# Multigranularity Reuse of Learning Resources

MAREK MEYER, CHRISTOPH RENSING, and RALF STEINMETZ, Technische Universität Darmstadt

This article investigates a scenario of reuse in which existing learning resources serve as preliminary products for the creation of new learning resources. Authors should be able to reuse learning resources and also parts of them at different levels of granularity in a modular way. The requirements of multigranularity reuse are analyzed and compared to existing solutions. A concept for modular, multigranularity reuse is presented in this article. It is also shown how this kind of reuse can be achieved in practise.

Categories and Subject Descriptors: K.3.1 [**Computers and Education**]: Computer Uses in Education—*Distance learning*

General Terms: Theory, Standardization

Additional Key Words and Phrases: Reuse, reusability, e-learning, learning resources, granularity

**ACM Reference Format:**

Meyer, M., Rensing, C., and Steinmetz, R. 2011. Multigranularity reuse of learning resources ACM Trans. Multimedia Comput. Commun. Appl. 7, 1, Article 1 (January 2011), 23 pages.
DOI = 10.1145/1870121.1870122 http://doi.acm.org/10.1145/1870121.1870122

## 1. INTRODUCTION

Since computers became omnipresent in working environments and households, people have also started to use them for education. The term *e-learning* (electronic learning) refers to scenarios whereby learning or teaching is assisted by computers. On the one hand, e-learning may support efficiency of teaching; on the other hand, learners gain the freedom to learn when, where and how they want. Learning resources comprise digital materials, for example, documents, images, videos, simulations, assessments, used in educational scenarios. When learning takes place in a formal way, a teacher usually has to provide these learning resources to the learners.

The production of learning resources either by teachers themselves or by third-party content authors has been investigated by many researchers. It is widely accepted that the production of high-quality content is labor-intensive work that incurs high costs. Therefore, it is advantageous to reuse existing learning resources in order to reduce costs. Besides being used repeatedly for other courses, learning resources may also be seen as the preliminary products of the authoring process [Hodgins 2002a]. Similar to the reuse of software libraries in software development, the reuse of learning resources is known to reduce the production costs of new learning resources. As e-learning is popular for educational scenarios in academia, vocational training, and product training, reuse of learning resources remains relevant.

Authors' addresses: M. Meyer, C. Rensing, and R. Steinmetz, Multimedia Communications Lab (KOM), Technische Universität Darmstadt, Rundeturmstraße 10, 64283 Darmstadt, Germany; email: {Marek.Meyer, Christoph.Rensing, Ralf.Steinmetz}@KOM.tu-darmstadt.de.

DOI 10.1145/1870121.1870122 http://doi.acm.org/10.1145/1870121.1870122

This article particularly considers the scenario of reuse in which existing learning resources serve as preliminary products for the creation of new learning resources for Web based training (WBT). If reuse of one learning resource on different levels of granularity may occur, this kind of reuse is called multigranularity reuse. Authors are interested in reusing the learning resources created by other authors. It is assumed that these authors belong to different organizations. Furthermore, these authors do not use a common authoring tool because they are obliged to use the tools specified by their respective organizations. There are content models which specify how learning resources may be constructed hierarchically. Authoring paradigms, such as authoring by aggregation [Hoermann 2005], allow, in principle, a new learning resource to be created as the aggregation of different smaller learning resources. However, it is necessary that these smaller learning resources are stored as individual resources. This approach works well if an organization systematically creates fine-grained, modular learning resources by using a suitable authoring environment [Barrit et al. 1999]. However, many authoring tools use arbitrary content formats that are incompatible with other authoring tools or learning management systems. Thus, learning resources are not exchanged in their source format; instead, the Shareable Content Object Reference Model (SCORM) [Advanced Distributed Learning 2004] specifies a common exchange format for the learning resources that learning resources are exported to. One disadvantage of this is that the modular components of a learning resource are no longer able to be distinguished as individual learning resources.

The contribution of this article is to discuss the issue of multigranularity reuse, to provide definitions and to present solutions that enable the reuse of modular learning resources which have due to an export process ceased to exist as individual learning resources. These learning resources are made available once again as preliminary products for the creation of new learning resources. Section 2 describes a use case for multigranularity reuse. Related Work is discussed in Section 3. Based on a review of related approaches, new definitions for reusability are developed in Section 4. Solutions for supporting multigranularity reuse are presented in Section 5, which are finally validated in Section 6.

## 2. AN EXAMPLE OF MULTIGRANULARITY REUSE

In order to clarify what we mean by multigranularity reuse, this section describes an example of such reuse.

Sarah works as a research assistant at a university. After having successfully organized a workshop about her research, she was asked to teach her colleagues how to organize a workshop. She decides to create a Web-based training course that will be made available for her colleagues. Because time is short, Sarah wants to reuse existing learning resources from other authors for her new course (see Figure 1).

In a public repository Sarah finds some learning resources about organizing events and particularly workshops. She finally decides to use a course that has been provided for free by a company. However, this course lacks some details that are important for workshop organization at Sarah's institution. For instance, the new course should contain descriptions of how to book conference rooms at the university, where to order a catering service, and how to publish workshop proceedings.

A guideline for publishing proceedings is offered by a publisher; it is suited to be reused by Sarah for her new course. The other two desired learning resources on room booking and catering are not so easy to find. Sarah searches for such learning resources but cannot find anything useful. Finally she asks a colleague, who tells her that the university has a general learning resource called "handbook of university services" that describes the various services offered by the university administration. This handbook also contains information about booking conference rooms and ordering a catering service. Even though the handbook is available as a digital learning resource, Sarah was not able to find it in

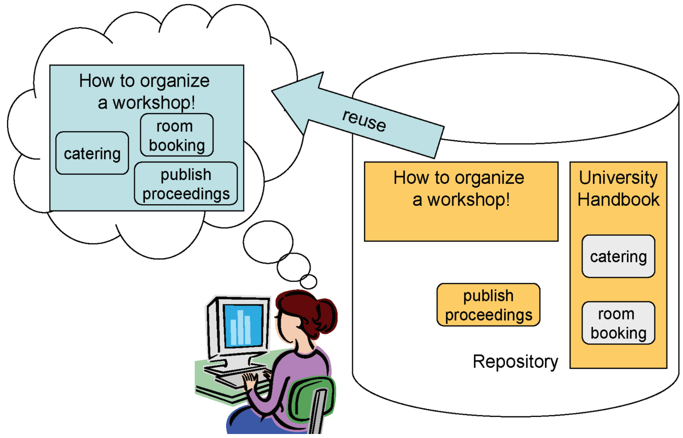


Fig. 1. Use case for multigranularity reuse.

her first search because the details of the contents were not reflected in the metadata record of that learning resource.

Now that Sarah has obtained all required learning resources, she starts to combine them into a new learning resource. First she uses the general learning resource on workshop organization as a basis. The contents are mostly suitable, but Sarah has another structure in mind. She reorganizes the learning resource by changing the order of contents; a few parts are removed because they are not relevant for her workshop scenario. Afterwards, she integrates the learning resource about publishing proceedings into the course. She also wants to aggregate the contents from the handbook of university services. In order to do so, she has to extract those fragments that she wants to reuse from the handbook. Finally, the contents of the resulting learning resource are complete. But the learning resource now contains parts in different designs and different language styles. As a last step, Sarah adapts the learning resource to unify the appearance of the course.

What makes this use case different from reuse as it has been considered before? In the given scenario parts of a larger learning resource are reused, learning resources are adapted to suit a new purpose, and learning resources that have been created with different authoring tools are combined.

## 3. RELATED WORK

This section discusses the related work about reuse in e-learning. Starting with a comparison of different definitions of learning resources, this section also gives an overview over content models and support for reuse by repositories.

### 3.1 Reusable Learning Resources

Because the production of learning resources may be quite expensive, there is a demand for reusing existing learning resources in order to decrease the costs of teaching. In the course of standardization of reuse in the e-learning field, the need for definitions of (digital) learning resources have become clear. In the last ten years, many researchers and organizations have defined reusable learning resources. The IEEE Learning Technology Standards Committee (LTSC) has defined a learning object (LO) very broadly as “any entity—digital or non-digital—that may be used for learning, education or training” [Hodgins 2002b].

Several other definitions of reusable learning objects have been discussed in the community over the last years. The definitions vary mainly in how strong the requirements from a pedagogical perspective they are. We do not want to discuss each definition here in detail and refer to Meyer [2008] instead. One definition though should be mentioned here: Polsani names the three attributes *accessibility, reusability, and interoperability* that are also found similarly in other definitions.

### 3.2 Content Models and Instructional Design

Many issues regarding learning objects have a technical dimension. However, the actual purpose of learning objects is education. Learning objects are used to teach and to learn. Therefore, didactics is an important aspect of a learning object. With the emergence of e-learning, new styles of teaching have evolved. Didactics is based on learning theories; a learning theory is a psychological explanation for how learning is assumed to happen within a person. Different learning theories, such as behaviorism, cognitivism, or constructivism, exist and partly contradict each other [Niegemann et al. 2004]. These learning theories also imply different instructional theories. The term *instructional design* is commonly used to describe the task of arranging contents and communication for educational use.

Didactics are not only useful for classroom scenarios. WBTs also require didactics to enable successful learning. Basically, each teacher may plan and realize his WBT freely with regard to structure, contents, media usage, and application of interaction and communications. In practice, though, content models have evolved. These content models are templates for structure and didactic arrangement of Web-based trainings. Usually, a content model specifies different levels of granularity of learning objects and how multiple objects of the same granularity are aggregated on the next higher granularity level. Content models have been developed in the first place by companies that produce large amounts of learning objects. For them, the standardization of a learning object's structure and sequencing leads to better quality and reduced costs.

The Advanced Distance Learning Initiative (ADL) has been founded by the U.S. Department of Defense in order to improve and standardize tools for Web-based distance learning. ADL has assembled the Sharable Content Object Reference Model (SCORM) out of different existing specifications and standards, such as IEEE Learning Oject Metatada (LOM) or IMS Content Packaging. SCORM is today the defacto standard for exchanging WBT contents, as almost every learning management system is SCORM compliant. In the current version SCORM 2004, the reference model is composed of three books: the Content Aggregation Model (CAM), Sequencing and Navigation (SN), and the Run-Time Environment (RTE) [Advanced Distributed Learning 2004].

The most relevant book of the SCORM specification regarding the focus of this article is the Content Aggregation Model, as it specifies "assembling, labeling and packaging of learning content." According to the CAM, SCORM-compliant contents are bundled as a package; this package contains the content files and a manifest file, describing structure and metadata of the contents. The content model allows the following structural elements: assets, Sharable Content Objects (SCO), activities, content aggregations and organizations. The big picture of content aggregation in SCORM is illustrated in Figure 2.

There are also a lot of other content models, such as the CISCO model and the Learnativity model. All of these content models have in common that they define multiple levels of granularity, which are respectively aggregated to larger units. The properties of these units at different levels of granularity vary. Some unit sizes are only raw media without didactic intention, others have dedicated didactic functions; larger units again do not have didactic functions but are self-contained and intended to achieve a given learning objective. A detailed comparison of different content models has been compiled by Verbert and Duval [2004].

The discussed content models regard an aggregation of content units only as a target of a reuse process, but not as a source of learning resources for reuse. Learning resources for reuse are obtained in

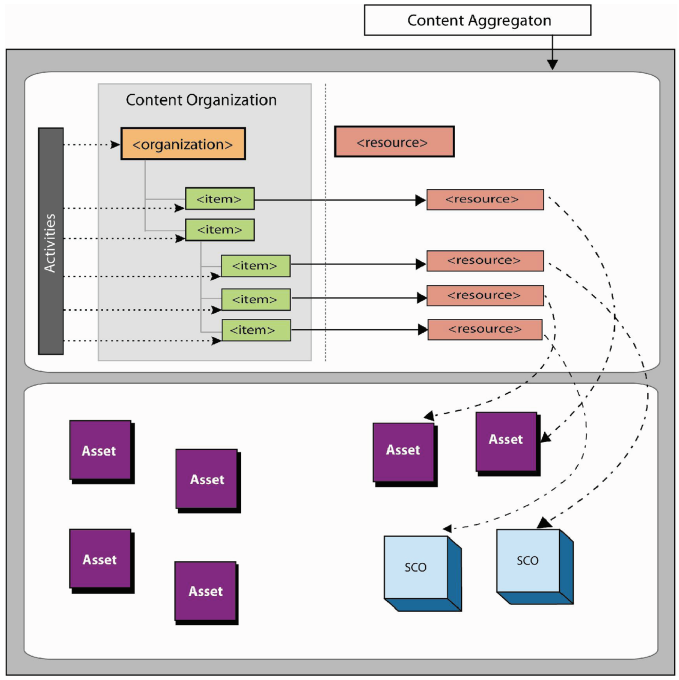


Fig. 2. Content aggregation in SCORM [Advanced Distributed Learning 2004].

the appropriate granularity from a repository. It is not intended to break an aggregation into its constituting learning resources in order to reuse them separately. Thus, existing content models support reuse of learning resources that each may have one out of multiple specified levels of granularity and to aggregate them to higher levels of granularity.

Content models specify the structure of learning resources and particularly which elements may be aggregated in which order with other elements. The structure of learning resources is well specified and also standardized for exchange between different tools (i.e., SCORM), whereas the content format of assets is not standardized. Assets within SCORM packages are often delivered as HTML documents. Some authoring tools, such as LearnCube,[1] ship arbitrary content players along with each SCORM package (in the case of LearnCube the player is developed as an Adobe Flash[2] file). The variety of content formats makes reuse difficult, especially if changes of the content are required. There are efforts for specifying XML-based content formats specifically for e-learning content. An example for such initiatives is the Learning Material Markup Language Framework [Süß and Freitag 2002]. However, there is no specific e-learning content format that is widely spread and accepted as de facto standard. Thus, even if content models are harmonized, the incompatibility of content formats is still a challenge. Recently, a new project called *Resource Aggregation Model for Learning, Education and Training* (RAMLET)[3] was initiated by the IEEE Learning Technology Standards Committee. This project is going to specify a common aggregation model, which can be mapped to different existing content models.

[1]http://www.x-pulse.de/learncube.php.
[2]http://www.adobe.com/support/documentation/de/flash/.
[3]http://www.ieeeltsc.org/working-groups/wg11CMI/ramlet.

### 3.3 Learning Object Repositories

For effective reusability it is not enough that learning resources are reusable in an educational and technological sense (that is, a learning resource can be used by a user, and the learner successfully learns from the usage). They also have to be accessible, as mentioned in the previous section. Thus, reusability is not only a property of learning resources, but also a requirement towards supporting systems. There are several types of systems that are involved over the life cycle of a learning resource: authoring environments during the creation phase, storage systems for storage and distribution, and finally learning management systems for learning.

If learning resources are to be reused, they have to be stored and made accessible. Databases for learning resources are called learning object repositories (LOR). A learning object repository is a digital archive for learning resources, which enables users to upload, search and download learning resources. Thus, a LOR is a special case of a content management system. Many different variations of LORs exist. Some repositories are completely open to everybody, such as MERLOT.[4] Others, such as the Ariadne repository [Duval et al. 2001], are available only to a closed community, where members pay for the access. The Content Sharing project[5] has worked towards a commercial learning resource marketplace, where access to the system is open for everyone, but users have to pay for downloading and using learning resources. A study in 2002 has revealed that the size of LORs varied between two-digit numbers and up to 15,000 learning objects [Neven and Duval 2002]. A more recent study lists LOR sizes of up to 22,000 learning objects and so called referatories (containing only references to learning objects) of about 120,000 learning resources [Ochoa and Duval 2008].

Storing learning resources as files in a repository is not enough. They have to be retrievable, and users should be able to get a quick overview over the contents. Metadata is used to summarize the contents and other attributes of a learning resource. In the past, there have been several approaches to specify metadata formats [Steinacker 2001]. Finally, the Learning Object Metadata (LOM) specification has prevailed. It has also been accepted as an IEEE standard [Hodgins 2002b]. However, LOM has some major drawbacks: insufficient interoperability, and a conflict of goals between automatic and human processing. LOM has been designed for enabling interoperability, but only on a syntactical level. The semantics of LOM entries, though, are not fully standardized. Two LOM-compliant applications may both read and write LOM records, but they cannot necessarily “understand” the attribute values written by the other application [Brase et al. 2003]. The second drawback is that LOM is used for both automatic processing and presentation to humans. Feedback from users in the Content Sharing project indicated that some fields are too ambiguous to be understood by users. Other fields are unspecific free text fields, making it hard to process the values by algorithms [Kabel et al. 2004].

### 3.4 Reuse and Granularity in Other Areas

E-learning is not the first application area in which reuse plays an important role. Industrial manufacturing has benefitted for a long time from standardized parts that can be assembled to different products. Examples are the assembly of cars and personal computers, where most parts are obtained from suppliers. However, this is a lopsided comparison, as the parts to assemble have to be physically built in order to be used.

More similar to reuse in e-learning is code reuse in software development. Code reuse is a valuable comparison particularly because it demonstrates the variety of what can be understood as reuse. First of all, simply copying lines of code from one computer program to another one is already code reuse. This method is likewise applied for authoring of e-learning contents. But software engineering soon

[4] http://www.merlot.org.
[5] http://www.contentsharing.com.

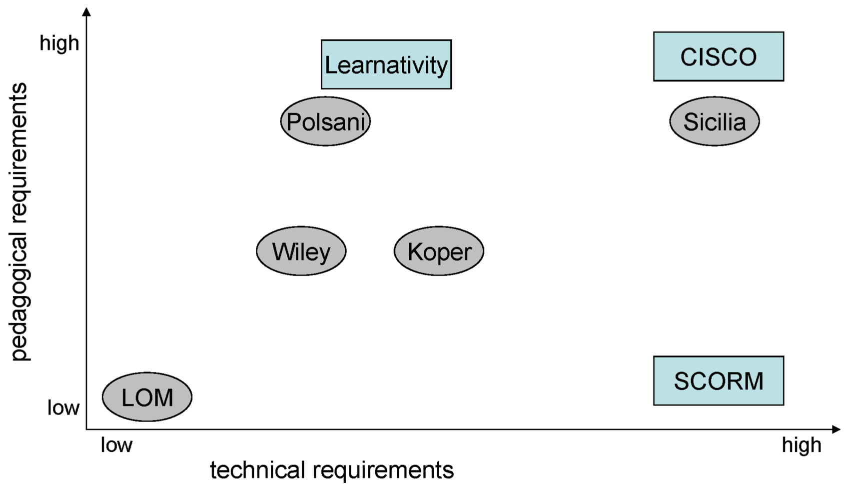


Fig. 3. Classification of learning object definitions and content models.

advanced to encapsulation of code to self-contained components, so that portions of code could be reused several times without copy&paste. There are a lot of different approaches of encapsulation, leading also to various forms of reusable code, called *methods*, *classes*, *libraries*, or *components*. Currently, the concept of reusable *Web services* is en vogue [Alonso 2004]. Nevertheless, the basic principle remains always the same: A portion of code is encapsulated and made available via an interface.

Though there are many parallels between reusable software components and reusable learning objects, there are also differences. While software components manipulate mainly data, hiding the particular implementation from the developer and user as a black box, learning objects interact with the learner. In consequence, the implementation, that is, the contents of a learning object, is always visible to the user.

## 4. REVIEW OF EXISTING APPROACHES AND DERIVED DEFINITIONS OF REUSABILITY

As Section 3 has pointed out, reuse in e-learning involves different concepts: learning objects, reusability, LORs, metadata, content models, and instructional design. Unfortunately, for most of these concepts quite different definitions and embodiments exists. This section resumes the observed definitions from the previous section, tries to extract a consensus and derives definitions valid for this article.

The definitions of learning objects and learning resources should be reviewed with respect to content models: Some definitions of a learning object relate to a particular pedagogical granularity within a given (or implicit) content model [Hodgins 2002a]; other definitions consider *learning object* as an embracing term for all entities of a content model [Hodgins 2002b].

### 4.1 Notions of Reusability

In fact, the requirements on learning objects from the various definitions can be classified in two dimensions: a pedagogical and a technical dimension. Pedagogical requirements are, for instance, Polsani's demands for independence and the dedication to the purpose of learning, and the approach of Sicilia and Sanchez-Alonso that learning objects perform a measurable transformation of a learner's state of knowledge. Technical requirements are, for example, accessibility and interoperability. Figure 3 illustrates what requirements the various definitions of learning objects and content models contain with respect to the technical and pedagogical dimensions. Interestingly, the most successful standards—LOM and SCORM—do not make pedagogical demands on learning objects. Based on this finding, a broad definition without strict pedagogical limitations will be used throughout this article. We define the term *learning resource* (LR) as “a digital resource used for e-learning” [Rensing et al. 2005].

Tightly connected to learning object definitions is the notion of *reuse* and *reusability*. Again, there are a pedagogical and a technical dimension of reusability. Furthermore, reusability can be understood either as a property of a learning object, or as functionality of systems that deal with learning objects. From a pedagogical point of view, reusability principally means that a learning object, which is pedagogically useful for its original purpose, achieves either the same, or a different predictable pedagogical result in another educational context. Typical pedagogical requirements found in reusability definitions are independence, self-containment, or a defined learning objective.

The focus of this article, though, is on *technical properties of reusability*. From the definitions of learning objects and reusability, three major requirements for successful reuse can be extracted: interoperability, availability and retrievability. As different definitions and meanings of these terms exist as well, we clarify our notion of these terms as follows.

—*Interoperability.* Interoperability means that a reusable learning resource can be exchanged between two systems (e.g., authoring tools, repositories, learning management systems), which both comply with a common specification. [Hodgins 2002a; Polsani 2003; Duval and Hodgins 2003].

—*Availability.* Availability of a reusable learning resource is given if it is published either publicly or to a restricted community, allowing users to physically access (i.e., use or download) the learning resource. [Wiley 2000; Polsani 2003; Koper 2003].

—*Retrievability.* A learning resource is retrievable, if a user, who would benefit from this learning resource, is able to find it. Retrievability demands first adequate metadata for a learning resource, and second a retrieval system that makes use of this metadata for search interfaces. [Hodgins 2002a; Duval and Hodgins 2003].

Existing approaches for reuse of learning resources interpret reuse as *simple reuse*, that is, applying a learning resource in a new educational scenario without changes to the learning resource. Changing a learning resource before reuse is not covered by existing definitions of reuse. Furthermore, there is a gap between the hierarchical approach of content models and systems supporting reusability. All content models which have been presented in the previous section intentionally aim at enabling the creation of fine-grained contents, which can be reused in different combinations. This approach works fine within a given homogeneous system landscape. It is assumed that learning resources that are already aggregated into larger structures are still separately stored in a repository in order to be reused in another aggregation. A decomposition of aggregated structures into learning resources of lower levels of granularity is not intended by these approaches.

The interoperability requirement leads to another concept: modularity. Can modularity be a criterion for reusability? And what means modularity with regard to learning resources? In order to answer these questions, a definition of modularity has to be found.

### 4.2 Definition of Modularity and Modular Operations

To approach what modularity means we take a look at the concepts of modularity in two other research areas: system design and software engineering. The first excursion takes us into the design of systems, such as computers or other complex industrial devices. According to Baldwin and Clark,

> Two subsidiary ideas are subsumed in the general concept [of modularity]. The first is the idea of interdependence within and independence across modules. (...) The second idea is

captured by three terms: abstraction, information hiding and interface. [Baldwin and Clark 1999]

Modularity is seen as a design principle, which splits a complex system into several modular components, which each have less complexity than the overall system. Modules have high internal and low external dependencies; they provide an interface for communication with other components of the system. Thus, the module concept of Baldwin and Clark is similar to the entities of content models in e-learning. The process of transforming a design by increasing its modularity is called modularization. Baldwin and Clark specify six basic modular operations, which are sufficient to describe all structural changes of an overall system. These six modular operators are splitting a design into modules, substituting one module design for another, adding a new module to the system, excluding a module from a system, inverting to create new design rules, and porting a module to another system. With this set of modular operators, all structural changes of a system design can be described. If this notion of modularity is transferred to learning resources, a congruent set of structural operations should exist for the constitution of learning resources.

From software engineering we know that the existence of interfaces, self-containment, and interchangeability are important attributes. Furthermore, modularity definitions demand that modules of a modular program can be exchanged without disassembling the program. If a program is not yet modular, it can be modularized (according to Baldwin and Clark) in order to obtain a modular structure, which enables exchange (i.e., reuse) of modules.

Although reuse in industrial and software engineering is different from reuse in e-learning, the two presented concepts of modularity may indicate the direction for modularity considerations in reuse of learning resources. As in engineering, the reasons for modularization in e-learning are primarily the reduction of costs and complexity. Costs can be reduced by avoiding production costs for new learning resources as long as the effort for reuse—retrieval costs, royalties, and repurposing costs—is lower than the production costs for a new learning resource. Content models for learning resources already apply the concept of partitioning a learning resource to the same two purposes: reduction of complexity by standardized aggregation structures and reduction of costs by reuse. To support reuse, it is necessary to expose modular learning resources for exchange, to provide a clear interface, and to support abstraction in order to reduce complexity for creation and reauthoring. Abstraction of learning resources comes in the form of metadata. Ideally, it suffices to read the metadata record of a learning resource to know whether and how the learning resource may be used, though in practice metadata is mostly not precise enough.

What is missing in current content models and reuse definitions is the dynamic aspect of modularization that the concept of Baldwin and Clark contains. Baldwin and Clark incorporate that the breakdown of contents into modules can be changed by modular operators. Applied to learning resources, this would mean that a learning resource can be repartitioned. It should be possible to transform parts of a learning resource into modules, to aggregate external modules, and to port modules within a learning resource to other learning resources. Such a modularity property as a complement to reusability would give authors, tutors and other users more freedom of choice; they could chose the boundaries of modules to reuse. Derived from the discussed modularity definitions, modularity of learning resources is defined as follows:

*Definition* 4.1 (*Modularity*). A content model or specification for learning resources is *modular* if it allows to encapsulate, expose and separately reuse parts of a learning resource; these parts are called modules. Furthermore, modularity requires that modular operations can be performed on these modules. By means of these modular operations, the content of modules and the structure by which modules are organized may change: the order of modules may change, modules may be removed, new modules may be inserted, fragments of modules may become modules themselves, and modules may be adapted to fit the surrounding modules.

Which modular operations are required is still left to be defined. Based on the modular operators of Baldwin and Clark in conjunction with the open research issues of Duval and Hodgins [2003] six modular operations for learning resources have been identified. These six operations allow a wide range of changes of learning resources. However, the list is not meant to be complete; further modular operations may be added. The six basic modular operations on learning resource are

—modularization,
—aggregation,
—exclusion (removal of modules),
—replacement (substitute one module by another),
—reorganization (changing the order of modules), and
—adaptation (transformation of a module).

Having defined modularity as a new property of learning resource formats, reusability could be redefined to put the envisioned choice of module boundaries into practice. The next section will analyze how the definition and requirements of reusability can be extended to make use of learning resource modularity.

### 4.3 Multigranularity Reuse

Modularity of a learning resource format has been defined in the previous section as the property of allowing to encapsulate, expose and separately reuse parts of a learning resource. This property is already more advanced than the notions of simple reuse that were examined before: the definition of modularity allows to partition and repartition a learning resource into reusable modules. However, the definition of modularity does not specify exactly which parts of a learning resource can be turned into modules. In this perspective, modularity is a weak definition with respect to the goal of supporting reuse of fine-granular learning resources. A stronger definition would claim that any part of a learning resource that is desired to be reused can be transformed into a module.

If the creation of a learning resource is based on an underlying content model (cf. Section 3.2), this content model specifies particular content element types of different levels of granularity that can be aggregated. It can be assumed that at least all these elements of a content model may be parts that someone might want to reuse. Ideally, reuse of learning resources of each aggregation level of an underlying content model should be enabled; in this case we speak of multigranularity reusability.

*Definition* 4.2 (*Multigranularity reusability*). *Multigranularity reusability* of learning resources means to generally enable reuse of learning resources and all their reusable fragments at multiple levels of granularity as modules.

The definition of multigranularity reusability is stronger than modularity. It ensures that all learning resources of all aggregation levels of a content model can be reused. So far, we have a definition of multigranularity reusability. There are some challenges for realizing this property in practice. The next section will describe these challenges and propose solutions.

## 5. CHALLENGES AND SOLUTIONS FOR SUPPORTING MULTIGRANULARITY REUSABILITY

In the previous section existing approaches of reusability have been analyzed. New definitions for modularity of learning resources and multigranularity reusability were introduced. These definitions, when applied in practice, may arouse some new challenges. This section analyzes which challenges are caused by the demand for modularity and multigranularity reusability. Four solutions are presented that contribute to supporting multigranularity reuse. The first contribution is a module concept that extends the SCORM standard. This module concept enables that SCORM packages can be aggregations of separate reusable modules. A second contribution is the definition of a generic reference model for modularization processes. The process model describes how monolithic learning resources can be tranformed into a modular form, that is, an aggregation of modules. The third contribution introduces an improved authoring by aggregation process. And finally, a fourth contribution deals with the need to adapt learning resources to new contexts.

### 5.1 Definition of Requirements of Multigranularity Reusability

Multigranularity reusability has been defined as enabling reuse of learning resources and all their reusable fragments at multiple levels of granularity as modules. In Section 4.1 three main technical requirements of reusability were identified for the case of simple reuse. In the scenario description, reuse was extended to cover different kinds of usage: simple reuse of an unmodified learning resource, but also re-authoring and aggregation. Thus, multigranularity reuse has to become a granularity aware realization of simple reuse, as well as re-authoring, and aggregation.

Reusability is, as discussed earlier, not only a property of a learning resource itself, but also of the system, which enables and supports retrieval and reuse. Proceeding from simple reuse to multigranularity reuse, the impact of the system(s) on reusability increases; not only does the sheer exploding number of learning resources put higher demands on retrieval systems; also system support for modularization, aggregation and adaptation of learning resources is essential for successful reuse.

Basically, the technical reusability requirements that have been discussed in the previous section for reuse in general can be applied as well for multigranularity reuse. Yet they are not sufficient to completely ensure multigranularity reuse. Some of the requirements have to be adapted and extended. For example, interoperability in the sense of multigranularity reuse means more than being transferable and executable in a different learning management system; multigranularity interoperability premises that an aggregation of multiple learning resources of similar complexity results again in a valid learning resource. As Section 4 has pointed out, there are three main properties that are often used to characterize the technical dimension of reusability. These properties are interoperability, availability, and retrievability. Now, these properties can be concretized with regard to multigranularity reuse.

*Definition* 5.1 (*Availability*). All fragments of a learning resource are individually available for reuse, and especially for aggregation in larger learning resources.

*Definition* 5.2 (*Retrievability*). All fragments of a learning resource can be appropriately found and individually obtained by users. This particularly requires that adequate metadata for each fragment is either already available or can be automatically generated.

*Definition* 5.3 (*Interoperability*). All fragments of a learning resources can be separately reused, both individually and for aggregation. Aggregations of learning resources or fragments of learning resources are required to be sound learning resources themselves.

These requirements are of technical nature; they do not directly address didactic issues. Assuming that reusability (whether simple or multigranularity) always depends strongly on pedagogical skills of authors or teachers, we leave pedagogical tasks to human experts and focus on technical issues.

### 5.2 A Module Concept for SCORM-Compliant Learning Resources

As SCORM probably is the most popular exchange format for learning resources, it has been chosen as a basis for the implementation of a module concept. First, the SCORM specification has been compared to the requirements of modularity. Modularity of a learning resource specification is defined by two properties: first it has to enable encapsulation, exposure and separate reuse of parts of a learning resource. And second, modular operations have to be possible. SCORM would be considered a modular specification if it fulfills these two properties.

First, the requirement of encapsulation, exposition, and separate reuse of fragments is to be checked. A whole SCORM package is encapsulated and exposed via an interface. SCORM as well defines a content aggregation model and provides interface specifications for fragments of a SCORM package (SCOs, assets, activities). Thus, these fragments can be encapsulated and exposed. However, the SCORM specification does not meet the requirement of enabling the separate reuse of these fragments. Therefore, SCOs, assets and activities of a SCORM package are not modules according to the above definition. Even though a SCORM package has an inner structure that reflects structural aspects of an original content model, the aggregation elements of a SCORM package are not full-fledged modules (these elements lack the property of separate reusability).

Content aggregation elements of SCORM are SCOs, assets and activities (so called *items* in a manifest). All of these element types are specified in the manifest of a SCORM package and may optionally be described by a separate metadata record. However, such an element cannot be extracted for separate reuse or exchanged without a thorough analysis of the actual content. The reason for this nuisance is that the actual content files are sometimes not assigned to the corresponding aggregation element. Even worse, some files, such as style sheets and particular images, are used by multiple elements. It is not necessarily bad to use only a single instance of a style sheet or logo for a whole learning resource; on the contrary, it often makes sense to avoid redundancy. But if parts of a SCORM package should be handled as modules (e.g., a particular SCO is worth to be separately reused, replaced or updated), these shared resources impair modularity.

As the analysis has shown, SCORM does not fully meet the modularity requirements. What SCORM particularly lacks is the exposition and separate reusability of fragments. Although SCORM uses a content model, the elements within a SCORM package are not ready to be separately reused. Aggregation elements of a SCORM package can neither be individually retrieved from learning object repositories nor be immediately reused without the containing package. A module concept based on SCORM needs to reflect both, the theoretical demands of modularity and the practical situation of a variety of existing systems for creation, distribution and usage of learning resources. The module concept has to particularly establish a separate reusability of fragments of a learning resource, as this is a shortcoming of the current SCORM specification. Six requirements have been identified that have to be fulfilled in order realize modularity in SCORM. These requirements are

(1) compliance with original SCORM systems,

(2) fulfillment of modularity requirements (according to Section 4.2),

(3) support for individual metadata records per module,

(4) enabling modularity-awareness of repositories, and

(5) support for modular operations,

(6) allowing revisions and updates of modules.

The concept for modular learning resources extends SCORM in order to fulfill the requirements specified above. The resulting concept has three parts: first, it consists of a rather conceptual understanding of what modular learning resources should be like; this part is reflected mainly by the requirements and by some further properties that can be observed from the actual implementation. Second, the concept comprises a tangible realization in the form of a specification of an extension to

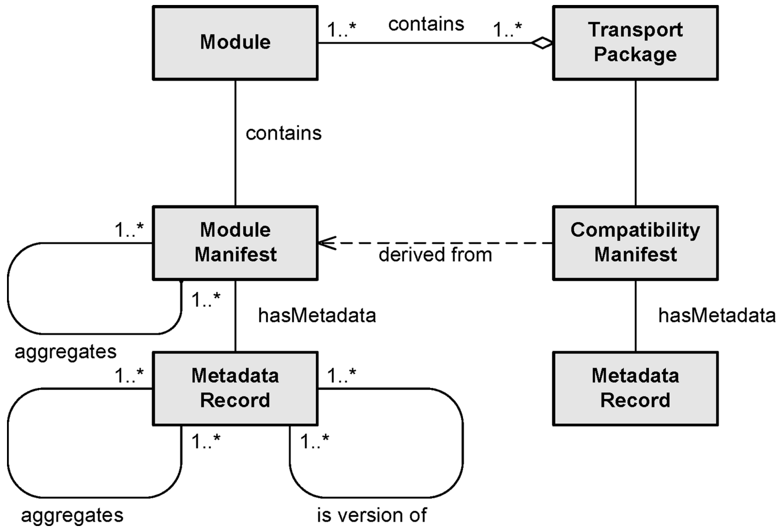


Fig. 4. UML diagram of module concept.

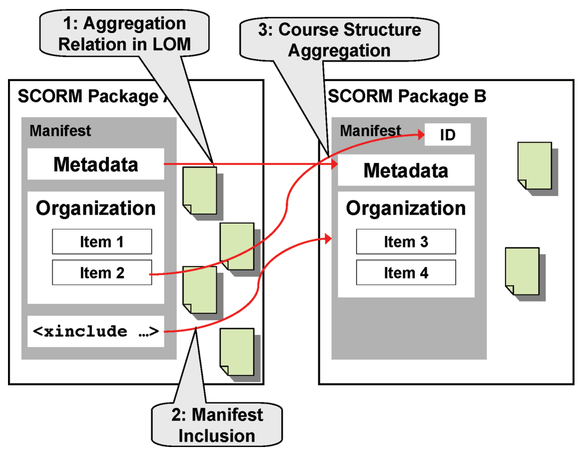


Fig. 5. Three levels of aggregation.

the SCORM format. And third, the concept includes also instructions for how modular operations on these learning resources can be performed.

The aggregation by reference of modules is performed on three layers (see Figure 5). These three layers serve different purposes. The first layer is the metadata layer: The LOM record of a module contains typed relation entries for all directly included modules. The relation entry contains only the identifier of an included module and can be used by repositories and other systems to easily identify which modules are to be delivered together as a bundle. The second layer, manifest aggregation, uses an option of the IMS Content Packaging (IMS CP) specification.[6] IMS CP optionally allows to use so called *xinclude* tags [Marsh and Orchard 2006]; xinclude is a mechanism for merging multiple XML

[6]IMS Content Packaging Best Practice Guide: Version 1.1.2 Final Specification: http://www.imsglobal.org/content/packaging/cpv1p1p2/imscp_bestv1p1p2.html.

documents into a single one. For the aggregation of modules, *xinclude* tags are used to merge all manifest documents of included modules into the aggregating module manifest. By integrating activities or SCOs of other modules into the course structure, aggregation of learning resources can be achieved. An important requirement is SCORM compliance in order to be interoperable with existing systems. Therefore, the modularity extension supports compatibility manifests. For compatibility reasons, it is also possible to process the inclusions and generate a single physical manifest file for the whole aggregated learning resource. This compatibility manifest can be stored in the root of a distribution package as a regular SCORM manifest.

Finally, we have realized the six modular operations in a SCORM specific way. The module concept has been described in more detail in Meyer et al. [2006].

### 5.3 A Generic Reference Model for Modularization Processes

One of the six modular operations, modularization, has also been researched from a more generic perspective. The focus of this section is postproduction modularization. Postproduction is concerned with the splitting of already existing learning resources into multiple modules. Some people use the term *modularization* for the systematic design and creation of modules from scratch. This meaning is not considered within this article. Modularization is regarded strictly as the process of splitting an existing learning resource into several modules. More precisely, it is the process of transforming a learning resource into an aggregation of multiple smaller learning resources. The necessity for modularization of learning resources is highlighted, for example, by Duval and Hodgins [2003], though they name this process *decomposition*. The difference between modularization and decomposition is that modularization results in the same learning resource with a higher degree of modularity, whereas decomposition produces multiple unconnected learning resources.

Some descriptions of modularization tools or modularization guidelines can be found in the literature [Doorten et al. 2004; Rostanin and Ludwar 2007; Fernandes et al. 2005]. However, these descriptions are difficult to transfer to other scenarios. Therefore, we have decided to develop a generic reference model for modularization processes. Humphrey and Feiler have defined a process as "a set of partially ordered steps intended to reach a goal" [Feiler and Humphrey 1992]. Splitting a learning resource into several smaller learning resources (modularization) could be such a goal; the activity of modularization may be called *modularization process* if it consists of partially ordered steps. According to Curtis et al. [1992], a *process model* is "an abstract description of an actual or proposed process that represents selected process elements that are considered important to the purpose of the model and can be enacted by a human or machine." In other words, a process model is a generic blueprint from which tangible, more detailed processes can be derived.

The creation of a generic modularization process strives for multiple goals. The first one is quality control: whether modularization is performed manually or automatically, the quality of modularization can be increased by sticking to a process model. The user is aware of the necessary steps and can keep the proposed order; furthermore, the execution of the individual process steps can be logged and provides an opportunity to be checked later for quality assurance. The generic modularization process can be concretized for particular formats, purposes and tools. A second goal is the development and optimization of modularization tools. Based on a modularization process a tool for supporting this process can be implemented. The existing tools mostly offer only a limited (merely technical decomposition) support of modularization. A process model can help to build tools that cover the whole modularization process. And finally modularization tools can be optimized by systematic analysis and improvement of process steps, which turn out to be either labor intensive or error-prone.

Our process model is based on an analysis of existing modularization methods found in the literature. A detailed discussion of these methods can be found in Meyer [2008]. Several tasks have been observed

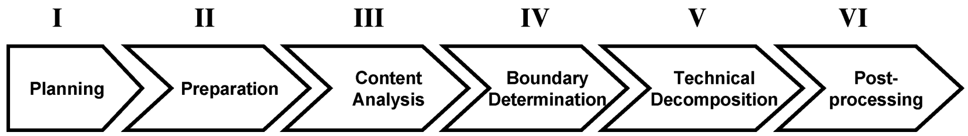


Fig. 6. Steps of the generic modularization process model.

in the descriptions of modularization methods. These tasks were grouped into coherent steps that resemble a partial order. The grouping of tasks has resulted in a generic process model for the modularization of learning resources (see Figure 6). The steps of the process model are *planning*, *preparation*, *content analysis*, *boundary determination*, *technical decomposition*, and *postprocessing*. The process model is designed as a linear sequence of six process steps. A linear order facilitates comparisons of different processes. Furthermore it ensures repeatability of results if process steps are always executed in the same order. Besides rearrangements of process steps, it may also happen that particular steps are omitted in implementations. The given order has been chosen because of potential dependencies between the involved tasks. For example, it is hardly possible to determine module boundaries without having analyzed the learning resource before. If the order of process steps in a process instance differs from the reference process model, the reasons for the change should be documented.

Each process step stands for different tasks that may be performed in that step, depending on the particular situation. There is no a priori list of potential tasks; the tasks to perform have to be analyzed and implemented depending on the environment and parameters of the modularization. The existing modularization approaches, which have been reviewed above, can be expressed as instances of the generic process model. For this purpose, the approaches have to be decomposed into tasks that each fit into one of the process steps.

The presented generic process model has been implemented as a proof-of-concept [Meyer et al. 2007]. The resulting application has proved to be suited for the modularization of SCORM learning resources.

### 5.4 Aggregation of Modular Learning Resources

There are many examples of tools in the area of e-learning, which enable users to aggregate learning resources. For instance, Sanchez and Sicilia describe a system that automatically aggregates learning resources, which in combination help to achieve a given learning objective [Sanchez and Sicilia 2004]. Aggregation is also strongly related to content models (cf. Section 3).

When learning resources are assumed to be structured, authoring tools can and should also support a structured authoring process. There are different kinds of authoring processes, depending on which and how many learning resources are produced, which and how many roles are involved and last but not least which tools are used. Authoring by aggregation is one of these authoring processes. As defined by Duval and Hodgins, authoring by aggregation means that learning resources "are created by selecting content/information objects from a repository, usually with the significant assistance of metadata and profiles to do so." These content objects (which are also learning resources according to our definition) are assembled to form a new learning resource [Duval and Hodgins 2003]. This approach has been realized as an authoring system by Hoermann in the ResourceCenter [Hoermann et al. 2005]. The ResourceCenter is an integrated repository and authoring tool, which is used mainly by university lecturers for creating Web-based trainings. The ResourceCenter has a proprietary content format and permits exporting courses into the SCORM format. A second system supporting decomposition and authoring by aggregation is the ALOCoM framework [Verbert et al. 2006].

Instructional design often follows the ADDIE model. ADDIE stands for five developments phases for the development of learning resources; the phases are *analysis*, *design*, *development*, *implementation,* and *evaluation*. The ADDIE model has evolved over time and has become commonly accepted as design principle [Molenda 2003].

Authoring by aggregation is a lightweight authoring approach that is particulary based on the selection and aggregation of existing learning resources. One characteristic of existing authoring by aggregation approaches is that they are narrowed to the development phase of the ADDIE model. Particularly, extensive instructional design tasks are left out in order to achieve lean tools.

However, including technical support for the first ADDIE phases—analysis and design—could improve authoring by aggregation. The support should be unobtrusive and optional in order to retain the light-weight characteristic of the authoring by aggregation approach. Thus it is a challenge to integrate analysis and instructional design in an unobtrusive way into authoring by aggregation processes.

There are two challenges that were identified for authoring by aggregation. The first challenge is to integrate further development phases (particularly the design phase) into authoring by aggregation systems. And second, information from the analysis and design phases should be utilized for retrieval of learning resources. This section introduces an extended authoring by aggregation process that takes the need for a separate design phase into account.

The authoring by aggregation process presented in this section is an evolution of existing processes. Duval and Hodgins [2003] have laid the foundation for authoring by aggregation. They describe the concept of authoring by aggregation as the creation of a learning resource by selection and composition of existing learning resources from a repository. Hoermann's authoring by aggregation process lets the author choose between selecting an existing learning resource and creating a new one for each position until the learning resource is complete. The concept of parallelizing reuse and creation has proved to be reasonable. Hoermann's process requires that the author creates the structure of a learning resource and the contents at the same time. But this concurrency prevents the author from thoroughly designing the didactic structure of the learning resource. This shortcoming is eliminated by an improved authoring by aggregation process: A separate didactic design phase is scheduled to take place before the content authoring.

In the didactic design phase the author creates only a placeholder; afterwards, he tries to retrieve a suitable existing module for this placeholder; if no suitable module exists, the placeholder remains empty and has to be filled with new contents later. A placeholder is described with various attributes that indicate intended instructional and topical properties. These attributes are available in the retrieval phase as additional information.

The improved process also contains a further phase for adaptation of included modules. The need for this adaptation phase originates from the heterogeneous scenario of this article: when modules from different sources are combined, the mosaic effect occurs and has to be corrected. The mosaic effect and adaptations are dealt with in more detail in Section 5.5. A further improvement to be introduced is the stronger integration of retrieval into the authoring by aggregation process.

The improved process is shown in Figure 7. In this process, the first four phases may be repeated in any order. The author may choose at any time to either edit the course structure, to replace placeholder items with existing learning resources, to adapt a learning resource or to create new content. When the author has decided that the learning resource is finished he invokes the publishing phase and thereby ends the process.

### 5.5 Adaptation and Unification of Learning Resources

A core idea of the SCORM specification is to enable reuse of SCOs. Ip et al. [2003] have identified that reuse is practically prevented by the *mosaic effect.* The mosaic effect means that different SCOs have

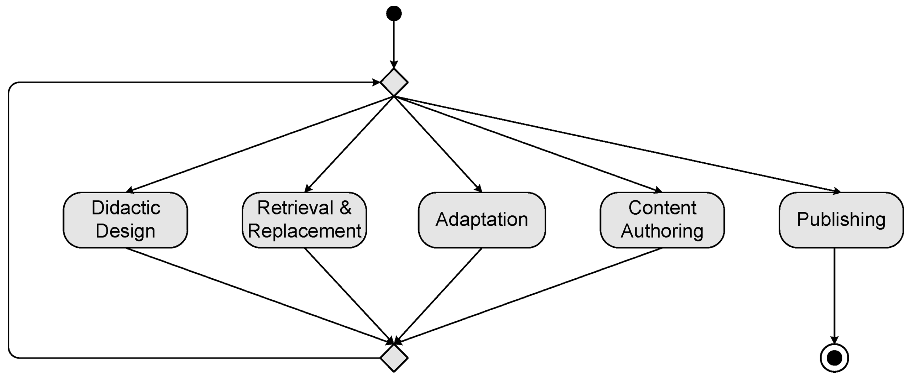


Fig. 7. Improved authoring by aggregation process.

a different 'look and feel'. If these SCOs are aggregated into one course, that course does not have a unified appearance.

Whoever wants to reuse a SCO has to manually change its look and feel. The solution proposed by Ip et al. is to separate contents and styles by rendering the contents of a SCO via stylesheets. However, as SCO stylesheets are still not frequently used, this solution cannot be applied to the adaptation of existing learning resources. Furthermore, the mosaic effect covers much more than only the visual presentation of a learning resource. Also the pedagogical style, the language, terminology, or the degree of interaction may differ. Thus, it is often necessary to adapt a learning resource in more than just one dimension. Performing an adaptation of a module is one of the modular operations specified in Section 4.2.

Zimmermann et al. [2006] have analyzed which kinds of adaptations of learning resources are frequently needed and, even more important, how they are performed by users. For the target group regarded by Zimmermann there are 15 important adaptation types, which can be separated into two categories: structured adaptations that can be supported by completely or partially automated tools, and unstructured adaptations that are difficult to support by such tools. Structured adaptation processes can be internally organized in three levels: top level adaptation processes, process fragments, and adaptation functions. An adaptation process is constituted of several process fragments; and a process fragment is divided into a number of adaptation functions. These adaptation functions are not dedicated to a single process or process fragment, but may be shared by several processes. Adaptation functions are, for example, identifying a graphical element, deleting a text fragment, changing a text or background color, or replacing one image by another one.

In order to support adaption and unification of learning resources, we have modeled these processes, fragments and functions as formal operations on multimedia documents [Meyer et al. 2007]. A repurposing framework has been developed to implement adaption and unification methods besides other methods for multigranularity reuse. More details about the framework and applications are given in Section 6.

## 6. VALIDATION

The concepts for multigranular reuse and modular learning resources have been validated by implementation in a prototypical application, the so-called Repurposing Tool Suite. The environment for this implementation has been the Content Sharing project. The Repurposing Tool Suite serves as a platform for different repurposing tools, such as a modularization tool, an aggregation tool and an adaptation

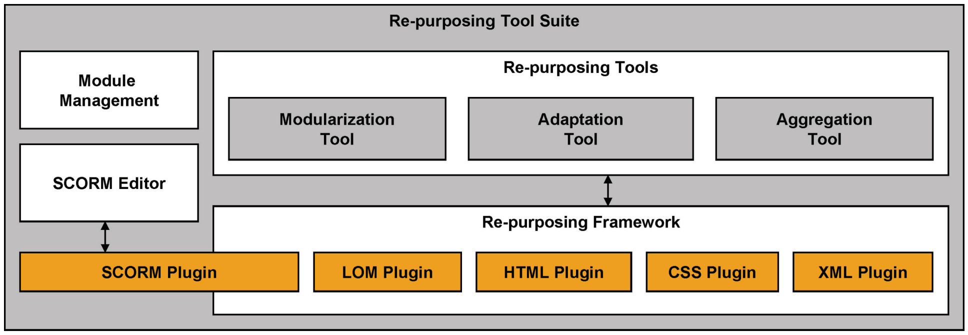


Fig. 8. Components of the Repurposing Tool Suite.

tool. The tool suite also provides support for the SCORM-based module concept from Section 5.2. The architecture and components of the too suite is illustrated in Figure 8.

The prototype implements the concepts for the base modular operations from the previous sections of this article: modularization, aggregation and adaptation of learning resources. These concepts are proposed solutions for achieving modularity of learning resources and especially multigranularity reusability. The question remains if the requirements for modularity and multigranularity reuse are fulfilled by the implemented prototype. The first part of this section reflects the implementation process of the Repurposing Tool Suite and feedback from users the Content Sharing project. The second part compares the requirements from the Sections 4 and 5 with the implementation in order to find out if the requirements could be achieved.

## 6.1 Lessons Learned from Development and Feedback

Over the duration of the Content Sharing project, several milestone versions of the prototype have been rolled out to selected users. The users have tested the prototype in their typical work environment. Besides a free test of functionality, the users were also given certain tasks, which they had to perform. Because of the low number of users, a statistical evaluation was not feasible. A detailed user study for the adaptation tool is currently conducted by a colleague. However, the previous feedback provided a qualitative assessment of the prototype in the consecutive development stages.

The result of the test were, that the latest version of the implemented prototype actually offered the expected functionality. Users were able to import, modularize and aggregate SCORM-based learning resources. They were also able to perform the supported adaptations on learning resources, which contained HTML documents as content documents.

The reactions from the users indicated that the usability of the prototype still has a potential for improvements. Especially invalid user actions should be detected or prevented earlier.

However, the most important result of the usability test has been the identification of a new user group: the nonauthors. Nonauthors are users who are not educated for content authoring and therefore do not have a technological background, such as knowledge about SCORM, HTML, or image formats. When the user group of reuse systems expands to also cover nonauthors, new requirements for reuse tools arise. Modularization and aggregation have to become more intuitive. Technical details, such as the SCORM nomenclature, have to disappear or to be replaced by colloquial language. New metaphors need to be found for enabling nonusers to naturally handle these tools.

This finding is notable, because the scenario of multi-granularity reuse apparently attracts potential users, who before did not engage in content authoring. Those users previously simple reused existing learning resources unchanged. When repurposing applications become available, they address a wider user community than only professional authors. Nonprofessional reusers want to benefit from these new tools, as well. Thus, repurposing applications have to meet the technical skills of this broadened user community.

### 6.2 Validation of Reusability Requirements

For a validation of the concepts, which have been realized in the prototype, the requirements for enabling modularity of learning resources and multigranularity reuse are compared to the functionality of the prototype. The proposed concepts are valid only if they (respectively their implementation) enable modularity and multigranularity reuse. This subsection focusses on functional aspects of the prototype. Usability aspects have already been addressed in the previous section.

6.2.1 *Modularity.* The first validation is the realization of modularity of learning resources. The definition of modularity in Section 4.2 consists of two necessary requirements towards a learning resource specification: first, the support of encapsulation, exposition, and separate reuse of parts of a learning resource; and second, the availability of six modular operations on modular learning resources.

The first requirement, encapsulation, exposition, and separate reuse of parts of a learning resource, concerns the property of a learning resource specification to statically allow a distribution of contents over several modules, and particularly to treat these modules also as self-contained, reusable units. This property is fulfilled by the implemented modular learning resource format. Contents of a learning resource can be indeed distributed over multiple modules, because a learning resource can be represented as an aggregation (by reference) of modules. Each of the modules, which are part of a whole learning resource, is again valid learning resource: it can be exported as an individual, self-contained SCORM package; and it can be aggregated into different learning resources.

While the first requirement resembles a static property, the second one concerns dynamic behavior. A learning resource specification is considered modular if it supports six defined modular operations. Actually, as a static format specification cannot perform dynamic behavior, support of reusable operators is more a system property than a specification property. Nonetheless, the specification has to ensure that a system can successfully perform modular operations. The six modular operations are evaluated one after the other to check if and how they are realized by the Repurposing Tool Suite as a reference system.

—*Modularization.* Modularization is supported by two dedicated modularization tools: the main modularization tool, and an additional extraction tool for media files. Both tools can be used to split a learning resource into multiple modules.

—*Aggregation.* The functionality of the aggregation operation is distributed over multiple components of the tool suite. According to the aggregation concept, the planning of an aggregated learning resource is done in the SCORM editor. The actual retrieval composition is performed using the integrated retrieval and replacement tool.

—*Exclusion (removal of modules).* Removing modules from a learning resource is simply performed by deleting the reference to that module in the SCORM editor. By deleting this reference, all other aggregation information is adjusted.

—*Replacement (substitution).* The replacement of one module by another one has to be executed manually in two steps. First, the old module has to be deleted; and second the new one is included. It would be desirable for further implementations to provide a tool, which facilitates, for instance, to replace a module by a variant or a newer revision of that module.

—*Reorganization (changing the order of modules).* The order in which modules are integrated into the overall learning resource can be changed in the SCORM editor.

—*Adaptation (transformation of a module).* Adaptation is supported by the adaptation wizard. Currently, five of fifteen possible adaptation types are technically supported. The remaining adaptation types are supported by pattern-based guidelines.

It can be concluded that all six modular operations are supported by the Repurposing Tool Suite. Some of the operations (e.g., modularization, aggregation, and adaptation) are realize more sophisticated, whereas others (e.g., replacement) require more manual work. Nonetheless, all six modular operations can be performed using the Repurposing Tool Suite.

6.2.2 *Multigranularity Reuse.* Second, the prototype has to be checked for its compliance with the requirements for multigranularity reusability. Multigranularity reusability has been defined as the general ability to reuse learning resources and their reusable fragments at all levels of granularity.

Multigranularity reuse can be assessed regarding this general definition. And there are also three properties, which constitute more detailed requirements towards reuse supporting systems. First, the general definition is considered. A system, which claims to support multigranularity reuse, needs to support reuse of learning resources and learning resource fragments at multiple levels of granularity; and furthermore, different types of reuse, namely simple reuse, aggregation, and reauthoring, have to be supported. The Repurposing Tool Suite supports modularization of an existing learning resource; each element within the organizational SCORM structure, and each media element can be transformed into a separate module. Thus, equating *reusable fragments* of the definition multigranularity reuse with structural SCORM elements and media objects, the condition of reusability at all levels of granularity as separate is met. These modules can be simply reused; they can be aggregated; and they can be reauthored, for example, by performing adaptations.

Three requirements specify in more detail what makes systems for multigranularity reuse usable in practice: availability, retrievability, and interoperability of modules.

—*Availability.* The Repurposing Tool Suite supports aggregation by reference: A learning resource can be constructed as a tree structure of modules, where the inclusion of contents of another module is embodied by a reference to that module. Learning resources, which are aggregated in this manner, can be exported and transferred to a repository that is able to handle the constituting modules at the same time as parts of the overall module and as separate modules. The central repository of the Content Sharing marketplace offers this functionality. Thus, the contents of a learning resource can be made available at multiple levels of granularity at the same time.

—*Retrievability.* By making modules available in the way just described, retrievability is already half achieved. Retrievability, though, requires two more properties, that go beyond mere availability of modules. The first requirement is the existence of adequate metadata for each module. Without suitable metadata, the modules can hardly be found. And second, retrieval methods have to make use of metadata about granularity in order to manage the growing amount of modules that come into existence by modularization.

Metadata generation is implemented in the present prototype only to a small extent. However, our new method for generating subject metadata [Meyer et al. 2008] could be integrated either into the tool suite or into learning object repositories.

—*Interoperability.* Interoperability regarding multi-granularity reuse particularly means that multiple modules can be jointly used, for example, by aggregation into a larger learning resource. It is evident that not all modules can be freely combined, because of topical and didactic differences. However, from a technical point of view, modules have to fit together under the condition that the contents are also compatible. Technical interoperability is ensured by the implemented prototype by two means: first by the module handling, which supports aggregation of modules; and second, the adaptation tool supports the transformation of modules towards a unified appearance (e.g., adaptation of design, language, or terminology).

All in all, the requirements of multi-granularity reusability are met by the implementations presented in this thesis. Most of the requirements are covered by the Repurposing Tool Suite. Additionally, retrievability can be improved by our new metadata generation method [Meyer et al. 2008].

## 7. CONCLUSIONS

This article has dealt with the modular reuse of learning resources within a heterogeneous scenario, under the assumption that learning resources are exchanged among participants who use different authoring systems. In such cases, Web-based learning resources are today exported into the SCORM format that is understood by almost any e-learning system (including authoring systems, learning management systems and learning object repositories). One disadvantage is that learning resources are no longer modular after this transformation. A definition even stronger than modularity is multigranularity reusability, which has been specified in this article. Multigranularity reusability in particular allows any relevant fragment of a learning resource to be reused as a module. SCORM does not fully satisfy the requirements of modularity and multigranularity reusability in this way. However, the SCORM specification has been an important milestone in the development of e-learning systems because it ensures compatibility. The specification of a completely new learning resource format would break the compatibility already achieved for e-learning systems. The overall goal of this article was to demonstrate how to enable and support modular and multigranularity reuse of learning resources with a special focus on heterogeneous systems; for compatibility reasons, solutions in particular have adopted the SCORM specification as their exchange format.

The overall goal has been pursued through four individual contributions. These contributions focused on the different requirements of multi-granularity reusability. A module concept based on the SCORM standard has been introduced. Solutions for three modular operations (modularization, aggregation, and adaptation) have been presented. More details about the presented topics can be found in Meyer [2008].

## REFERENCES

ADVANCED DISTRIBUTED LEARNING. 2004. Sharable Content Object Reference Model (SCORM) 3rd Ed.

ALONSO, G. 2004. *Web Services: Concepts, Architectures and Applications*. Springer.

BALDWIN, C. AND CLARK, K. 1999. *Design rules: Vol. 1, The Power of Modularity*. MIT Press.

BARRIT, C., LEWIS, D., AND WIESELER, W. 1999. CISCO Systems Reusable Information Object Strategy Version 3.0. White paper. http://www.cisco.com/warp/public/779/ibs/solutions/learning/whitepapers/el_cisco_rio.pdf.

BRASE, J., PAINTER, M., AND NEJDL, W. 2003. Completing LOM—how additional axioms increase the utility of learning object metadata. In *Proceedings of the 3rd International Conference on Advanced Learning Technologies*, V. Devedzic, J. M. Spector, D. G. Sampson, and Kinshuk, Eds. IEEE Computer Society, Los Alamitos, CA, 493–493.

CURTIS, B., KELLNER, M. I., AND OVER, J. 1992. Process modeling. *Comm. ACM 35,* 9, 75–90.

DOORTEN, M., GIESBERS, B., JANSSEN, J., DANIELS, J., AND KOPER, R. 2004. Transforming existing content into reusable learning objects. In *Online Education Using Learning Objects.* Routledge/Falmer, London, 116–1127.

DUVAL, E., FORTE, E., CARDINAELS, K., VERHOEVEN, B., DURM, R. V., HENDRIKX, K., FORTE, M. W., EBEL, N., MACOWICZ, M., WARKENTYNE, K., AND HAENNI, F. 2001. The Ariadne knowledge pool system. *Comm. ACM 44,* 5, 72–78.

DUVAL, E. AND HODGINS, W. 2003. A LOM research agenda. In *Proceedings of the 12th International World Wide Web Conference (WWW'03)*. 20–24.

FEILER, P. H. AND HUMPHREY, W. S. 1992. Software process development and enactment: Concepts and definitions. Tech. rep., ADA258465. DTIC Research Report.

FERNANDES, E., MADHOUR, H., MINIAOUI, S., AND FORTE, M. W. 2005. Phoenix tool: A support to semantic learning model. In *Proceedings of the 5th IEEE International Conference on Advanced Learning Technologies.* 948–949.

HODGINS, H. W. 2002a. The future of learning objects. In *e-Technologies in Engineering Education: Learning Outcomes Providing Future Possibilities,* J. R. Lohmann and M. L. Corradini, Eds. 281–298.

HODGINS, W. 2002b. IEEE 1484.12.1-2002, Draft Standard for Learning Object Metadata. http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf. (accessed 6/08).

HOERMANN, S. 2005. Wiederverwendung von digitalen lernobjekten in einem auf aggregation basierenden autorenprozess. Ph.D. thesis, TU Darmstadt.

HOERMANN, S., HILDEBRANDT, T., RENSING, C., AND STEINMETZ, R. 2005. ResourceCenter—A digital learning object repository with an integrated authoring tool set. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications (ED-MEDIA'05)*. AACE, 3453–3460.

IP, A., RADFORD, A., AND CANALE, E. 2003. Overcoming the presentation mosaic effect of multi-use sharable content objects. In *Proceedings of the 20th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education (ASCILITE)*. Vol. 1. 256–262.

KABEL, S., DE HOOG, R., WIELINGA, B., AND ANJEWIERDEN, A. 2004. Indexing learning objects: Vocabularies and empirical investigation of consistency. *J. Educ. Multimed. Hyperm. 13,* 4, 405–425.

KOPER, R. 2003. *Reusing Online Resources: A Sustainable Approach to E-learning*. Routledge, 46–59.

MARSH, J. AND ORCHARD, D. 2006. XML Inclusions (XInclude) Version 1.0 (2nd Ed.) - W3C Recommendation 15. http://www.w3.org/TR/2006/REC-xinclude-20061115/.

MEYER, M. 2008. Modularization and multi-granularity reuse of learning resources. Ph.D. thesis, TU Darmstadt.

MEYER, M., BERGSTRÄSSER, S., ZIMMERMANN, B., RENSING, C., AND STEINMETZ, R. 2007. Modeling modifications of multimedia learning resources using ontology-based representations. In *Advances in Multimedia Modeling*, T.-J. Cham, J. Cai, C. Dorai, D. Rajan, T.-S. Chua, and L.-T. Chia, Eds. Lecture Notes in Computer Science, Vol. 4351. Springer, 34–43.

MEYER, M., RENSING, C., AND STEINMETZ, R. 2006. Modellierung eines generischen Prozesses für die Modularisierung vorhandener Lernressourcen. In *Proceedings der Pre-Conference Workshops der 4. e-Learning Fachtagung Informatik DeLFI*. Logos, Berlin, 19–26.

MEYER, M., RENSING, C., AND STEINMETZ, R. 2008. Using community-generated contents as a substitute corpus for metadata generation. *Int. J. Adva. Media Comm. 2,* 1 (Jan), 59–72.

MEYER, M., ZIMMERMANN, B., RENSING, C., AND STEINMETZ, R. 2007. An interactive tool for supporting modularization of scorm-based learning resources. In *Proceedings of the World Conference on Educational Multimedia, Hypermedia and Communications (EDMEDIA)*. AACE, 3164–3171.

MOLENDA, M. 2003. In search of the elusive ADDIE model. *Perform. Improve. 42,* 5, 34–36.

NEVEN, F. AND DUVAL, E. 2002. Reusable learning objects: a survey of LOM-based repositories. In *Proceedings of the 10th ACM International Conference on Multimedia (MULTIMEDIA'02)*. ACM Press, New York, NY, 291–294.

NIEGEMANN, H., HESSEL, S., HOCHSCHEID-MAUEL, D., ASLANSKI, K., DEIMANN, M., AND KREUZBERGER, G. 2004. *Kompendium E-learning*. Springer.

OCHOA, X. AND DUVAL, E. 2008. Quantitative analysis of learning object repositories. In *Proceedings of the World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA)*. AACE, 6031–6048.

POLSANI, P. 2003. Use and abuse of reusable learning objects. *J. Digit. Inform. 3,* 4.

RENSING, C., BERGSTRÄSSER, S., HILDEBRANDT, T., MEYER, M., ZIMMERMANN, B., FAATZ, A., LEHMANN, L., AND STEINMETZ, R. 2005. Re-use and Re-authoring of learning resources—Definitions and examples. Tech. rep. KOM-TR-2005-02, Multimedia Communications Lab., Technische Universität Darmstadt.

ROSTANIN, O. AND LUDWAR, M. 2007. From informal learner to active content provider: SLEAM approach. In *Proceedings of the EC-TEL Poster Session*.

SANCHEZ, S. AND SICILIA, M.-A. 2004. On the semantics of aggregation and generalization in learning object contracts. In *Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies (ICALT'04)*.

STEINACKER, A. 2001. Medienbausteine für web-basierte Lernsysteme. Ph.D. thesis, Technische Universität Darmstadt.

SÜSS, C. AND FREITAG, B. 2002. LMML–The Learning Material Markup Language Framework. In *Proceedings of the International Workshop Interactive Computer Aided Learning*.

VERBERT, K. AND DUVAL, E. 2004. Towards a global component architecture for learning objects: A comparative analysis of learning object content models. In *Proceedings of the World Conference on Educational Multimedia, Hypermedia and Telecommunications*. 202–208.

VERBERT, K., JOVANOVIC, J., DUVAL, E., GASEVIC, D., AND MEIRE, M. 2006. Ontology-Based Learning Content Repurposing: The ALOCoM Framework. *Int. J. E-Learn. 5,* 1, (Special Issue: Learning Objects in Context). 67–74.

WILEY, D. A. 2000. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. http://reusability.org/read/chapters/wiley.doc. (accessed 6/08).

ZIMMERMANN, B., BERGSTRÄSSER, S., RENSING, C., AND STEINMETZ, R. 2006. A Requirements Analysis of Adaptations of Re-Usable (E-Learning) Content. In *Proceedings of the World Conference on Educational Multimedia Hypermedia and Communications (EDMEDIA)*.

Received January 2009; revised June 2009; accepted August 2009