects is claimed by the metadata. If the matrix cannot be inverted this indicates that either an error occurred in the definition of the metadata or that the granularity of the ELOs is too large because, e.g., the middle part of object j depends in the first part of object i and the last part of i depends on the middle part of object j . Thus, the existence of the inverse may be used to assess the integrity of the relationship metadata and the appropriate granularity of the ELOs.

4.2. Integer Programming Models

In general, there are trade-offs in the composition of a course and several constraints have to be considered in this process. An instructor will (at least implicitly) assign some benefit values b_j for incorporating a certain learning object j into the course. Variable costs u_j for obtaining object j from the library may exist and there may be a limited budget U for buying ELOs. We assume that the total time effort of an average learner for passing the course should not exceed E time units. From the metadata the instructor may derive an estimate for the time effort e_j a learner has to invest for studying the content of object j . Some ELOs may be different versions of the same content and the instructor should select at most one of them. As discussed above, in case of dependencies the model should ensure that ELOs i for which $t_{ij} > 0$ are selected if object j is selected.

This problem can be formulated as an integer programming model for determining the optimal values of the binary variables

$$x_j = \begin{cases} 1 & \text{if learning object } j \text{ is selected} \\ 0 & \text{if learning object } j \text{ is not selected} \end{cases}$$

The integer programming model reads as follows:

Objective Function:

$$\sum_j b_j x_j \rightarrow \max!$$

subject to

$$\sum_j u_j x_j \leq U$$

$$\sum_j e_j x_j \leq E$$

$$\sum_j x_j \leq 1 \text{ for some } j \text{ representing variants of a certain content}$$

$$-1 x_i + a_{ij} x_j \leq 0 \text{ for all } j, i \neq j, a_{ij} > 0$$

$$x_j = \{0, 1\} \text{ for all } j$$

The logical conditions are applied for all existing relationships and, thus, as well the direct as the indirect relationships are taken into account. An alternative formulation could be based on the column sum $a_j = \sum_i a_{ij}$ which can be used in reformulating the logical constraints as

$$\sum_i - a_{ij} x_i + a_j x_j \leq 0 \quad \text{for all } j$$

This formulation needs fewer constraints and leads to the same optimum but is probably computationally less efficient if, e.g., Branch&Bound methods are applied to solve the problem [39, pp. 196].

The instructor may also consider to add some additional content, e.g., to replace ELOs that are, for some reason or other, not acceptable for him. We denote by h_j the benefit of an individually generated object j and by c_j the cost of substituting the library object x_j by individual content y_j . Also the time effort needed by the learner to study the object may change from e_j to f_j . This leads to the following integer programming model:

Objective Function:

$$\begin{aligned} & \sum_j (b_j - u_j) x_j + \sum_j (h_j - c_j) y_j \rightarrow \max! \\ \text{subject to} & \quad \sum_j u_j x_j + \sum_j c_j y_j \leq U \\ & \quad \sum_j e_j x_j + \sum_j f_j y_j \leq E \\ & \quad \sum_j x_j \leq 1 \text{ for some } j \\ & \quad \text{representing variants of a certain content} \\ & \quad x_j + y_j \leq 1 \text{ for all } j \\ & \quad -1 x_i + a_{ij} x_j - 1 y_i + a_{ij} y_j \leq 0 \text{ for all } j, i \neq j \\ & \quad x_j, y_j = \{0, 1\} \text{ for all } j \end{aligned}$$

In another model one could allow to waive some objects although they are defined as prerequisites for selected objects. Such violation options could be modeled by constraints

$$\sum_i - a_{ij} x_i + a_j x_j - v_j \leq 0 \quad \text{for all } j$$

The variable v_j represents the number of direct prerequisite objects for object j that are not selected in a solution. It may be formally treated as a continuous variable but will always be integer because there is a (usually high) penalty cost associated with it in the objective function. In our example, if one would waive object 10, we would obtain $v_{15} = 1$. The missing content could, e.g., be provided in an enhancement of object 4 or by explanations in the discussion forum associated with the course.

However, if a certain object is not chosen because it is cheaper to accept the penalty, also its predecessors may not be selected without additional penalties. This could result in huge additional explanation efforts. Therefore, it may be more appropriate to use the total number of direct and indirect relationships between two objects as indicator for the disadvantages from waiving an ELO. This time we compute the column sums of matrix T , denote them by $t_j = \sum_i t_{ij}$ and use them in formulating the logical constraints as

$$\sum_i - t_{ij} x_i + t_j x_j - v_j \leq 0 \quad \text{for all } j$$

With this formulation the penalty is assumed to be higher if a certain object i interacts closely with object j . Penalties occur for all missing relationships and not only for the direct ones.

4.3. A Graph-based Sequencing Procedure

When the set of ELOs to be used has been specified it is necessary to determine the sequence in which the objects will be presented. Based on the graph shown in Figure 2 it is comparatively easy to determine feasible paths through the network. We can label all arrows that leave an object which has already been sequenced as feasible edges. Each node for which all entering arrows are labeled may be positioned in the next sequencing step. One objective of determining the schedule could be to minimize the number of jumps, defined as an event where the learning path is continued at another node. Reducing the number of jumps is attractive because it will lessen the interface problems resulting between the ELOs.

As above, we assume that we want to present the target objects 5 and 13. According to the adjacency data, also the objects 1, 2, 3, 4, 6, 8, and 12 have to be chosen, resulting in 9 objects to be presented. One jump is unavoidable at the start of the sequence because the objects 1 and 12 are not related and both are prerequisites for presenting object 2. One may start with either object 1 or object 12. After that, object 2 has to be presented. In the next step either object 3 or 4 may be chosen. Near the end of the course another jump is unavoidable because the two target objects 5 and 13 are also independent. The last column of Table 3 gives the number of jumps associated with each schedule. Of the 28 possible sequences only sequence 14 and 28 content themselves with 2 jumps.

5. Navigation through e-Learning Objects

In some learning situations the task of course composition formally disappears because the learner makes the selection and sequencing decisions by himself when browsing through the ELOs. The models developed in section 4 are also applicable if we assume that the learner should be directed towards reasonable selection and sequencing choices by recommending appropriate learning paths [32]. There could also be synchronization rules to specify or at least to restrict the order in which objects are available to the learners; access to objects could evolve dynamically according to the objects the learner already visited [34]. Links to objects preferred by the instructor could be presented more prominently than others.

Number of sequence	Objects arranged in sequence									Number of jumps
1	1	12	2	3	4	5	6	8	13	3
2	1	12	2	3	4	6	5	8	13	5
3	1	12	2	3	4	6	8	5	13	5
4	1	12	2	3	4	6	8	13	5	4
5	1	12	2	3	6	4	5	8	13	3
6	1	12	2	3	6	4	8	5	13	5
7	1	12	2	3	6	4	8	13	5	4
8	1	12	2	3	6	8	4	5	13	3
9	1	12	2	3	6	8	4	13	5	3
10	1	12	2	4	3	5	6	8	13	4
11	1	12	2	4	3	6	5	8	13	4
12	1	12	2	4	3	6	8	5	13	4
13	1	12	2	4	3	6	8	13	5	3
14	1	12	2	4	5	3	6	8	13	2
15	12	1	2	3	4	5	6	8	13	3
16	12	1	2	3	4	6	5	8	13	5
17	12	1	2	3	4	6	8	5	13	5
18	12	1	2	3	4	6	8	13	5	4
19	12	1	2	3	6	4	5	8	13	3
20	12	1	2	3	6	4	8	5	13	5
21	12	1	2	3	6	4	8	13	5	4
22	12	1	2	3	6	8	4	5	13	3
23	12	1	2	3	6	8	4	13	5	3
24	12	1	2	4	3	5	6	8	13	4
25	12	1	2	4	3	6	5	8	13	4
26	12	1	2	4	3	6	8	5	13	4
27	12	1	2	4	3	6	8	13	5	3
28	12	1	2	4	5	3	6	8	13	2

Table 3: 28 possible sequences of learning objects if objects 5 and 13 should be presented

Other options would be to display with highest priority those ELOs with high similarity to the last objects visited or to provide evaluations of former learners as basis for further selections. If such support is not intended in unaffected learning, one could at least support information that is relevant for selection and sequencing decisions.

6. Summary and Outlook

The definition and documentation of e-Learning objects are a main concept for developing courses that are customized to different audiences. Standardization bodies and consortia recommend to define relationships between learning objects in their metadata.

In this paper we discuss as central tasks of human or computerized course composition to find

- feasible collections of e-Learning objects, i.e. a set of objects which can be sequenced in such a way that the learner has been provided with all prerequisites necessary to understand the current learning content
- an optimal solution according to the teaching goals of the instructor or the learning objectives of the learners.

Feasible solutions can be obtained by applying a Gozinto-type computation, similar to procedures used for bill of materials explosion in Materials Requirements Planning. Optimal solutions can be determined by solving an integer programming model which allows to take into account many constraints that may exist in the learning environment. Feasible sequences of the selected objects can be defined by applying a graph-based procedure. The same models may also be helpful in supporting the autonomous navigation of learners through learning object libraries.

Prerequisites for applying these models are well-defined meta-data about the relationships between the objects and, for the optimization model, an explicit definition how the achievement of the learning goals depends on the inclusion of single learning objects. Unfortunately, there are not too many empirical data available about existence and quality of detailed meta-data for e-Learning objects. The scientific community should emphasize that the definition of high-quality meta-data is an important task in instructional engineering which should not be neglected, e.g., in case of budget constraints, and its members should act accordingly when developing course materials.

From a database view, the recommendations of the standardization bodies with respect to documenting the relationships between objects should be aware of potential data integrity problems. If the relationship data between two objects are stored in the meta-data sections of both objects, e.g., as attribute data *IsBasisFor* and *IsBasedOn*, insert, delete, and update anomalies may occur. They can be avoided by normalized relational data design or by implementation of trigger mechanisms.

A general remark has to be made with respect to the composition process. Solely selecting e-Learning objects will lead to a rather fragmentary presentation because the interfaces between the objects will not be smoothed and the student will be confronted with abrupt changes in the content presented. In a movie production, the Continuity Director has to avoid unintended breakpoints. In e-Learning customization a similar task becomes highly relevant because not only one version or a few versions are prepared, as in the movie production, but a comparatively large number of course variants may result. A satisfying course may need some smoothing activities in addition to the selection work.

7. References

- [1] Abramowicz, W., Kowalkiewicz, M., Zawadzki, P.: Corporate Knowledge vs. Knowledge within Corporation, in: NAISO (Ed.): CD-ROM Proceedings of the World Congress NL 2002, Networked Learning in a Global Environment, Challenges and Solutions for Virtual Education, NAISO, Slidrecht 2002
http://www.kie.ae.poznan.pl/~marek/publications/2002_NL.pdf
- [2] ADL: Sharable Content Object Reference Model (SCORM), Version 1.2, The SCORM Content Aggregation Model
http://www.adlnet.org/ADLDOCS/Document/SCORM_1.2_CAM.pdf [2001-10-01]
- [3] Ateyeh, K., Mülle, J.A.: Making Courseware Reusable, in: NAISO (Ed.): CD-ROM Proceedings of the World Congress NL 2002, Networked Learning in a Global Environment, Challenges and Solutions for Virtual Education, Slidrecht, NAISO 2002
- [4] Blackboard: Learning Units
http://company.blackboard.com/Bb5/manuals/Bb5-LevelThree-Instrucor/Blackboard_5_Level_One_Instructor_Manual_HTML/Learning_Units.htm [2001-07-03]
- [5] Cisco Internet Learning Solutions Group: E-Learning Glossary, Version 2.6
http://www.cisco.com/warp/public/10/wwtraining/elearning/pdf/elearn_glossary.pdf [2001-11-14]
- [6] Cisco Systems: Reusable Learning Object Strategy, Version 4.0 [2001-11]
http://www.cisco.com/warp/public/10/wwtraining/elearning/implement/rlo_strategy.pdf [2001-12-04]
- [7] Commission on Technology & Adult Learning: Glossary
http://www.internetttime.com/itimegroup/astd_web/glossary.htm [2001-01-03]
- [8] Cross, J.: The eLearning FAQ
<http://www.internetttime.com/itimegroup/forum/faq.htm> [2000-03]
- [9] Diffuse: Diffuse Guide to Metadata
<http://www.diffuse.org/metadata.html> [2002-02-04]
- [10] Dublin Core Metadata Initiative: Dublin Core Qualifiers
<http://dublincore.org/documents/dcmes-qualifiers/> [2000-07-11]
- [11] Duncan, C., Campbell, L., Slater, J., Graham, G.: Using Metadata in Packaged e-Learning Content: Common Practice in the UK. Paper presented to UK-MEG, 2002
<http://www.ukoln.ac.uk/metadata/education/meetings/agendas/2002-04-18/duncan.pdf> [2002-04-12]
- [12] Friesen, N.: What are Educational Objects?, in: Interactive Learning Environments 9 (2001) 3, pp. 219-230
<http://www.careo.org/documents/objects.html> [2002-05-30]
- [13] GESTALT: Getting Educational Systems Talking Across Leading-Edge Technologies
<http://www.fdgroupp.co.uk/gestalt/metadata.html> [2001-04-02]
- [14] Grütznert, I., Pfahl, D., Ruhe, G.: Systematic courseware development using an integrated engineering style method, in: NAISO (Ed.): CD-ROM Proceedings of the World Congress NL 2002, Networked Learning in a Global Environment, Challenges and Solutions for Virtual Education, Slidrecht, NAISO 2002
- [15] Hiddink, G.: Using XML to Solve Reusability Problems of Online Learning Materials, in: International ICSC Congress on Intelligent Systems and Applications (ISA'2000), Wollongong 2000
<http://wwwhome.cs.utwente.nl/~hiddinkg/professional/papers/ic2000.ps>
- [16] Hiddink, G.: Solving reusability problems of online learning materials, in: Campus-Wide-Information-Systems 18 (2001) 4, pp. 146-152
- [17] IEEE LTSC: Assignable Unit
http://ltsc.ieee.org/doc/wg3/interactive_glossary/assignable_unit.html [2000-03-03]
- [18] IEEE LTSC: IEEE P1484.12 Learning Object Metadata Working Group, Scope & Purpose
http://ltsc.ieee.org/wg12/s_p.html [2001-05-03]
- [19] IEEE LTSC: Draft Standard for Learning Object Metadata P1484.12.1/D6.4
http://ltsc.ieee.org/doc/wg12/LOM_WD4.htm [2002-03-04]

- [20] IMS: IMS Learning Resource Metadata Information Model, Version 1.2.1 Final Specification
http://www.imsproject.org/metadata/imsmdv1p2p1/imsmd_infov1p2p1.html [2001-09-28]
- [21] Kaplan-Leiserson, E.: E-Learning Glossary (2001)
<http://www.learningcircuits.org/glossary.html#L>
- [22] Longmire, W.: Content and Context: Designing and Developing Learning Objects, in: W. Longmire et al. (Eds.): Learning Without Limits, Vol. 3, Informania 2000, pp. 21-30
<http://www.learnativity.com/download/LwoL3.pdf>
- [23] Lucke, U., Tavangarian, D., Vatterrott, H.-R.: The Use of XML for the Development of an Adaptive Multimedia Teaching and Learning System, in: NAISO (Ed.): CD-ROM Proceedings of the World Congress NL 2002, Networked Learning in a Global Environment, Challenges and Solutions for Virtual Education, NAISO, Slidrecht 2002
- [24] Merrill, M.D.: Knowledge Objects, in: CBT Solutions (1998) 2, pp. 1-11
<http://www.id2.usu.edu/Papers/KnowledgeObjects.PDF> [2000-05-23]
- [25] Muzio, J., Heins, T., Mundell, R.: Experiences with reusable E-learning Objects: From Theory to Practice, in: The Internet and Higher Education 5 (2002) 1, pp. 21 - 34; Center for Economic Development and Applied Research, Royal Roads University, Victoria
http://www.cedarlearning.com/CL/eloe/eLearningObjects_sml.pdf [2001-06-06]
- [26] Myrach, T., Knolmayer, G.: Modularity and Integrity Issues in Instructional Engineering, in: NAISO (Ed.), CD-ROM Proceedings of the World Congress on Networked Learning in a Global Environment (NL2002), NAISO, Slidrecht, 2002
<http://www.ie.iwi.unibe.ch/e-learning/100029-03-TM-024.pdf>
- [27] NN: Learning Objects @ Wesleyan
<http://www.wesleyan.edu/its/acs/projects/learningobjects/> [2002-03-13]
- [28] Rada, R.: Developing Educational Hypermedia: Coordination and Reuse, Ablex, Norwood 1995
- [29] Reigeluth, C.M., Nelson, L.M.: A new paradigm of ISD?, in: Branch, C., Minor, B.B. (Eds.): Educational media and technology yearbook, Vol. 22, Libraries Unlimited, Englewood 1997, pp. 24-35
- [30] Roschelle, J., DiGiano, C., Koutlis, M., Repenning, A., Phillips, J., Jackiw, N., Suthers, D.: Developing Educational Software Components, in: Computer 32 (1999) 9, pp. 50-58.
- [31] South, J.B., Monson, D.W.: A University-wide System for Creating, Capturing, and Delivering Learning Objects, in: Wiley, D.A. (Ed.): The Instructional Use of Learning Objects, Association for Instructional Technology and Association for Educational Communications and Technology 2001
<http://reusability.org/read/chapters/south.doc> [2000-12-14]
- [32] Theilmann, W., Wessner, M., Altenhofen, M., Gerteis, W.: Instructional strategies for collaborative e-learning, in: NAISO (Ed.), CD-ROM Proceedings of the World Congress NL 2002, Networked Learning in a Global Environment, Challenges and Solutions for Virtual Education, NAISO, Slidrecht 2002
- [33] University of Karlsruhe, Institute for Program Structures and Data Organization: SCORE-Project, System for Courseware Reuse
<http://www.ipd.uka.de/SCORE/en/> [2002-01-24]
- [34] Vargas-Solar, G., Dobre, A., Dittrich, K.: Towards a Content Management Framework for Learning-Environments, in: Flückiger, F., Jutz, C., Schulz, P., Cantoni, L. (Eds.): Proceedings of the 4th International Conference on New Educational Environments, net4net, Berne 2002, pp. 3.2/19-22
- [35] Vazsonyi, A.: The use of mathematics in production and inventory control, in: Management Science 1 (1954) 1, pp. 70-85.
- [36] Vitaglione, G.: Lecture Object
<http://giosue.home.cern.ch/giosue/lectureobject/> [2001-03-27]
- [37] Wiley, D.A.: Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy, in: Wiley, D.A. (Ed.): The Instructional Use of Learning Objects, Association for Instructional Technology and Association for Educational Communications and Technology, 2001
<http://reusability.org/read/chapters/wiley.doc> [2000-12-14]
- [38] Wiley, D. A., Gibbons, A., Recker, M. M.: A reformulation of the issue of learning object granularity and its implications for the design of learning objects
<http://reusability.org/granularity.pdf> [2001-12-26]
- [39] Williams, H.P.: Model Building in Mathematical Programming, 2nd ed., Wiley, Chichester et al. 1985