DISS. ETH NO. 16062

MODULARIZATION AND STRUCTURED MARKUP FOR WEB-BASED LEARNING CONTENT IN AN ACADEMIC ENVIRONMENT

A dissertation submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of

Doctor of Technical Sciences

presented by

SAMUEL SCHLUEP

Dipl. El.-Ing. ETH

born 30.05.1964

citizen of Messen (SO)

accepted on the recommendation of

Prof. Dr. Dr. Helmut Krueger, examiner
Prof. Dr. Bernhard Plattner, co-examiner
PD Dr. Sissel Guttormsen Schär, co-examiner

2005

Ergonomie
Mensch
-Produkt
-Arbeit
-Systeme

Band 10

Samuel Schluep

Modularization and Structured Markup for Web-Based Learning Content in an Academic Environment

Shaker Verlag
Aachen 2005

Bibliographic information published by Die Deutsche Bibliothek
Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data is available in the internet at http://dnb.ddb.de.

Zugl.: Zürich, ETH, Diss., 2005

Printed in Germany.

ISBN 3-8322-4432-8
ISSN 1610-1898

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen
Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9
Internet: www.shaker.de • eMail: info@shaker.de

Foreword

The development of multimedia learning contents is time-consuming and expensive. Accordingly, the question raised at the beginning of every new e-learning project is whether elements of existing learning materials can be reused for a new educational context. But what are the requirements to produce truly reusable learning contents? The idea of using the flexibility and the power offered by computers today is very appealing, not only for the presentation of multimedia e-learning materials, but also for the management of such data. This concept, however, is not just related to technological problems. It also implies a new way of thinking about learning material that has consequences on the management of learning contents and possibly has an impact on the didactics and thus on the teaching as well. On the one hand, content management that supports the reuse of learning material may constrict the way learning contents are designed, but on the other hand, it may open up new possibilities that go beyond traditional linear structured textbooks. The XML technology allows for the separation of content and graphical layout which offers an interesting approach for the implementation of reusable learning contents.

The problems to be solved are well described by the following questions: Which are the relevant characteristics of learning materials that ensure their flexible reuse in various educational situations? The question certainly does not solely rest with the storage and retrieval of electronic learning materials. An important aspect in this case is the granularity of the contents. Small grained learning objects designed to be independent of a specific educational context are easier to reuse than large modular units. But how can contents be broken down into such elementary, “atomic” learning objects? For this purpose a model is needed which takes into account the way authors think about elementary learning content entities and the way teachers would handle such reusable materials. The contents may have been created by the teachers themselves or collaboratively with other authors. We may generally ask if it is possible to identify elementary learning entities which are suitable to be modularized and stored in an appropriate format and which can be retrieved from a learning content management system for reuse in diverse educational contexts.

Despite the many standardization activities in the field of e-learning, the great number of published learning objects today integrate contents, layout, navigation, and user interfaces, all of which are tightly intertwined. This results in learning objects that provide a particular presentation format for a specific educational context and thus prevent their reuse in varying learning contexts.

The present work is a successful approach to modularize learning contents, as described above, in order to produce learning objects which are completely self-contained. The key idea is the separation of content and layout which is the basic requirement for such elementary learning objects to be universally reusable. The main contribution of the learning content management system (LCMS) presented in this work is the implementation of a consistent modular learning object component model and the consequent separation of content and graphical presentation. It enables flexible composition of singular learning objects to a complete and didactically coherent learning application.

The author provides a substantial overview of the actual state of the art of development methods for e-learning materials and their underlying concepts. These methods establish a solid foundation for the powerful “dynamic Learning Content Management System” (dLCMS). The presented studies of initial e-learning projects using the dLCMS show the possibilities and limitations from the point of view of teachers and, above all, reveal its great potential for future development. Actually, the support to create self-contained, modular learning objects which separates content, graphical presentation, and didactics is a fundamental advantage of the LCMS presented here.

Hence, this work pursues the really simple idea, that it should be possible to create small and modular learning objects which separate content and graphical presentation and which can be aggregated into larger didactically coherent lessons and courses. As a proof of the significance of this idea, four case studies from diverse fields of academic teaching are presented.

The dLCMS, which is based on clear theoretical concepts and has been evaluated in the field, is undoubtedly worth follow-up by way of future research and development. The author’s conceptual model is a significant contribution to the ongoing debate about the possibility of interweaving e-learning materials originating from differing sources.

Zürich, May 2005

PD Dr. Sissel Guttormsen Schär — Prof. Dr. Dr. Helmut Krueger

Acknowledgement

I would like to thank many people have contributed to the success of this work over the last few years.

Prof. Helmut Krueger, head of the Institute of Hygiene and Applied Physiology, offered me the opportunity to work on this project. The interdisciplinary and stimulating research environment of his institute proved to be an excellent place to study not only the latest groundbreaking technologies, but also the implications of these technologies on humans in their daily work.

PD Dr. Sissel Guttormsen Schär's inexhaustible optimism was a great support and motivation for this project. Further, she was ready to discuss concepts and questions at almost anytime and she gave valuable feedback on the final publications.

Many thanks to Prof. Bernhard Plattner and Prof. Walter Schaufelberger, both program directors of ETH World, and Anders Hagström, ETH World project manager, for their commitment to the dLCMS project. Further, Prof. Brigitta Danuser's engagement was invaluable to get the project going.

My colleagues of the man-machine research group, Pamela Ravasio, Marc Arial, Hans-Jörg Zuberbühler, Christine Hitzke, Morten Fjeld, and Philippe Zimmermann were always ready for discussions and many of their ideas contributed to the success of this work. Many thanks also to all the members of the Institute of Hygiene and Applied Physiology for the inspiring atmosphere making my time at the institute one that I will always look back to with great pleasure.

I would like to thank Diana Hornung, Kristina Isacson, and Doris Tomaschewski for testing the dLCMS system and their critical feedback and suggestions for improvement. Marco Bettoni, Christoph Clases, and Verena Friedrich were a great help to evaluate the work presented here. Thanks to Reinhard Hess and Wolfgang Korosec from the ICT department for their support and encouragement.

Furthermore, Lori Katz has thoroughly proofread this writing. Her corrections and formulations were a great help.

Thanks to my friends and family and all who contributed to the success of this work, but whom I did not mention personally.

Special thanks go to my partner, Gaby Willi, for her continuous motivation and invaluable moral support throughout the whole project and especially during the writing of this book.

Table of Contents

Abstract

The basic idea of a *learning object* is that it is a small, modular and self-standing chunk of learning content, which can flexibly be reused and assembled to electronic courses. Today, learning objects come in a variety of types of learning resources (lectures, presentations, reference material, simulations) and data formats (HTML with JavaScript, PowerPoint, Flash, Java, etc.). Most of the learning objects are individually designed and styled, and navigational and user interface controls are directly integrated into the objects. This phenomenon prevents a coherent assembly into larger learning units due to the inconsistencies in the graphical and navigational design.

In order to be able to successfully assemble learning objects from various origins into larger learning units, these objects must have similar *granularity* and they must be *self-contained*. Further, a standard data format separating contents from their visual presentation and navigation is needed. *Learning object component models* define different levels of granularity and specify how the components can be aggregated. *Structured markup*, using XML-based languages, provides a means to separate contents and visual presentation. The present work investigates learning object component models, the process to create modularized learning components, and structured markup for web-based learning contents.

The *dynamic Learning Content Management System (*dLCMS*)* project, presented in this work, defines and implements a learning object component model and structured markup. The level of granularity proposed is based on, what we call, *didactic content types* (e.g. definitions, examples, exercises, simulations, self-assessments). The suggested basic structured markup schema uses traditional typographical elements such as headings, paragraphs, lists, tables, etc. Because of its simplicity, it is anticipated to be easily understood by content authors and to be readily convertible to possible future data formats. The dLCMS provides a separate markup schema for questions and tests.

Content authors are a key factor for the successful application of these concepts. To support the authors to divide contents into learning objects representing a single didactic content type, the *Learning Unit Development Guidelines* have been developed as part of the dLCMS project.

A qualitative evaluation of the dLCMS and the *Learning Unit Development Guidelines,* together with content authors in an academic environment, focused on the modularization of learning contents and the application of structured markup. Authors from three different scientific domains (natural sciences, social sciences, engineering sciences) and one author working in the ICT services department used the dLCMS to create a web-based learning unit for the education of students or university personnel. The authors' task was the development of a learning unit to teach a topic from their respective knowledge domain. The participants were free to choose the didactic strategy and methods which they believed would best suit their purposes. Additionally, a student evaluation of a learning unit composed of small, self-contained learning components was conducted. The evaluation investigated whether or not students perceive such learning units as didactically coherent.

As a result of the evaluation, it can be concluded that content authors in an academic environment generally understand the concepts of modularization. They are able to create self-contained modular building blocks of learning content which are based on didactic content types. However, the authors need support to divide learning content into learning objects representing a single didactic content type. Further research is needed to improve the *Learning Unit Development Guidelines*.

These findings suggest that the process of creating such learning objects may foster the didactic quality of the whole learning unit. The analysis of the contents with respect to the didactic content types helps authors to clearly structure the subject matter into small comprehensible learning steps.

The simple structured markup schema has been found to be sufficient, provided it contains markup elements for literature references and glossary entries. The approach to provide a separate markup schema for questions and tests is feasible, but the schema provided needs to be improved.

The results of the student evaluation show that students are able to easily detect the logical relationship between the self-contained learning objects in a learning unit. Thus, it may be concluded that it is possible to aggregate self-contained learning objects into larger didactically coherent learning units.

Hence, the dLCMS provides a simple and flexible component model. The granularity level of the basic building blocks is based on didactic content types which may be a basis to define a standard level of granularity. Together with the structured markup schema using standard typographical elements, and a schema for questions and tests, this framework allows contents from different sources to be coherently aggregated into learning units. As a benefit of such a system, different authors and institutions can define a corporate styling of their e-learning courses, even if the original contents come from sources from all over the world. However, it remains to be seen whether or not the *reusability* of e-learning contents will be improved in practice using such a framework.

Zusammenfassung

Learning Objects sind kleine, modulare Lernbausteine, die flexibel wiederverwendet und zu elektronischen Kursen zusammen gestellt werden können. Heute verfügbare Learning Objects verwenden unterschiedlichste Datenformate (HTML mit JavaScript, PowerPoint, Flash, Java, usw.) und repräsentieren unterschiedliche Arten von Lernmaterialien (z.B. Vorlesungen, Präsentationen, Vertiefungsmaterial, Simulationen). Die meisten Learning Objects sind individuell gestaltet und integrieren z.T. Navigation und Bedienungselemente. Die individuelle Gestaltung des grafischen Designs und der Navigation, sowie die unterschiedliche Grösse verhindern, dass solche Objekte zu grösseren, kohärenten Lerneinheiten zusammengestellt werden können.

Um Learning Objects unterschiedlichster Herkunft problemlos zu kombinieren, müssen diese eine ähnliche Grösse (*Granularität*) haben und *in sich selbst abgeschlossen* sein. Ferner wird auch ein standardisiertes Datenformat benötigt, das die Inhalte unabhängig von der grafischen Darstellung und der Navigation speichert. Sogenannte *Learning Object Component Modelle* definieren unterschiedliche Komponentenebenen und sie spezifizieren wie die Komponenten zu grösseren Einheiten zusammen gefügt werden können. *Strukturiertes Markup*, basierend auf XML, ermöglicht die Trennung von Inhalt und Darstellung. Die vorliegende Arbeit untersucht Learning Object Component Modelle, den Prozess zur Herstellung solcher modularisierter Lernkomponenten und strukturiertes Markup für web-basierte Lernmaterialien.

Das Projekt *dynamic Learning Content Management System (dLCMS)*, das hier vorgestellt wird, spezifiziert und implementiert ein Learning Object Component Modell und strukturiertes Markup für Lernmaterialien. Die vorgeschlagene Grösse der wiederverwendbaren Grundbausteine basiert auf *didaktischen Inhaltstypen* (z.B. Definitionen, Beispiele, Übungen, Simulationen, Selbsttests, usw.) Das strukturierte Markup-Schema für darstellende Lerninhalte verwendet die üblichen typografischen Elemente, wie Überschriften, Absätze, Listen, Tabellen, usw. Wegen seiner Einfachheit ist zu erwarten, dass es für Autoren leicht verständlich und einfach in allfällige zukünftige Formate zu übertragen ist. Für Multiple-Choice-Fragen bietet das dLCMS ein separates Markup-Schema an.

Die Autoren von Lerninhalten sind ein Schlüsselfaktur zur erfolgreichen Umsetzung dieser Konzepte. Deshalb wurden die *Learning Unit Development Guidelines* als Teil des dLCMS Projektes entwickelt. Diese sollen die Autoren unterstützen, Lerninhalte in kleine, in sich abgeschlossene Learning Objects zu unterteilen, die einen einzelnen *didaktischen Inhaltstyp* repräsentieren.

Eine qualitative Evaluation des dLCMS und der Learning Unit Development Guidelines mit Autoren in einem akademischen Umfeld untersuchte den Modularisierungs-

prozess und die Anwendbarkeit von strukturiertem Markup. Autoren aus drei verschiedenen wissenschaftlichen Disziplinen (Naturwissenschaften, Sozial- und Geisteswissenschaften, Ingenieurwissenschaften) und eine Autorin der ICT Dienste verwendeten das dLCMS, um jeweils eine web-basierte Lerneinheit aus ihrem Gebiet für die Schulung von Studenten und Personal zu entwickeln. Die Autoren waren frei in der Wahl der didaktischen Strategien und Methoden. Weiter wurde eine Lerneinheit, bestehend aus kleinen in sich selbst abgeschlossenen Learning Objects, von Studierenden evaluiert. Diese Evaluation untersuchte, ob die Lerneinheit als didaktisch kohärent wahrgenommen werden kann.

Aus den Ergebnissen der Untersuchung kann gefolgert werden, dass die Autoren in einem akademischen Umfeld die Konzepte der Modularisierung im Grossen und Ganzen verstehen und in sich selbst abgeschlossene, modulare Lernbausteine auf der Basis von *didaktischen Inhaltstypen* entwickeln können. Aber sie benötigen dazu gute Unterstützung. Weitere Forschung ist nötig um die Learning Unit Development Guidelines zu verbessern.

Die Resultate legen den Schluss nahe, dass der Prozess zur Erstellung von Learning Objects die didaktische Qualität der gesamten Lerneinheit fördert. Die Analyse der Lerninhalte zur Identifizierung der *didaktischen Inhaltstypen* unterstützt die Strukturierung des Lernmaterials in kleine, gut verständliche Lernschritte.

Unser Ansatz, ein einfaches Markup-Schema für darstellende Inhalte anzubieten, wurde durch die Ergebnisse bestätigt, vorausgesetzt das Schema enthält auch Elemente für Literaturreferenzen und Glossareinträge. Es konnte auch gezeigt werden, dass ein separates Schema zur Erstellung von Multiple-Choice Fragen ausreicht; das von dLCMS angebotene Schema muss jedoch verbessert werden.

Die Evaluation einer Lerneinheit durch Studenten hat gezeigt, dass diese den Zusammenhang einzelner in sich abgeschlossener Learning Objects in einer Lerneinheit leicht erkennen können. Daraus kann gefolgert werden, dass in sich abgeschlossene Bausteine zu didaktisch kohärenten Lerneinheiten zusammen gefügt werden können.

Mit dem dLCMS steht ein System zur Verfügung, das ein flexibles Learning Object Component Modell und strukturiertes Markup integriert und es erlaubt Lerninhalte aus unterschiedlichster Herkunft zu kohärenten Lerneinheiten zusammen zu fügen. Die *didaktischen Inhaltstypen* bilden dabei ein Mass, um eine Standardgrösse der Grundbausteine zu definieren. Dadurch können Autoren und Institutionen ihre elektronischen Kurse in einem einheitlichen Erscheinungsbild publizieren, selbst wenn Materialien aus unterschiedlichsten Quellen verwendet werden. Dass in der Praxis die Wiederverwendung von Lernmaterialen mit einem solchen System wirklich verbessert wird muss jedoch noch gezeigt werden.

1 Transfer to Practice

In the last decade, the concept of *learning objects* has been widely proposed to enhance the reusability of electronic learning materials. The idea of learning objects is based on the assumption that small pieces of learning content have greater potential to be reused in different educational contexts than large units, such as whole courses. While whole courses are designed to suit a specific target group of learners, learning objects may be used as building blocks to produce diverse learning units for different learner groups and learning objectives.

If learning objects are to be freely selected to create larger, coherent learning units, they must fulfill at least three criteria:

- Learning objects must be *self-contained*. If a learning object depends on another specific object, it can no longer be freely combined. Therefore, a learning object must not refer to other objects.
- Learning objects should have a similar size or *granularity*. If the size of learning objects varies considerably, e.g. one object represents a whole lesson while another object is a single image, then these objects cannot be easily combined to create a new course. For example, a single media element, such as a picture of the Mona Lisa, cannot be simply put next to an object representing a whole lesson, e.g. a course module on Leonardo da Vinci.
- Learning objects should have a *standard structure*. To help students to easily grasp the structure of a learning unit which is composed from different learning objects, the learning unit must provide a coherent appearance. Therefore, the single learning objects should have a standardized structure.

It is widely accepted that learners need a context to understand the meaning of learning material. Furthermore, it is assumed that the smaller the learning objects are, the more easily they will be reusable. However, the learning context that small, self-contained learning objects can provide is very limited. There are many controversial discussions as to whether or not small, self-standing learning objects can be simply joined to provide a learning context which enables students to grasp the logical relationships of the different aspects of a subject matter.

This work takes a closer look at *modularizing* learning contents into small, self-contained learning objects using a standard granularity level and it investigates how a standard structure for learning objects can be specified using *structured markup*. Then, an evaluation of these concepts from the point of view of learning content authors and students in an academic environment is presented.

Modularization

The standard level of granularity which we propose is what we call *didactic content types*. These are definitions, examples, exercises, simulations, self-assessments, etc. A good example to show how didactic content types can be combined to serve different learner groups' needs is the subject matter of statistics. Students of pedagogy, medicine, psychology, sociology, and economics need to learn the same theoretical concepts, definitions, and principles. Therefore, a learning object representing a definition, e.g. for "standard deviation", can be reused for students of different disciplines. However, examples which are used to illustrate the theoretical concepts should apply to the domain familiar to the student – one might want to present a patient population to medical students, while enterprise performance data would better suit the needs of students in economics. Using didactic content types, we can flexibly combine components with a high potential for reuse together with elements which apply more specifically to a particular scientific discipline.

The results of our evaluation suggest that this approach is feasible. The following points summarize the relevant results we have found:

- Content authors have no problems creating self-contained learning objects.
- Self-contained learning objects can be aggregated into didactically coherent learning units. Students are able to perceive the logical relationship of the self-contained learning objects in a larger learning unit.
- In general, content authors are able to create learning objects which represent a didactic content type.
- Although content authors are familiar with the didactic content type categories (definitions, examples, exercises, etc.), they need good support to divide the contents into learning objects representing these didactic content types. Further research is needed to improve the guidelines which we have developed in order to support content authors.
- The findings suggest that creating such learning objects may foster the didactic quality of the whole learning unit. The analysis of the contents with respect to the didactic content types helps authors to clearly structure the subject matter into small comprehensible learning steps.

Structured Markup

Separating contents from their visual presentation is an important concept to enhance the reusability of learning contents. *Structured markup* provides a means to define a standard structuring schema for contents. It enables the separation of contents and its graphical styling. Pieces of contents are marked up through tags which relate to the structural meaning of the pieces, e.g. text as a title, an introduction, an answer to a quiz, etc. This structuring is different from formatting, where typographical styles, such as color, typefaces, type size, spacing, etc., are directly applied to the contents. Structured markup serves as a flexible data format and combines contents and structural information. Using the structural information, machine processing can adapt the contents to different presentation contexts such as presentation media (web-based, paper, etc.) or corporate designs.

A *structured markup schema* defines the set of markup elements that may be used. However, no standard schema for general learning contents has been established yet. It seems to be difficult to define a standard which integrates the various needs of different teaching methods and contents.

Based on the results of our work, we suggest providing separate markup schemas for expository learning contents and special didactic issues, e.g. questions and test. The set of markup elements for expository learning contents that we propose is simple. It is based on the traditional typographical elements and contains the following items:

- Headings,
- Paragraphs,
- Lists,
- Tables,
- Images and multimedia elements,
- Basic inline elements: Emphasis, strong, superscript, and subscript,
- Links and references to glossaries and bibliographic information.

If the size of learning objects is based on didactic content types, no markup elements are needed for definitions, examples, exercises, etc. This information can be assigned to the learning object as a whole using metadata. The advantage of such a simple schema is that content authors are familiar with using these standard typographical elements. Furthermore, contents marked up by such a schema should be readily convertible to potential future data formats.

For Questions and tests we suggest using the *IMS Question and Test Interoperability Specification (QTI)* which provides a standard markup schema for multiple-choice type questions (IMS 2002a).

2 Introduction

The basic idea of a learning object is that it is a small, modular and self-standing chunk of learning content, which can flexibly be assembled to electronic courses. The learning objects available today are individually designed and styled, and navigational and user interface controls are directly integrated into the learning resources. Aggregating them to larger coherent learning units is hardly possible. To overcome these problems, component oriented models for learning contents as well as the separation of content, presentation and navigation are needed. The aim of the present work is an investigation of modularization, understood as the concept of a modular component model as well as the content chunking process to create such modular contents, and structured markup, which provide the basis for the separation of content and presentation, for web-based learning content. It focuses on the applicability of these concepts from the point of view of learning content authors in an academic environment. After presenting an overview of the state of the art, the contributions of this work include the design of a flexible learning content component model; the proposition to base the granularity for basic building blocks on didactic content types rather than on learning objectives; the proposition to use a simple structured markup schema which is based on traditional typesetting elements; the implementation of the proposed learning content components model and structured markup schema; the proposition of a content chunking method which supports content authors to divide learning contents into small modular components; the proposition of a qualitative evaluation of the new concepts from a learning contents author’s point of view; and the proposition of a student evaluation of a learning unit composed of self-standing, modular learning objects.

2.1 Reusability Strategies for Web-Based Learning Contents

The production of learning contents for computer-based training is demanding and expensive. It is therefore a necessity to reuse e-learning material as many times as possible. Usually, electronic courses are designed to teach specific learning objectives to specific learner target groups from a specific teacher’s or lecturer’s perspective. However, the design of whole courses constricts their reuse. Whole courses can only be reused for educational contexts that are similar to the ones for which the courses were originally designed. Courses are typically reused only by the authors who created them for the same target audience (Downes 2001).

Recently, the concept of *learning objects* has been proposed to enhance the reusability of e-learning contents, which resulted in extensive discussions and research world-wide. The basic idea of a learning object is that it is a small, modular and self-standing chunk of learning content, which can flexibly be assembled into electronic courses. These objects can be shared with other users. Through the selection and recombination of learning objects, courses can be created cost effectively to suit the desired educational context (Longmire 2000; Wiley 2000; Downes 2001; Hamel et al. 2002).

Learning objects are a new way of thinking about learning content. This thinking requires a paradigm shift for the creation of learning material. Authors of learning resources need to adapt their thinking of learning material, which traditionally meant whole courses or lecture notes (Chitwood et al. 2000; Nichani 2001; Polsani 2003).

There have been many standardization activities in the field of learning objects. Important standards are the IEEE Learning Object Metadata standard aimed at establishing a uniform way to describe learning contents (IEEE LTSC 2002) and the IMS Content Packaging specification to encapsulate learning materials into interoperable, distributable packages (IMS 2001a). The SCORM Content Aggregation Model (ADL 2001a) combines and harmonizes the standards and specifications of IEEE, IMS, and other organizations. It provides a consistent and functional reference model that enables *reusability, accessibility, interoperability*, and *durability* of web-based learning contents.

Despite the many standardization activities, there are no established specifications for the structure of learning content itself. Today, several learning object repositories give public access to a great number of learning resources (ARIADNE 2001; EducaNext 2004; McGee 2004; MERLOT 2004). The learning objects contained in these repositories come in a variety of types of learning resources (lectures, presentations, reference material, simulations, etc.) and data formats (HTML with JavaScript, PowerPoint, Flash, Java, etc.). Most of them are individually designed and styled, and navigational and user interface controls are directly integrated into the learning objects. Aggregating them into larger learning units is hardly possible due to inconsistencies in graphical and navigational design. This phenomenon prevents the presentation of aggregated learning objects to learners in a coherent way.

This problem was addressed by Duval at the ED-MEDIA conference 2004:

> *"Documents must die: As long as efforts for 'share and reuse' are based on a simple file oriented document model, they are bound to fail. We need a more sophisticated component oriented model (or set of models with a common metamodel) that will enable seamless integration of document fragments from diverse origins … An absolute requirement to make this work in any real sense is the separation of content, presentation, and navigation – unless we limit ourselves to authoring and consuming MS-PowerPoint files!" (Duval 2004)*

Hence, providing collections of "document centered" learning objects in large repositories is not enough to successfully enhance the reusability of learning contents. In conjunction with simple sharing, we also need to look at the properties of the learning objects themselves. Therefore, we formulate the following three principal strategies which should foster the reusability of learning contents (see also Figure 2.1):

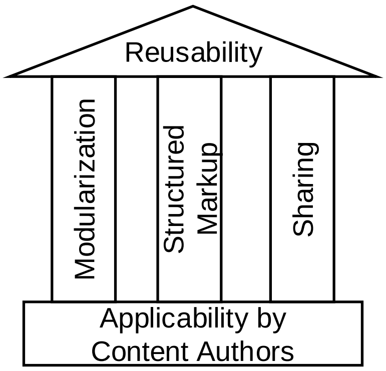

Figure 2.1 Reusability is based on three principal strategies: modularization, structured markup and sharing. Authors and teachers must be able to apply these strategies.

1. *Modularization:* Small, modular learning objects, which can be flexibly combined, allow for the reuse of learning contents in different educational contexts. The properties of these objects, e.g. size, didactics, etc. must be well defined to ensure that components from different origins can be assembled into larger, didactically coherent units. *Learning content component models* define different levels of components, the properties of these components, and how the components can be aggregated. Another important aspect is the process of dividing learning contents into such modular components, which is performed by the authors. We call this process *content chunking*. Here, *modularization* is understood as the concept of a modular component model as well as the content chunking process.

2. *Structured markup* serves as a flexible data format for the learning contents contained in the components. It implies an abstraction of the visual presentation and thus enables the separation of content, presentation, and navigation. Such contents can be easily reused in different presentation contexts, such as didactic scenarios, presentation media (web-based, paper, etc.), or corporate designs. The general structure of the contents is defined by *structured markup schemas*. Structured markup can be processed by machines. Using the structural information, the contents can automatically be adapted to various instructional and graphical contexts. However, the design of learning contents will be limited depending on the possibilities offered by the markup schema. Authors will need to understand the schema and adapt their contents accordingly.

3. *Sharing:* In order to be reused by a large community of users, learning objects need to be accessible. This necessity requires that interested parties

are able to find resources which suit their teaching needs and that resources are made available to be downloaded over a network or to be distributed by other means.

Much research has already been done on the topic of content sharing (South et al. 2000; ADL 2002; Koppi et al. 2003; McGee 2004; Najjar et al. 2004). Still, not many works can be found in literature which combine modularization and structured markup for learning contents. Therefore, the present work focuses on modularization and structured markup and the question of how a learning content component model and a structured markup schema should be designed.

Since we provide the first step in the life cycle of reusable modular contents, we also pay attention to the creation of modularized learning contents by learning content authors. As no such contents are available yet, the major goal of modularization and structured markup, of improving the reusability of learning contents, must be shown through future research.

In order to be flexible to the various teaching needs in an academic environment, we do not want to impose a single didactic model. Rather, we aim for modular learning contents to be (re-)usable in different didactic scenarios. Furthermore, while standards for pedagogical models are emerging, e.g. IMS Learning Design (IMS 2003), there is still no structured markup schema in sight for “pure” learning content which may be a true candidate for standardization. As we perceive a big need for such standards in the domain of web-based learning at our university, the focal point of this work concentrates on the modeling of learning contents for web-based learning.

In the subsequent sections of this chapter, we initially examine the relevant properties of modular learning objects. Then, a short overview of content chunking methods, which support content authors creating such learning objects, is presented. The next section contains an introduction to structured markup for learning contents. Then, the research of this work is outlined as a list of contributions. Finally, the outline section provides an overview of the subsequent chapters.

2.2 Modularization

In this section, we look at the relevant properties of learning objects or modular learning components to ensure that learning objects from different origins can be assembled into larger, didactically coherent units.

Teachers should be able to select and combine the learning objects which best suit their needs. Therefore, learning objects should be independent from one another. If a learning object can only be used together with another specific object, there will be no true choice and the advantage of flexible aggregation will be lost. Hence, learning objects must be *self-contained* or *self-standing* (Chitwood et al. 2000; ADL 2001a;

Hamel et al. 2002; Polsani 2003). Clearly, there will always be prerequisites to the learner's knowledge. It is therefore impossible to compile an object which includes all learning prerequisites. Here, we consider a learning object to be self-contained if it does not have any explicit references to other contents, such as links or linguistic references. Still, it may have prerequisites to the learner's knowledge.

An important attribute of modularization is *granularity*, i.e. the general size of the learning objects. There is no generally accepted specification for granularity. The IEEE LOM Metadata standard defines a learning object "as any entity, digital or non-digital, that may be used for learning, education or training (IEEE LTSC 2002)." This means a learning object can be of any size. Obviously, granularity has a big impact on reusability and the ability to aggregate learning objects into larger learning units. Therefore, it is generally accepted that they should be relatively small (Hamel et al. 2002), which is still a rather imprecise measure.

With smaller granularity, the potential for reuse increases as the pieces of information may be used in several different instructional contexts. For example, "a learning object can be a picture of Mona Lisa, a document on Mona Lisa (that includes the picture), a course module on da Vinci, a complete course on art history, or even a 4-year master curriculum on western culture (Duval et al. 2003)." On the other hand, the *context* in which the learning information is perceived has a strong influence on how the learner will interpret the contents and therefore on what will be learned. Longmire states "without context, learning objects can be confusing, misleading, or utterly meaningless (Longmire 2000)." With respect to granularity, there is a trade-off between reusability and the didactic value added through learning context.

As mentioned above, learning objects should be self-contained. This has an impact on learning context as well. It implies that learning objects must be de-contextualized. A key to the successful implementation of modularized learning content is to find ways to contextualize the information for the learner (Longmire 2000).

A level of granularity proposed by many researchers is to base learning objects on a single *learning objective* (Barritt et al. 2000; Longmire 2000; South et al. 2000; Baruque et al. 2003; LSAL 2003). Such an object has a sound didactic design and contains several pieces of information relating to the different didactic steps, such as giving an introduction (e.g. motivating information, presentation of learning objectives, overview), elaborating the topic (e.g. definitions, explanations, examples) and consolidating the lesson learned (e.g. exercises, feedback, assessment; see also Figure 2.2). This forces teachers to use the didactic design provided by the learning object. The didactic steps are an integral part of the learning object. They cannot be adapted to the differing needs of other target learner groups.

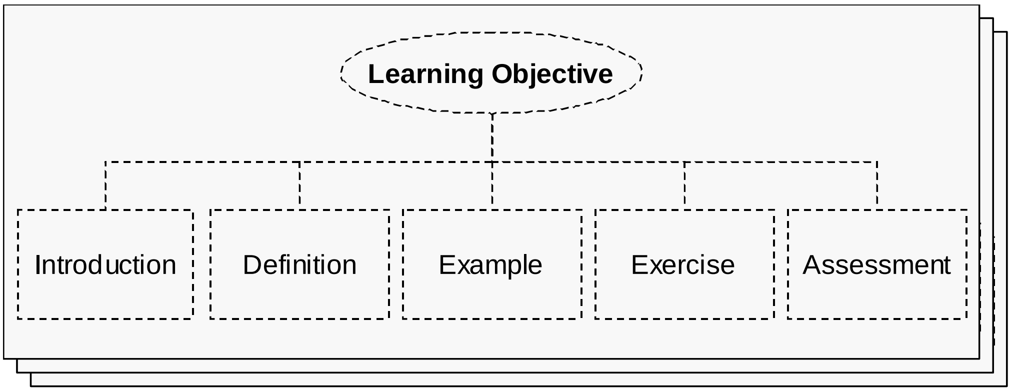

Figure 2.2 Many researchers propose that the granularity level of learning objects should be based on a learning objective. Such a learning object has a sound didactic design and contains information relating to the different didactic steps to teach the learning objective, e.g. an “introduction”, a “definition”, an “example”.

Another approach to a level of granularity supporting reuse may be based on, what we call, *didactic content types*, e.g. definitions, examples, exercises, simulations, self-assessments, etc. (Schulmeister 2003, see also Figure 2.3). A didactic content type may be seen as a piece of learning content which relates to one of Gagné’s nine instructional events (Gagné 1985).

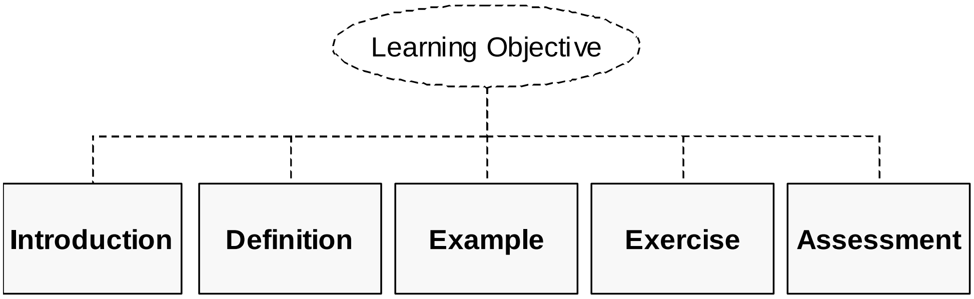

Figure 2.3 Learning objects whose granularity level is based on *didactic content types* (definitions, examples, exercises, etc.) can be flexibly combined to suit different learner groups’ needs. Components with a high potential for reuse can be used together with elements which apply more specifically to a particular scientific discipline.

A good example of how didactic content types can be combined to serve different learner groups’ needs is the subject matter of statistics. Students of pedagogy, medicine, psychology, sociology, and economics need to learn the same theoretical concepts, definitions and principles. Therefore, a learning object representing a definition, e.g. for “standard deviation”, can be reused for students of different disciplines. However, examples which are used to illustrate the theoretical concepts should apply to the domain familiar to the student – one might want to present a patient population to students of medicine, while enterprise performance data would better suit the needs’ of students of economics. Using didactic content types, we can flexibly combine components with a high potential for reuse together with elements which apply more specifically to a particular scientific discipline.

In order to be assembled into larger, didactically coherent units, we may conclude that learning objects should be *self-contained* and that they should have a similar level of *granularity*. Although a level of granularity based on learning objectives may yield a complete, didactically sound learning experience, such a learning object cannot be easily adapted to the differing needs of specific learner groups and restricts the use of the didactic design provided by the learning object. Therefore, we argue that a granularity level based on *didactic content types* is more flexible.

2.3 Content Chunking

Authors of learning contents find it difficult to create small, self-contained learning objects (Duval et al. 2003). Given the traditional method of writing text and creating learning contents, the development of de-contextualized learning objects is challenging. Using presentation software, like Microsoft PowerPoint, many authors might already have some experience in breaking down information into slides. However, we assume that authors will need additional support to create well designed, self-contained learning objects. Two design methods which explicitly deal with this issue are the “SCORM Best Practices Guide for Content Developers” and the ISDMELO methodology which are briefly presented below (a more detailed description can be found in Section 3.6).

In its “SCORM Best Practices Guide for Content Developers” (LSAL 2003) the Carnegie Mellon’s Learning System Architecture Lab (LSAL) suggests starting with an instructional strategy or with an existing course. Then, the learning objects should be identified based on well defined learning objectives and an analysis of potential audiences. The contents should be “divided” into *Sharable Content Objects (SCO)* which are optimally reusable for several potential audiences. Nonetheless, the LSAL still leaves open which level of granularity learning objects should have. The “roles” a SCO can play are: learning objectives in a lesson, segments in a lesson, lessons in a module, modules in a course, etc.

Baruque and Melo (2003) propose a methodology to create learning objects called ISDMELO, which is based on Instructional Systems Design (ISD). It claims to be grounded in sound pedagogical principles and to allow for the combination of principles from the major learning schools (behaviorist, cognitivist and constructivist). Using a top-down approach, the methodology analyzes the tasks and contents to be taught and breaks down the contents into different “elaboration levels”. The resulting learning objects are then classified by their learning outcomes, contents to be covered, evaluation method, example, practice, media and instructional approach. Thus, a single learning object is based on a learning objective and contains information related to different didactic content types.

In both methods presented, the level of granularity proposed is based on learning objectives. As our aim is for a granularity level based on didactic content types, the methods would have to be adapted accordingly.

2.4 Structured Markup for Learning Content

In order to aggregate learning contents from different authors into larger units with a coherent appearance, a method to standardize content structuring must be found. Traditionally, text is visually structured with the help of typography. Typography uses the elements of alignment, white space, typefaces, type size, etc. to establish a visual presentation of the content structure. For example, titles are presented with large type sizes, white space between lines serve as visual delimiters of paragraphs, etc. Consistent formatting is important for a reader to be able to discern the structure of a text and to comprehend its contents.

One way to standardize the content structuring is by separating contents and graphical styles using *structured markup*. Salo (2001) describes structured markup as follows: “break down a text into pieces ... name the pieces and determine how they fit together.” Generally, the naming of the pieces should identify meaningful structural information, e.g. a piece of text as a title, a paragraph, an introduction, or a citation, rather than styling information, e.g. a piece of text as 12 points in size. As such, structured markup represents an abstraction of the visual appearance and provides a means to separate contents and presentation.

Structured markup can be applied to contents using markup languages, such as HTML and XML. The different pieces of contents are marked up using tags which assign a name to the content fragments. The tagged fragments are called *elements*. The different types of elements can be distinguished by the name of the tags.

HTML (W3C 1999a) is the general markup language of the World Wide Web. Therefore, it is a natural choice for web-based learning contents as well. It provides a useful set of elements which more or less covers basic typography, hypertext links, and elements to embed external media objects. Unfortunately, HTML allows content creators to mix structured markup with graphical styling. The needs of web-publishers to create fancy graphical designs have resulted in a great number of techniques for “misuse” of the basic structuring elements in favor of sophisticated styling and layout. Hence, HTML does not truly separate content and presentation.

XML (W3C 2004a) is a general markup language for contents containing structured information. It provides a generalized syntax which can be extended to represent the specific structures of different kinds of data through *structured markup schemas*[1]. XML does not predefine a specific set of elements. Rather, it provides the means to create special-purpose markup languages by structured markup schemas which formally define a set of elements and the structuring rules.

Structuring contents with XML enables powerful machine processing. Individual elements of information and their structural relationships can be identified automatically which provide great flexibility to publish content. Content can easily be exchanged between different systems and the presentation of the various elements can be adapted to the system's particular method of presenting the information. For example, in one system a glossary entry might be displayed in a special box on the right side of the page, whereas in another system, the entry might be presented in a separate window. A rich set of tools is available to flexibly transform XML-based contents into other data formats. This feature supports cross-media publishing to different media, such as web, paper, etc.

XML can be used to specify markup languages that go beyond typography using tags that describe the type of content in a more meaningful way. For example, a markup schema for learning contents could specify tags for the didactic purpose of the content, such as definitions, examples, exercises, self-assessments, etc. In the past, some work has been done to define specific XML-schemas for learning content (Rawlings et al. 2002). However, no proposed schema has been established as a basis for further standardization to date. The wide range of learning content, the various didactics styles applied, and teachers' varying ideas of how to present learning materials may make it very difficult to find a general abstraction for didactical structures of learning contents.

Today, in most organizations, authors create unstructured text. To provide some visual structure they simply choose a larger font, or separate paragraphs with an empty line. Although structured markup might seem to add complexity to the authoring process, authors may find that it is easier to produce professional looking content because it enables them to focus on content rather than on format and design. But the typical author in an academic environment will not take the time needed to learn how to use structured markup if a structuring schema is complex and contains a very rich set of structuring elements. Still, without its consistent application, the advantages of structured markup are lost.

[1] The World Wide Web Consortium (W3C), the standardization organization for web technologies, provides two types of markup schemas: Document Type Definitions or DTDs (W3C 2004a) and XML Schemas (W3C 2004b). Here we refer to both of them using the notion "structured markup schemas".

Given the difficulty of specifying a general markup schema for learning contents which includes didactic markup, and the problem that authors use complex schemas inconsistently, we propose using a simple markup schema. Such a schema is based on typographical elements, e.g. headings, paragraphs, list, tables, etc. These elements have been utilized for a long time and authors are familiar with their uses. Further, because the schema is restricted to common elements used in many other markup languages, we assume the schema to be easily convertible to potential future data formats for learning contents.

While a traditional set of typographical markup may be very flexible, it seems clear that special didactic functionality is needed, such as questions and test. As specialized didactic markup might reduce flexibility of the basic structured markup schema, our approach is to provide a separate markup schema for didactic issues, e.g. multiple-choice questions.

2.5 Research Contributions

Focusing on modularization and structured markup and with a content author's perspective in mind, we present the modularization strategy combined with structured markup which we have developed. The evaluation of this strategy focuses mainly on the points of view of learning content authors. As this field is relatively new, we consider our work to be inductive research in order to generate hypotheses for further research and development. The main research contributions are detailed as follows:

- A flexible *learning content component model* has been specified, which defines different levels of learning components, the properties of these components, such as granularity, and how the components can be aggregated.
- We propose basing the granularity of the basic building blocks of learning contents on *didactic content types* such as definitions, examples, exercises, self-assessments, etc., rather than on learning objectives. We anticipate that this will promote the development of modular contents with a similar level of granularity. This scenario offers the opportunity to flexibly assemble building blocks into learning units that are well adapted for a target audience.
- In order to provide a flexible data format which separates contents and presentation and is easy to understand by authors, we propose providing a *simple structured markup schema* for learning content based on traditional typographical elements, such as headings, paragraphs, emphasis, etc.
- The proposed learning content components model and the structured markup schema have been implemented by the *dynamic Learning Content Management System (dLCMS)* project. This system provides a tool for the

development and storage of learning objects and for their assembly into whole courses. It further offers a platform to investigate the modularization and structured markup strategy from a learning content author's point of view.

- In an effort to assist content authors in dividing learning contents into self-contained modular building blocks based on didactic content types, a *content chunking method* was developed.
- A *qualitative evaluation* investigated the proposed learning content component model, structured markup schema and the content chunking method from a content author's point of view in an academic environment.
- A *student evaluation* of a learning unit that was developed for this study was conducted. It focused on the question of how students perceive the didactic quality of learning units which are based on self-contained modular learning components.

2.6 Outline

The remaining chapters are structured as follows:

Chapter 3 presents the state of the art of learning objects. It provides a detailed overview of existing learning content component models, structured markup schemas for learning content, and content chunking methods.

Chapter 4 describes the proposed learning content component model and structured markup schema and is summarized as an information model. It also contains a description of the functional architecture of the learning content management system which implements the information model and is based on an existing web content management system.

Chapter 5 presents the content chunking method which has been developed to assist content authors to create small modular learning objects.

Chapter 6 presents a pilot project study containing a qualitative evaluation of the proposed concepts from a learning content author's point of view in an academic environment.

Chapter 7 presents a student evaluation of one of the learning units which was developed in the pilot project study. It focuses on the student's perception of the didactic quality.

Finally, Chapter 8, summarizes the results and discusses the findings.

3 State of the Art

This chapter begins with a concise overview of the standard organizations in the field of learning objects. Then, learning object definitions found in literature are reviewed in order to define the relevant properties of learning objects. As the size of learning objects is an important issue to successfully aggregate learning contents from different origins into larger units, the next section examines different learning content classifications which may provide some insight into the structure of learning contents and possible levels of granularity. The learning content component models presented specify different levels of granularity and explain how the components on a lower level can be combined to form higher level units. In searching for a flexible data format which separates contents and presentation, different XML-based structured markup schemas for learning contents are reviewed. Finally, two content chunking methods are presented which should support content authors or instructional designers to break down contents into modular learning objects.

3.1 Learning Objects Standards

This section provides a concise overview of the standard organizations that are relevant with respect to learning objects.

IEEE LTSC

The *IEEE Learning Technology Standards Committee (IEEE LTSC)* develops technical standards, recommended practices, and guides for learning technology. We highlight two standards, the *Learning Technology Systems Architecture (LTSA)* and the *Learning Object Metadata (LOM)*.

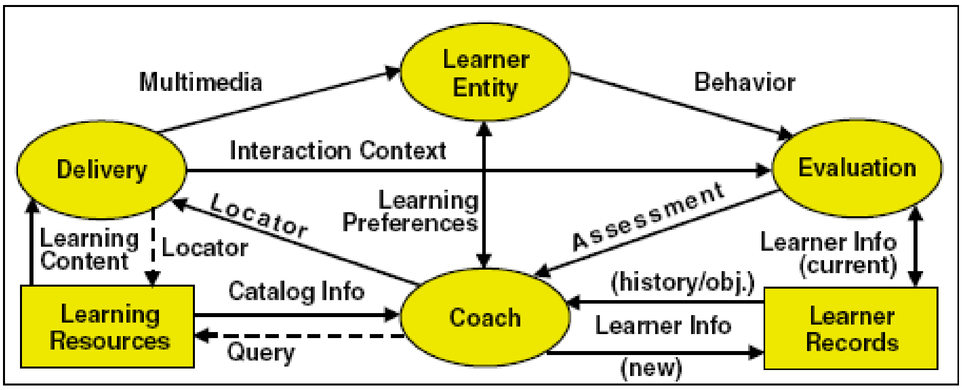

Figure 3.1 The LTSA components (IEEE LTSC 2001).

The LTSA specifies a high level architecture for information technology-supported learning (see Figure 3.1). The standard “(1) provides a framework for understanding existing and future systems, (2) promotes interoperability and portability by identifying critical system interfaces, and (3) incorporates a technical horizon (applicability) of at least 5-10 years while remaining adaptable to new technologies and learning technology systems (IEEE LTSC 2001).”

The LOM standard (IEEE LTSC 2002) specifies the syntax and semantics of metadata to facilitate the search, evaluation, acquisition and use of learning objects. About 80 metadata items describe relevant characteristics of a learning object and are grouped into nine categories: general, life cycle, meta-metadata, technical, educational, rights, relation, annotation, and classification. The standard provides a conceptual data schema, but it does not define a specific data format for the metadata.

IMS

The *IMS Global Learning Consortium* is a non-profit organization with contributing members from the global e-learning community. Its mission is to support the adoption and use of learning technology worldwide and it develops open technical specifications for interoperable learning technology. These specifications include:

- *IMS Content Packaging:* This specification (IMS 2001a) provides the functionality to describe and encapsulate learning materials, such as an individual course or a collection of courses, into interoperable, distributable packages.
- *IMS Digital Repository:* This specification (IMS 2000) defines digital repositories as being any collection of resources that are accessible via a network. The specification provides recommendations for the interoperation of the most common repository functions.
- *IMS Learning Design:* This specification (IMS 2003) provides a generic and flexible language to model a wide range of pedagogies in online learning.
- *IMS Learning Resource Meta-data:* This specification (IMS 2001b) provides an XML-binding schema for the IEEE LOM standard.
- *IMS Question & Test Interoperability (QTI):* This specification (IMS 2002a) defines a basic structure for the representation of question and test data which can be exchanged between Learning Managements Systems, content authors, and content libraries (for a more detailed description see Section 3.5).
- *IMS Simple Sequencing:* This specification (IMS 2002b) defines a method to represent the sequencing behavior of discrete learning activities in a consistent way. It contains the required behaviors and functionality that conforming systems must implement. It incorporates rules that describe the branching or flow of instruction.

ADL / SCORM

The *Advanced Distributed Learning (ADL) Initiative* (ADL 2001b) is sponsored by the US federal government in an effort to establish a new distributed learning environment for the government, industry and academia. The ADL initiative has defined high-level requirements to promote e-learning standardization for learning content: *reusability*, *accessibility*, *durability*, and *interoperability*.

An important contribution of ADL is the *Sharable Content Object Reference Model (SCORM)*. It defines a model for sharable learning content objects. Rather than creating a new standard, SCORM combines the standards and specifications of other organizations such as IEEE, IMS, and AICC (see below) to produce a consistent functional model. SCORM is divided into a collection of “technical books”: *Overview*, *Content Aggregation Model (CAM)*, *Run-time Environment (RTE)*, and *Sequencing and Navigation (SN)*. *Sequencing and Navigation* has been recently added the *SCORM 2004* version (ADL 2004). The *Content Aggregation Model* integrates:

- The *Content Model*, which is derived from AICC (the model is described in more detail in Section 3.3),
- *Content Packaging*, using the IMS Content Packaging specification,
- *Meta-data*, based on the IEEE LOM and the IMS Learning Resource Meta-data specifications,
- *Sequencing and Navigation*, integrating IMS Simple Sequencing.

AICC

The *Aviation Industry CBT Committee (AICC)* is an international association that develops guidelines for the aviation industry for the development, delivery, and evaluation of *computer-based training (CBT)* and related technologies. The objectives of the AICC are to:

- Assist the development of guidelines which promote the economic and effective implementation of CBT.
- Develop guidelines to enable interoperability.

However, AICC specifications go further than aviation and the AICC works together with other standard organizations, such as IEEE, IMS, and ADL.

ARIADNE

ARIADNE (Association of Remote Instructional Authoring and Distribution Networks for Europe) is an association of primarily higher education institutions in Europe whose aim is to share learning resources. The ARIADNE Knowledge Pool System is a distributed repository for learning and teaching resources. The most significant contribution of ARIADNE has been the development of a learning metadata schema which was the basis for the IEEE LOM specification.

3.2 Learning Object Definitions

This section presents an overview of learning object definitions in order to define the relevant properties of learning objects so as to ensure that learning objects from different origins can be assembled into larger, didactically coherent units.

The often cited definition of the IEEE LOM specification is: *"… a learning object is defined as any entity, digital or non-digital, that may be used for learning, education or training (IEEE LTSC 2002)."* This definition reflects the objectives of LOM to provide a standardized description for anything that could possibly be used for learning or education. As a definition for resources that may be used for web-based learning, this definition is too broad.

Wiley's definition attempts to be more specific: a learning object is *"any digital resource that can be reused to support learning (Wiley 2000)."* This definition should capture what the author understands to be the critical attributes of learning objects: *reusable*, *digital*, *resource*, and *learning*. Still, this definition does not sufficiently specify the relevant characteristics of learning content that enable learning objects to be aggregated into larger learning units.

Similarly broad is the definition of Polsani, *"a learning object is an independent and self-standing unit of learning content that is predisposed to reuse in multiple instructional contexts (Polsani 2003)."* Compared to the definition of Wiley, we find the additional attributes *independent* and *self-standing*. Polsani also points out that *learning* and *reusability* are to be the fundamental predicates of learning objects. Learning objects should be wrapped into a *learning intention* which has two aspects: (1) *form*, the framework in which a digital object is embedded, and (2) *relation*, an exposition or discourse guiding the learner towards the educational goal. Through its exchangeability, articulated here as reusability, the learning objects acquire value.

The idea of learning objects as modular entities of information which can combined into larger units of learning is expressed in the definition of the Wisconsin Online Resource Center, which defines learning objects to be *small learning units*, *ranging from 2 to 15 minutes*, which are *self-contained*, *reusable*, *tagged with metadata*, and *can be aggregated* (Chitwood et al. 2000). Here, some attempts have been made to define a standard granularity but whether a measure of time is a suitable approach to define the size of learning objects should be discussed. This definition also points out that learning objects should be described by metadata in order to be accessible to other interested parties.

A similar level of detail can be found in the "learning object design guidelines" presented by Hamel and Ryan-Jones: *"(1) Learning Objects must be units of instruction that stand alone, (2) Learning Objects should follow a standard instructional format, (3) Learning Objects should be relatively small, (4) A sequence of Learning Objects must have a context, (5) Learning Objects must be tagged and managed (Hamel et al. 2002)."* In order to be accepted and used by other instructional designers, the authors suggest the use of standard instructional formats which facilitate the creation of uniform quality learning objects. Furthermore, they point out that the aggregation of learning objects should provide a context that threads the objects into a coherent and meaningful larger learning unit.

Longmire lists the ideal attributes of reusable learning object content for an environment in which context is scalable and adaptive: *“(1) modular, free-standing, and transportable among applications and environments, (2) nonsequential, (3) able to satisfy a single learning objective, (4) accessible to broad audiences (such that it can be adapted to audiences beyond the original target audience), (5) coherent and unitary within a predetermined schema so that a limited number of metatags can capture the main idea or essence of the content, (6) not embedded within formatting so that it can be repurposed within a different visual schema without losing the essential value or meaning of the text, data, or images (Longmire 2000).”* In this description, we note the idea that a learning object should be designed to satisfy a single learning objective, and the idea of separating contents from formatting or presentation.

To end this overview of learning object definitions, which is not intended to be exhaustive, a final definition is presented without any further comment: *“... a learning object ... is basically a small ‘chunk’ of learning content that focuses on a specific learning objective (Baruque et al. 2003).”*

In conclusion, there seems to be a common understanding that learning objects should be relatively small, self-contained units of learning content which can be aggregated into larger learning units and which are reusable in multiple instructional contexts. However, the size of learning objects, an important factor in successfully aggregating learning objects from different origins into coherent larger units, is only vaguely defined. The determinations based on learning time (Wisconsin Online Resource Center) or learning objectives are arbitrary. Learning time varies from learner to learner depending on the individual’s learning characteristics. Learning objectives vary in the breadth of coverage of subject matter.

3.3 Classification of Learning Content

In order to define a level of granularity, which is based on the structure of learning contents itself, it might also be useful to consider different classification schemes for learning contents. This information may provide some insights into the general structure of learning contents and possible levels of granularity. In this section we present an overview of possible classifications which can be found in didactic theories, literature on writing educational materials, and learning object standards.

Gagné’s Nine Instructional Events

Gagné's work in identifying the conditions of learning, categorizing the learning outcomes, and specifying the nine events of instruction have contributed greatly to the field of instructional technology. His work initially dates back to 1965 but is still often referred to today.

Gagné’s book “Conditions of Learning” (Gagné 1985) stipulates that there are different types or levels of learning and that each type requires different types of instruction. Generally, five major categories of learning are identified: *verbal information*, *intellectual skills*, *cognitive strategies*, *motor skills* and *attitudes*.

Gangé’s theory outlines nine instructional events and corresponding cognitive processes for intellectual skills which are based on a cognitive model of a human's learning process:

1. *Gaining attention (reception)*
2. *Informing learners of the objective (expectancy)*
3. *Stimulating recall of prior learning (retrieval)*
4. *Presenting the stimulus (selective perception)*
5. *Providing learning guidance (semantic encoding)*
6. *Eliciting performance (responding)*
7. *Providing feedback (reinforcement)*
8. *Assessing performance (retrieval)*
9. *Enhancing retention and transfer (generalization)*

These events should satisfy or provide the necessary conditions for learning and serve as the basis for designing instruction and selecting appropriate media.

Component Display Theory

Merrill's Component Display Theory (Merrill 1983; 1987) is a strategy for designing instruction and is based primarily on the same assumptions as Gagne’s Instructional Events (see above). The Component Display Theory is an analysis that emphasizes different components of instruction for different types of instructional goals. It focuses on one single idea or objective at a time. The Component Display Theory provides a list of prescriptions for designing instructions for different kinds of instructional outcomes.

The Component Display Theory classifies learning along two dimensions: contents (facts, concepts, principles, and procedures) and learning performance (remembering, using, finding). This is represented as the “performance-content matrix” (see Figure 3.2).

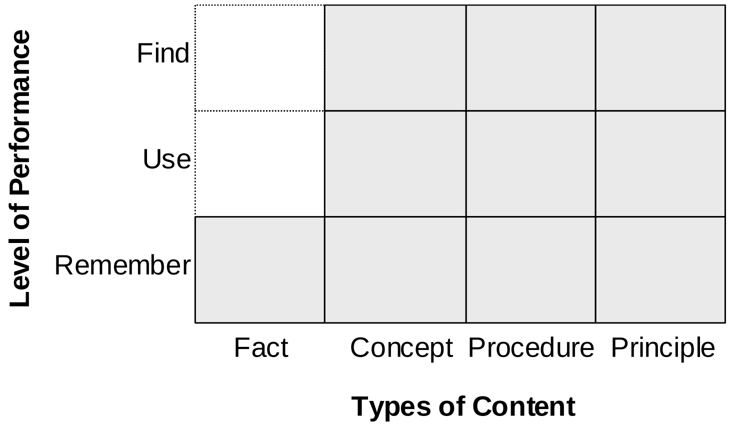

Figure 3.2 The performance-content matrix of the Component Display Theory (Merrill 1983).

The theory specifies primary presentation forms along two dimensions: the *generality – instance* dimension and the *expository – inquisitory* dimension. A *generality (rule)* is a statement of a definition, principle, or the steps in a procedure. An *instance (example)* is a specific illustration of an object, symbol, event, process, or procedure. *Expository* means to present, tell or show, while *inquisitory* means to question, ask, or require practice. Thus, the four primary presentation forms are: rules (*expository* presentation of a *generality*), examples (*expository* presentation of *instances*), recall (*inquisitory generality*) and practice (*inquisitory instance*). Secondary presentation forms are elaborations of the primary presentation forms. They represent information added to the primary presentation forms to enhance the learning that occurs.

According to Merrill, effective instruction should contain all necessary primary and secondary forms. Therefore, a complete lesson should be based on an objective followed by a combination of rules, examples, recall, practice, etc. which is appropriate to the subject matter and the learning task. The theory suggests that there is a unique combination of presentation forms for a given objective and learner, which results in the most effective learning experience.

Structured Writing

The *Structured Writing* method of Horn (1993; 1998) was developed as a comprehensive performance-based approach for instructional developers and business writers to prepare clear and concise training manuals, proposals, reports, and memos. It should enable managers, sales people, office personnel, and technicians to learn new products, services, and operating procedures rapidly and precisely.

The basic modules are called *Information Blocks*. An Information Block replaces the paragraph as the fundamental unit and is composed of one or more sentences and/or diagrams about a limited topic. It usually does not have more than nine sentences.

Information Blocks are always clearly identified with a label using one of the following categories: *analogy, block diagram, checklist, classification list, classification table, classification tree, comment, cycle chart, decision table, definition, notation, objectives, outlines, parts-function table, parts table, prerequisites to course, principle, procedure table, purpose, rule, specified action table, stage table, synonym, theorem, when to use, WHIF chart, who does what,* or *worksheet*.

A content analysis identifies seven information types *(procedure, process, concept, structure, classification, principle, fact)*. With the help of guidelines, this categorization is used to identify which key Information Blocks are necessary to fully understand a specific topic.

The underlying research focused on a deep understanding of the basic units of the subject matter and was aimed at providing an easy-to-understand taxonomy. The methodology claims to be capable of categorizing 80 percent or more of the content of virtually every subject matter that it has been applied to in the first pass.

Ballstaedt

Ballstaedt (1997) provides a classification of external representations of knowledge for paper based educational books. He differentiates *textual* and *non-textual* representations. With respect to textual representations, *oral* and *written* texts are distinguished. Written texts are further divided into the following categories:

- *Expository texts:* These texts contain a factual presentation of the subject matter to be taught. Such texts may contain definitions and explanations.
- *Narrative texts:* Narrative texts are subjective descriptions of personal experiences related to some subject matter.
- *Instructions:* Instructions provide a detailed description of how to perform a specific procedure step-by-step.
- *Supplementary didactic texts:* These texts are didactically motivated elements which support the learning process. Ballstaedt lists the following categories: *learning objectives*, *advanced organizers*, *summaries*, *examples*, *excursions*, *glossaries*, and *self-assessments*.

Non-textual representations of learning contents include: *charts*, *tables*, *diagrams*, *figures*, *icons*, and *maps*.

IEEE Learning Object Metadata (LOM)

The IEEE Learning Object Metadata Standard (IEEE LTSC 2002) provides two items which may be considered as describing the granularity level of learning objects.

The A*ggregation Level* metadata item aims at describing the functional granularity of learning resources. It specifies an enumeration of four types identified by a number. The aggregation levels are defined as follows:

1. *The smallest level of aggregation, e.g. raw media data or fragments.*
2. *A collection of level 1 learning objects, e.g. a lesson.*
3. *A collection of level 2 learning objects, e.g. a course.*
4. *The largest level of granularity, e.g. a set of courses that lead to a certificate.*

The *Learning Resource Type* describes the specific kind of a learning object by a pre-defined vocabulary. The vocabulary consists of the following terms: *exercise, simulation, questionnaire, diagram, figure, graph, index, slide, table, narrative text, exam, experiment, problem statement, self-assessment,* and *lecture.*

Discussion

Comparing the classification schemes described above, two levels of granularity can be found in two cases which use similar categories at the top level:

- Component Display Theory (Merrill): *fact, concept, principle, procedure*.
- Structured Writing (Horn): *procedure, process, concept, structure, classification, principle, fact*.

This top level classification of contents is used to analyze which second level content types are required for an effective learning experience. While the presentation forms of the Component Display Theory are rather abstract and theoretical, the Information Block categories of Structured Writing describe lower level contents in a less abstract way (*analogy, block diagram, checklist, classification list,* etc.). However, as it was developed for industrial use, this classification might not be adequate for learning contents in an academic environment.

Gagné's Nine Instructional Events do not provide a classification of the contents, but they do provide a level of granularity which may be useful in classifying contents from a didactic point of view on a level below the learning objectives into what we may call *didactic content types*.

Ballstaedt's classification of external representations of knowledge provides a schema based on an analysis of traditional educational books as they are used in academia today. This classification may also be considered useful to define *didactic content types*. However, categories for multimedia elements are missing.

The LOM *Aggregation Level* does not help to define different granularity levels more precisely. The terms "lesson" and "course" used to describe aggregation levels 2 and 3 are not very well defined. Furthermore, a level between levels 1 and 2 is missing for components which are composed from various media files but which cannot be considered to represent a lesson. From our perspective, the LOM *Resource Type* vocabulary is mixing up media *(diagram, figure, graph, slide, table)*, types relating to didactics (*exercise, simulation, questionnaire, narrative text, exam, experiment, problem statement, self-assessment),* and the overall form of presentation *(lecture).*

3.4 Learning Content Component Models

Beyond the learning object as the basic building block, we also have to specify how these building blocks can be aggregated into larger learning units and possibly how they integrate basic media elements. This implies that different levels of contents need to be specified. Learning content component models contain different levels of content components and they specify how the components on lower levels can be combined to form higher level units. Therefore, learning content component models allow for a more precise definition of the different levels of granularity. In this section, seven “learning objects components models”, which have been published and which we found to be important, are presented and discussed.

NETg Learning Object Model

The NETg Learning Object Model was one of the first models using the learning object concept (L'Allier 1997). It outlines how independent learning objects at different levels can be aggregated.

The model basically provides four levels: a *Course* contains independent *Units*, a *Unit* is made up of independent *Lessons*, a *Lesson* combines independent *Topics* (see Figure 3.3).

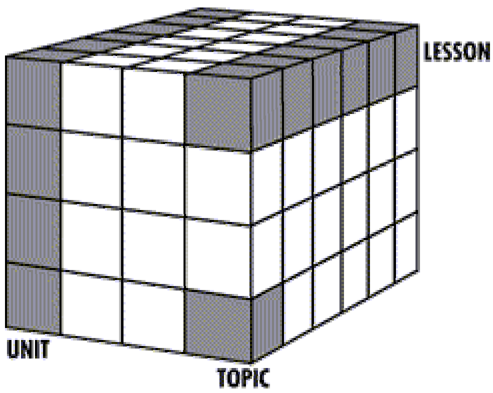

Figure 3.3 The NETg Learning Object Model: a Course consist of Units, Lessons and Topics (L'Allier 1997).

The contents of a *Topic*, also called a *learning object*, are made up of a *learning objective*, a *learning activity*, and an *assessment*. The structure of a Topic is based on the work of Bloom et al. (1956) with respect to the learning objectives and the work of Merrill (1983; 1987) with respect to a presentation model of contents.

Although the component levels of Course, Unit, Lesson, Topic are clearly specified, NETg’s Learning Object Model provides only an abstract definition of the contents of Topics.

Cisco Reusable Learning Object Strategy

The Cisco *Reusable Learning Object (RLO) Strategy* (Barritt et al. 2000) provides a well defined learning component model and represents a structured approach to create and categorize content based on five information types: concept, fact, process, procedure, and principle. This basically refers to Merrill's Component Display Theory (Merrill 1983) and adds the information type "process".

The RLO Strategy defines two components: the *Reusable Learning Object (RLO)* and the *Reusable Information Object (RIO)*. A RIO is a granular, reusable chunk of information that is media independent. Each RIO is classified into the five *RIO types*: *concept*, *fact*, *process*, *principle* or *procedure*. A RIO can stand alone as a collection of *Content Items*, *Practice Items* and *Assessment Items* that are based on a single learning objective. It should be noted that Content Items and Practice Items are not defined as individual reusable components. These items are "hard-wired" into a RIO.

A RLO combines 7 ± 2 RIOs, an overview, a summary and an assessment (see Figure 3.4). A RLO teaches a single objective which is derived from a specific job task. The RLO's major objective should be supported by the objectives of each RIO that it contains.

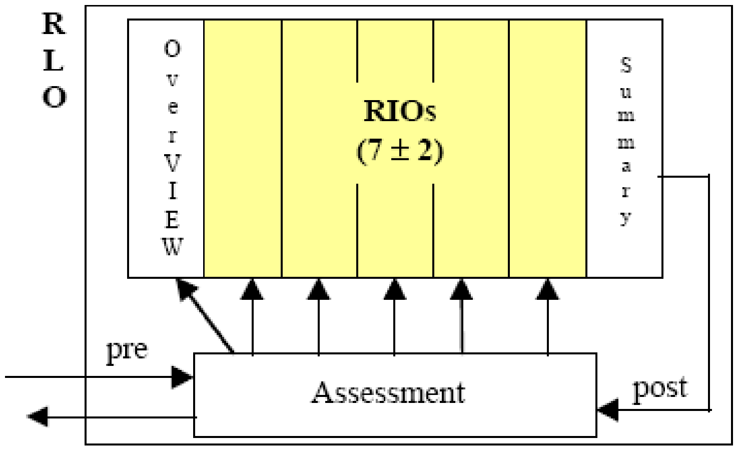

Figure 3.4 The Cisco Reusable Learning Object (RLO) combines 7 ± 2 Reusable Information Objects (RIOs), an overview, a summary and an assessment (Barritt et al. 2000).

From the learners' points of view, a RIO can be used as a stand-alone support tool, or just-in-time training coach. Depending on the delivery context, a RIO is also called a "page" or "job aid", which suggests that a RIO is presented on a single page when it is delivered over the web.

A RLO is also called a "lesson". It provides the learning context together with the knowledge and skills needed for the given objective and a method to assess mastery.

The RLO Strategy provides detailed guidelines to build RLOs and RIOs. For each of the RIO types, as well as for the RLO Overview and the RLO Summary, the guidelines describe which different types of Content Items are required and which ones may be used optionally (see Table 3.1).

Table 3.1 Overview of the Content Items to be used for the RIO types, RLO Overview, and RLO Summary; (r) = required, (o) = optional, (e) = either (at least one is required).

RIO Type	Content Items
RLO Overview	Introduction (r), importance (r), objectives (r), prerequisites (r), scenario (o), outline (r)
RLO Summary	Review (r), next steps (o), additional resources (o)
Concept RIO	Introduction (r), facts (o), definition (r), example (r), non-example (o), analogy (o), instructor notes (o)
Fact RIO	Introduction (r), facts (r), instructor notes (o)
Procedure RIO	Introduction (r), facts (o), procedure table (e), decision table (e), combined table (e), demonstration (o), instructor notes (o)
Process RIO	Introduction (r), facts (o), staged table (e), block diagrams (e), cycle charts (e), instructor notes (o)
Principle RIO	Introduction (r), facts (o), principle statement (o), guidelines (r), example (r), non-example (o), analogy (o), instructor notes (o)

While the Cisco model provides a well defined structure of learning object components, it contains only two independent component levels. It is strongly influenced by industrial training and it may be too restrictive for learning content in an academic environment.

Learnativity Content Model

The Learnativity Foundation has developed a content model to visualize how contents can be organized for e-learning and knowledge management applications (Wagner 2002). The model consists of five levels containing: *Content Assets*, *Information Objects*, *Learning Objects*, *Learning Components*, and *Learning Environment* (see Figure 3.5).

- *Content Assets* are the smallest level and relate to the raw media: images, text snippets, audio clips, applets, etc.
- *Information Objects* are classified as concepts, facts, procedures, principles, command references, exercises, or procedures.
- *Learning Objects* are formed by assembling Information Objects to teach a common job task on a single learning objective.
- *Learning Components* contain several Learning Objects which are bundled and sequenced to form “lessons” or “courses”.
- *Learning Environments* are obtained when Learning Components are wrapped with additional functionality such as communication tools, peer-to-peer computing and other community-of-practice-specific support, and the coming wave of modular web services learning components.

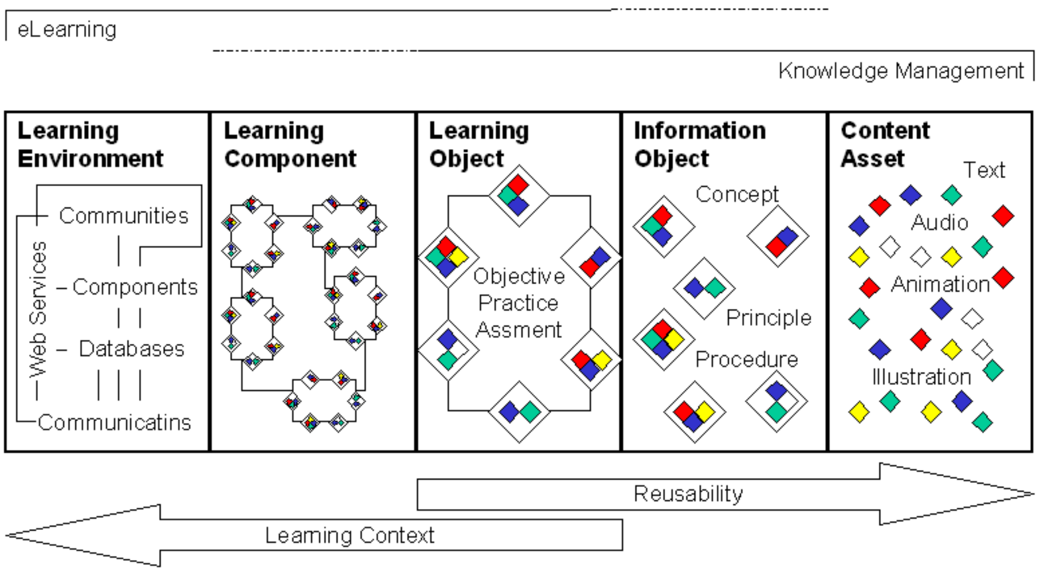

Figure 3.5 The Learnativity content model consisting of five levels of components: Content Asset, Information Object, Learning Object, Learning Component, Learning Environment (According to Wagner 2002).

The Learnativity content model aims at blending pedagogic and technical perspectives. The term learning objective as an instructional design concept is derived from Gagné (1985), Bloom et al. (1956) and many others. The model also incorporates concepts found in the Cisco RLO Strategy (Barritt et al. 2000).

With its five levels of granularity, the Learnativity Content Model provides a comprehensive specification of the different layers learning content component models may contain, including the raw media level. However, while in other models the components which are classified as concepts, facts, procedures, principles, or procedures relate to the level used to teach single learning objectives, here the classification is applied to components on a level below, the Information Objects.

PaKMaS

The *Passauer Knowledge Management System (PaKMaS)* is a hypermedia based knowledge management system which provides search, editing, evaluation and exchange facilities for learning material to teachers and students (Süß et al. 2000). It provides a well defined structure and a means to separate contents and presentation. It has been developed for academic environments.

PaKMaS utilizes the principle of modularization to support the creation and the maintenance of learning material. The contents are organized as a network of *Basic Modules* which are linked together by different types of references. PaKMaS is designed as a database to which both teachers and students have access.

Conceptually, learning content is divided into *Conceptual Entities*. Depending on the subject, different types of Conceptual Entities, e.g. courses, trainings, introductions, and summaries, can be distinguished. The granularity of a conceptual entity is not fixed. Nonetheless, a definition is provided: a conceptual entity is the smallest, self-contained building block which thematically and didactically sufficiently covers a topic.

Conceptual Entities are composed of *Content Objects*. Content Objects can be textual contents, e.g. definitions or proofs, and multimedia elements, such as illustrations, animations, sound, or video. Although from a conceptual point of view, Content Objects represent the elementary building blocks, the system stores Conceptual Entities as a whole in the database.

Conceptual Entities may contain references to other entities. Different kinds of references are provided, such as simple references, references to illustrations or proofs.

From a technical point of view, the smallest entities are called *Basic Modules*. They represent the Conceptual Entities. *Structuring Modules* combine Basic Modules into larger structures (see Figure 3.6). Structuring Modules may be used to combine selected contents for a specific target learner group or a specific instructional context (*guided tours*). However, Structuring Modules may also be used to organize all relevant modules of a specific topic into chapters, subchapter and sections.

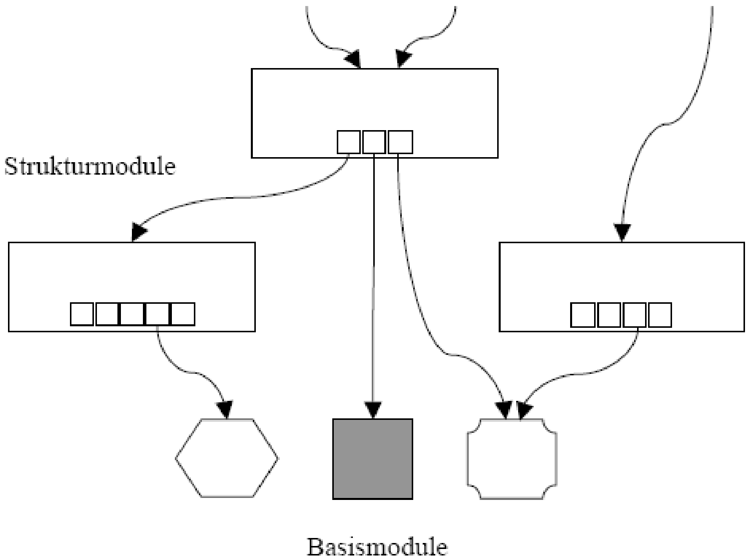

Figure 3.6 The PaKMaS modular technical structure containing Basic Modules (Basismodule) and Structuring Modules (Strukturmodule. Süß et al. 2000).

PaKMaS supports the separation of contents and presentation. The contents of the Basic Modules are stored as *LMML (Learning Material Markup Language)* documents. LMML is based on XML and is discussed below (see Section 3.5).

PaKMaS is an interesting approach to show how components can be combined to be accessed either thematically or using guided tours. It further supports the separation of contents and presentation using its own markup language. However, given its design as a large knowledge base to which teachers and learners have access, and given its possibility to specify cross-references between different Basic Modules, the modular contents cannot be considered to be self-contained and cannot be used outside the PaKMaS context. The learning contents are further stored in a document centric manner and the granularity level is not well defined.

"Methodenlehre-Baukasten"

The project "Methodenlehre-Baukasten" (building blocks for teaching research methods) is targeted towards a computer based learning environment used to teach research methods and statistics to students in human and social sciences (Schulmeister 2003). It provides an approach that enables one to adapt learning contents to the different needs of students from different scientific disciplines.

The project is considered to be represented by a complex structure using three dimensions: *Modules*, *Disciplines*, *Components*. Figure 3.7 shows a schematic representation of the "Methodenlehre-Baukasten" system as presented by the authors. They use the terms "modules" and "building blocks" synonymously.

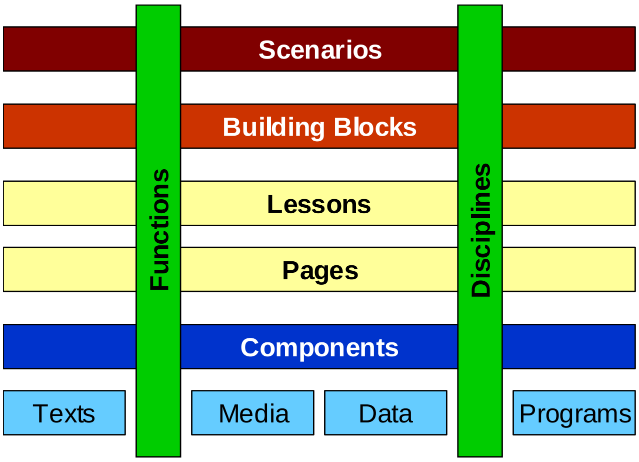

Figure 3.7 Schematic representation of the "Methodenlehre-Baukasten" system (According to Schulmeister 2003).

Modules are the building blocks which cover specific themes, for example descriptive statistics, data collection methods, practical training on empirical methods, etc.

Inside the Modules different *Disciplines*, such as pedagogy, medicine, psychology, sociology, and economics are differentiated. This defines a structure orthogonal to the modules dimension.

The Modules use a large number of *Components* (also called "learning objects"). These Components are multimedia objects, texts, exercises, media, programs, research examples, and data sets.

The system automatically combines texts, exercises, media, and data examples onto a *Page* (see Figure 3.8). The Pages may be composed dynamically depending on the learner, i.e. the student's discipline, and the instructional context. Therefore, all components need to be described by metadata. For example, a student of medicine may be presented a Page composed of four components: a title, a movie showing a medical examination, data from a medical examination, some text and an interactive exercise. For a student of psychology, the movie, the text, and the data will apply to the student's domain, whereas the title and the exercise remain the same.

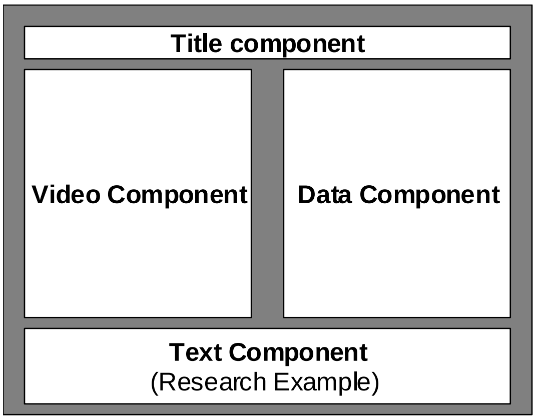

Figure 3.8 A Page layout of the "Methodenlehre-Baukasten" system: metadata defines which Components are displayed (According to Schulmeister 2003).

The lessons are based on a didactic structure which consists of several steps. Each Lesson starts with an advanced organizer and a motivating real life example which is illustrated graphically. Then, an easy-to-understand preliminary definition is given. The definition is immediately followed by interactive exercises with increasing complexity. An important element is the terminal exercise with real research data and an interpretation of the results. Schulmeister points out that interactive exercises play an important role in this concept. Hands-on experience should foster a student's understanding of difficult subject matter.

As an applied e-learning project, the "Methodenlehre-Baukasten" is an interesting approach in which the individual learning steps are presented as pages that are dynamically adapted to a student's scientific discipline. Pages are composed of examples, statistical data, interpretations, etc. which are selected according to the student's domain.

SCORM Content Model

The SCORM content model (ADL 2001a) is probably the most popular learning content component model and has been established as a standard. It contains the following components: *Assets*, *Sharable Content Objects (SCO)* and *Content Aggregations* (see Figure 3.9).

Assets are electronic representations of media, text, images, sound, web pages, assessment objects or other pieces of data that can be presented by a web client.

A *SCO* represents a collection of one or more Assets and includes a specific "launchable" Asset that utilizes the SCORM Run-time Environment to communicate with a *learning management system (LMS)*. To be reusable, a SCO should be independent of learning context and must not contain any links to other SCOs. A SCO may be reused in different learning experiences to fulfill different learning objectives. SCORM does not impose any particular constraints on the exact size of a SCO. A SCO should be the smallest logical unit of instruction to be delivered and tracked with a LMS.

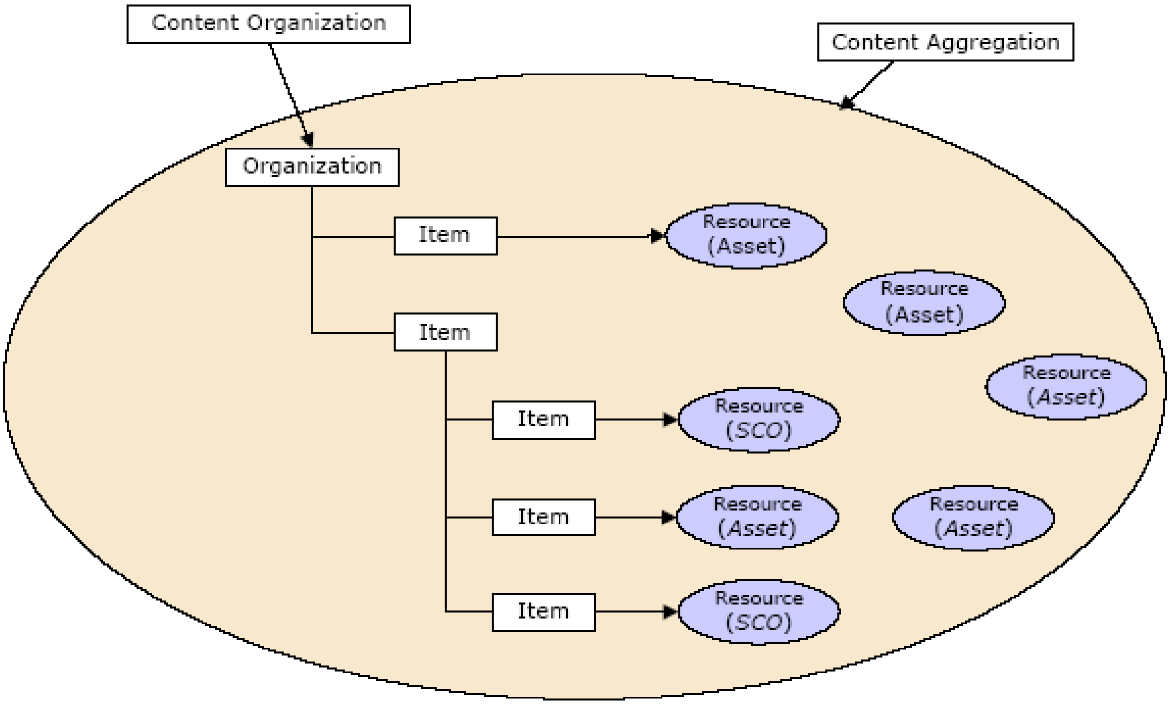

Figure 3.9 The SCORM component model, consisting of Assets, SCOs, and Content Aggregations (ADL 2004).

A *Content Aggregation* combines learning resources (SCOs and Assets) into a cohesive unit of instruction (e.g. course, chapter, lesson, etc.). A Content Aggregation is specified through a map known as the *content organization*. The Content Aggregation provides a mechanism to define the navigation structure between the SCOs and it indicates the sequence in which the learning resources are presented to the user.

A SCO may consist of several pages, but it will have to provide its own internal logic for the navigation between these pages. Thus, in a SCORM-based learning environment, the learner is confronted with two navigation systems: the SCO internal navigation and a navigation between SCOs provided by the LMS.

Although the SCORM Content Model provides a very detailed technical specification, in our opinion, the aggregation hierarchy is not clearly divided. On the one hand, HTML files are considered to be Assets while on the other hand, they provide means to embed other media files, such as images, multimedia objects, etc. From a pure conceptual perspective, these HTML files already aggregate other media components. From a technical perspective, each HTML file can be regarded as a basic data unit, which in our opinion, makes clear that the SCORM Content Model is a rather technical specification which is primarily aimed at the interoperability of learning contents.

We may conclude that the SCORM Content Model is a technical specification aimed mainly at the interoperability of learning contents between different LMS and other systems. It integrates standard web data formats (HTML, GIF, JPEG, JavaScript, Flash, etc.) into a standard model. The SCORM content model does not provide a separation of contents and presentation. The levels of granularity are not precisely defined.

Verbert and Duval 2004

Verbert and Duval (2004) provide a comparative analysis of different learning content component models. Therefore, they introduce a new general component model which serves as a basis to compare the other models.

This general component model defines three component levels: *Content Fragments*, *Content Objects*, and *Learning Objects* (see Figure 3.10).

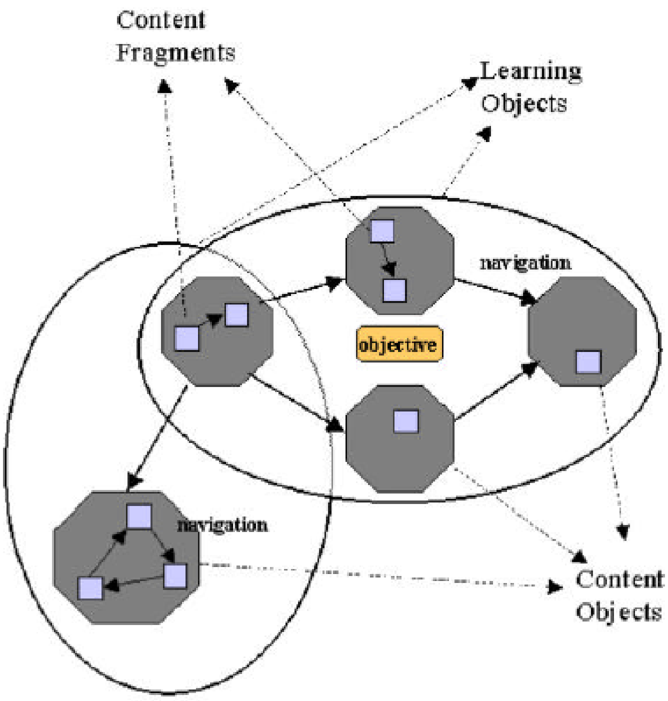

Figure 3.10 The general content model of Verbert and Duval consisting of Content Fragments, Content Objects and Learning Objects (Verbert et al. 2004).

- *Content Fragments* are learning content elements in their most basic form, e.g. text, audio and video representing individual resources.
- *Content Objects* aggregate Content Fragments and add navigation. They are abstract types, whereas Content Fragments are instances. A Content Object contains Content Fragments, zero or more Content Objects and adds navigation.
- *Learning Objects* aggregate Content Objects and other Learning Objects and add a learning objective. They define a topology between the components contained and can communicate with the outside world. A

Learning Object contains Content Objects, zero or more other Learning Objects and a learning objective.

Content Fragments can be extended with more elaborate didactic concepts such as activities and people; analogously, Content Objects can be extended with activity types and roles.

Verbert and Duval have analyzed four other component models, which are mapped to their component model: the Cisco RLO Strategy (see above), the SCORM Content Model (see above), the Learnativity Content Model (see above), and the NETg Learning Object Model (see above). The results are summarized in Table 3.2.

Table 3.2 The results of the mapping of four different learning content component models to the general content model according to Verbert and Duval (2004).

Model	Content Fragments	Content Objects	Learning Objects			
Learnativity	Raw media	Information Objects	Learning Object	Aggregate assemblies[1]	Collection[1]	
SCORM	Assets		SCO	Content Aggregation		
CISCO	Content Items	RIO	RLO			
Netg			Topic	Lesson	Unit	Course

[1] Although the authors refer to Wagner (2002) for the Learnativity Component Model, they use different terms which may be based on the paper of Duval and Hodgins (2003): they use “Aggregate Assemblies” instead of “Learning Component” and “Collection” instead of “Learning Environment”.

We may conclude that the general component model of Verbert and Duval represents a conceptual model which is mainly used to compare other learning content component models. Adding a navigation to the Content Objects suggests that these components are presented on several pages. Form our point of view, a level is missing which combines raw media elements to a single page.

Discussion

Seven different learning content component models have been presented: the NETg Learning Object Model, the Cisco RLO Strategy, the Learnativity Content Model, PaKMaS, the “Methodenlehre-Baukasten”, the SCORM Content Model, and the learning content component model of Verbert and Duval.

Two of the presented models contain two component levels (PaKMaS and Cisco). Interestingly, both models define a third bottom layer representing the elementary building blocks (PaKMaS’s Content Objects and Cisco’s Content Items). However, information on this conceptual level cannot be used as independent components because it is directly integrated into the next higher component level.

The SCORM Content Model and the model of Verbert and Duval define three levels, all of which can be reused independently. The NETg Learning Object Model consists of four levels. It specifies three higher levels for the aggregation of the "learning objects" but it provides only an abstract definition of the contents of "learning objects". Therefore, we do not discuss this model any further.

The Learnativity model has five levels, but from a pure learning content point of view, only four levels are relevant. The fifth level designates the whole learning environment, including the tools and services a LMS usually provides. The "Methodenlehre-Baukasten" does not explicitly define the different levels. An interpretation of the system description and the diagram presented in Figure 3.7 suggests that five levels are used: scenarios, modules (also called building blocks), lessons, pages, and components. The "scenario" level may also be related to the "Learning Environment" level, leaving four levels for pure learning contents.

If the conceptual basic levels of PaKMaS and Cisco are included, all models consistently contain media elements on the lowest level. PaKMaS provides a detailed technical model for this level by the LMML markup schema (see Section 3.5). Cisco describes the types of contents of this level conceptually, but no specification is given from a technical point of view. Further, both models do not contain an explicit specification as to the role of multimedia resources, which we expect to be individual "files". The other models seem to agree that this level consists of individual, reusable resources as electronic representations of media. However, the SCORM Content Model does not provide a clear aggregation hierarchy. HTML resources, as they are Assets, may integrate other media elements, which are classified as Assets too.

On the next level, however, the models start to differ. Some models suggest that this level should teach a single learning objective (Cisco's RIO, SCORM's SCO), while other models relate learning objectives to the third level (Learnativity, Verbert and Duval, and possibly the "Methodenlehre-Baukasten"). The latter models define a component level which aggregates media elements but which is not supposed to teach a "learning objective" on its own. For PaKMaS, it is not clear how its components are related to learning objectives.

We find a further difference which relates to its presentation on the second level. We may interpret that some second level components are presented as a single page in a web-based environment, such as PaKMaS's basic module, Cisco's RIO, and the page of the "Methodenlehre-Baukasten". However, SCORM's SCOs may be composed of several HTML pages. As Verbert and Duval's Content Objects add navigation, it may be interpreted that these Content Objects are presented on several pages. With respect to Learnivity's Information Objects, no information could be found as to how these objects relate to pages.

Table 3.3 A comparison of the granularity levels of different learning content component models.

Model	Component level				
	Content Asset	Information Object	Learning Object	Learning Component	Learning Environment
PaKMaS	Content Object[1]	Basic Module[2]		Structuring Module	-
Cisco	Content, Practice, Item[1]	RIO[3]		RLO	-
SCORM	Asset[4]		SCO	Content Aggregation	-
Learnativity	Content Asset	Information Object	Learning Object	Learning Component	Learning Environment
Methodenlehre-Baukasten	Component	Page	Lesson[5] / Module		Scenario
Verbert and Duval	Content Fragment	Content Object	Learning object[6]		

[1] These are only conceptual components which cannot be used independently from the next higher component level.

[2] It is not clear how the Basic Modules are related to a learning objective.

[3] A RIO is designed to teach a single learning objective but it is also an aggregation of “raw media” elements.

[4] SCORM considers HTML resources to be Assets, but from a conceptual point of view, they may combine the HTML’s textual contents with external media elements, thereby providing an aggregation of media elements.

[5] A Lesson is a complete sequence of didactic steps. Therefore it is considered to be the lowest level which teaches a learning objective. From a granularity point of view, a Lesson might be related to Cisco’s RLO.

[6] Verbert and Duval’s Learning Objects can be nested to provide topics, lessons, courses, etc.

Table 3.3 represents an attempt to compare the different learning content component models discussed (without the NETg model – for a comparison of this model see Table 3.2). The component levels of the Learnativity Content Model are chosen as a reference because this model provides the most detailed division. The following criteria for the classification were used: On the *Content Asset* level, the components were described as “raw media”. The *Information Object* level consists of an aggregation of “raw media” objects. Components on the *Learning Object* level are used to teach a single learning objective. *Learning Components* are a combination of Learning Objects, such as “lessons” or “courses”. We may point out that compared to the overview of learning content component models presented by Verbert and Duval, we relate Cisco's RIO and RLO to higher levels: the *Learning Object* and *Learning Components*, respectively.

We may conclude that a learning component model should provide at least three layers for learning contents: a media layer, a layer which integrates the raw media into reusable basic objects, and a layer which aggregates these basic objects into larger units. With the aim for a consistent technical implementation, there is a need to specify how the individual electronic representations of media are related to the model. We suggest that these resources are represented by the media layer. In order to clearly separate the navigation from the contents, a level which relates to pages in a web-based environment needs to be specified. In our opinion, relating pages to the second component level seems to be appropriate. Depending on whether or not a specific level should be related to a single learning objective, one or two further levels are needed. While having only one additional level may be more flexible, using two additional levels will clearly identify a level relating to single objectives and another level for larger lessons or courses.

3.5 Structured Markup for Learning Content

In order to aggregate learning contents coherently, the learning content component models provide a means to define a similar level of granularity of the basic building blocks. In addition, a standard data format for the actual information contained in these components is needed so as to ensure that contents originating from different sources can be consistently presented to students. Structured markup provides the means to separate contents from the visual presentation and allows for the adaptation of learning contents to different presentation contexts, such as didactic scenarios, presentation media (web-based, paper, etc.), or corporate designs. In fact, structured markup allows for the specification of an information model for the learning contents and is suited to specify textual structures, e.g. paragraphs, headers, tables, lists, superscripts, etc. of components containing text. It also provides the "glue" for the integration of resources using specialized multimedia data formats.

The *Extensible Markup Language (XML)* plays an increased role as a cross-platform compatible data format and enables the separation of contents and presentation (W3C 2004a). XML provides a generalized syntax which can be extended for specific data structures through a *Document Type Definition (DTD)* or an *XML Schema* (W3C 2004b). XML has been designed to exchange data in a standardized format between otherwise incompatible systems and applications.

One advantage of XML is the availability of programming libraries that offer standard tools for parsing and transformation. Together with a DTD or an XML Schema, any XML-based data format can be processed and validated using standard parsers. The *Extensible Stylesheet Language (XSL)* enables transformations into various output formats (W3C 1999d).

Below, an overview of existing XML-based languages for learning contents is presented. We restrict this overview to languages which may be used for learning contents. Therefore, languages that define learning scenarios without the possibility of marking up pure learning contents are not considered in this overview.

HTML / XHTML

The *HyperText Markup Language (HTML)* is the general publishing language of the World Wide Web (W3C 1999a). The fact that HTML documents can be displayed across different browsers and platforms is certainly an important factor for the explosive growth of the web. Therefore, HTML is a natural choice for web-based learning contents as well.

HTML provides a useful set of markup elements, such as headings, phrase elements, quotations, paragraphs, sub- and superscripts, lists, tables, and hyperlinks. It provides a means to embed images and other media elements.

Unfortunately, HTML allows content creators to mix structured markup with graphical styling. The web-publishers' need to create fancy graphical designs has resulted in a great number of techniques for the "misuse" of the basic structuring elements in favor of sophisticated styling and layout. Hence, HTML does not truly separate content and presentation.

HTML is based on SGML, a predecessor of XML. XML was conceived as a means of regaining the power and flexibility of SGML without most of its complexity. The *Extensible HyperText Markup Language (XHTML)* was developed in order to provide an XML-compatible "version" of HTML, which benefits by being XML conformant (W3C 2002). The elements and the attributes of XHTML correspond to the ones defined in the HTML specification. XHTML documents can be readily viewed, edited, and validated with standard XML tools. Using XML namespaces (W3C 1999c), other XML languages can be embedded into XHTML to extend its functionality.

DocBook

DocBook is an established markup language that was originally designed for the documentation of soft- and hardware (Walsh et al. 1999). Given the great number of tools available, such as editors, transformations, etc., DocBook is also used in many other areas. The DTD provides a very rich set of markup to represent the detailed structures of books (table-of-contents, chapters, cross-references, indexes, glossaries, bibliographies, etc.). The DocBook elements can be divided broadly into the following categories:

- A `Set` contains two or more `Book` elements.
- A `Book` consists of a mixture of the following elements: `Dedication`, navigational components (`Toc, Index`), divisions, and components.

- *Divisions* divide books into `Parts` and `References`. `Parts`, in turn, contain components.
- *Components* divide books or divisions into chapter-like elements: `Preface`, `Chapter`, `Appendix`, `Glossary`, `Bibliography`, and `Article`.
- Several flavors of *sections* subdivide components: `Sect1`…`Sect5`, `Section`, `SimpleSect`, `RefSect1` … `RefSect3`, etc.
- *Meta-information* elements contain bibliographic information, such as `Author`, `Title`, `Publisher`, and so on.
- *Block elements* are usually presented with a paragraph break before and after them. DocBook provides lists (`ItemizedList`, `OrderedList`, `SimpleList`, `VariableList` …), admonitions (`Caution`, `Important`, `Note`, `Tip`, and `Warning`), line specific environments which preserve whitespace and line breaks (`ProgramListing`, `ScreenShot` …), examples (`Example`, `InformalExample`), figures, tables, paragraphs (`Para`, `SimPara`, `FormalPara`), equations (`Equation`, `InformalEquation`), graphics (`Graphic`, `MediaObject`), and others (`BlockQuote`, `Epigraph`, `Procedure`, `Sidebar` …).
- *Inline elements* are generally represented without any obvious breaks. They can be classified into traditional publishing elements (`Abbrev`, `Emphasis`, `Footnote`, `Phrase` …), cross-references (`Anchor`, `Citation`, `Link`, `XRef` …), markup (`ForeignPhrase`, `ComputerOutput`, `Literal`, `UserInput`, …), mathematics (`InlineEquation`, `Subscript`, `Superscript`), user interfaces (`GUIButton`, `GUIMenu`, `Shorcut` …), programming languages (`ClassName`, `Constant`, `Function`, `Interface`, `ErrorCode` …), operating systems (`Application`, `Command`, `Filename` …), general purpose (`Email`, `InlineGraphic` …).

From a learning perspective, important elements are missing, such as exercises, interactive multimedia elements, etc. This fact and the complex set of markup elements, which is difficult to learn, suggest that DocBook is not well-suited as a markup language for learning contents in an academic environment.

However, there have been approaches to using DocBook as a basis for learning content. Wiest and Zell (2001) present a system for web-based training which uses an XML compliant subset of the DocBook DTD that is enriched with interactive hypermedia and educational elements.

Learning Material Markup Language (LMML)

LMML was developed at the University of Passau and has been implemented in the PaKMaS project (Süß 2000). LMML is designed to be more than just a single XML language for learning contents. Rather, it is considered to be a framework which is adaptable to different knowledge domains. Specialized languages have been developed for computer sciences (LMML-CP), operations research (LMML-OP), financial planning (LMML-FP), and generative programming (LMML-GP). It is extensible by elements of other XML languages (e.g. MathML). LMML provides a document-oriented fragmentation of learning contents into modules of arbitrary granularity.

The LMML framework and its languages are based on the Passau Teachware Model, describing the general modular structure for e-learning contents. LMML organizes content into modules which may contain further modules. The general parts of LMML are (see Figure 3.11):

1. *ContentModules:* These are small, thematically self-contained entities, e.g. *definitions*, *examples*, etc. They are composed of *MediaObjects* and can be structured by *StructureObjects*.
2. *MediaObjects:* Text, image or sound elements contain or refer to the actual multimedia contents. Text may contain further inline markup such as *emphasized*, *quoted*, *annotated*, *referencesLink*, *externalLink*, etc.
3. *StructureObjects:* These are lists and tables which may be nested recursively.
4. *StructureModules:* ContentModules can be combined into larger entities, such as sections or collections.

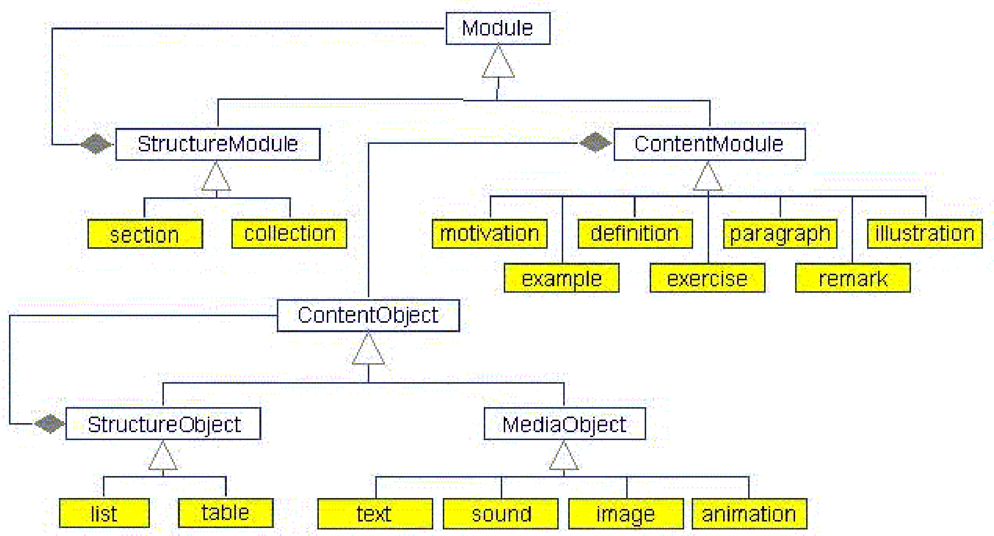

Figure 3.11 The Passau Teachware Model (http://www.lmml.de/).

Educational Modeling Language (EML)

The main purpose of the *Educational Modeling Language (EML)* is a standardized description of educational scenarios (Koper 2001). The pedagogical meta-model aims at being flexible enough for different didactic methods, such as programmed instruction, competency based learning, learning by doing, problem based learning, etc.

It is based on four “packages”: a *learning model* that describes how learners learn, a *unit of study model* that describes learning objectives, prerequisites, roles of staff and learners, learning activities, and the learning environment, a *domain model* that takes into account the characteristics of the content domain (e.g. mathematics, economics, psychology, etc.), and *theories of learning and instruction*, such as empiricist (behaviorist), rationalist (cognitivist and constructivist), and pragmatist-sociohistoric (situationalist).

EML defines a *unit-of-study,* which is made up of the following parts (see Figure 3.12):

1. *Metadata:* Information on the unit of study, mainly for retrieval purposes.
2. *Roles:* The roles of the actors in the unit of study.
3. *Learning-objectives:* The learning objectives of the unit of study (optional).
4. *Prerequisites:* The prerequisites for learners to start the unit of study (optional).
5. *Content:* The content contains *Environments* and *Activities*. An *Environment* consists of so-called *Knowledge-objects*, *Announcement-objects*, *Communication-objects*, *Tool-objects*, *Questionnaire-objects*, etc. An *Activity* is an indivisible task.
6. *Method:* The sequence of how the activities are presented and performed.

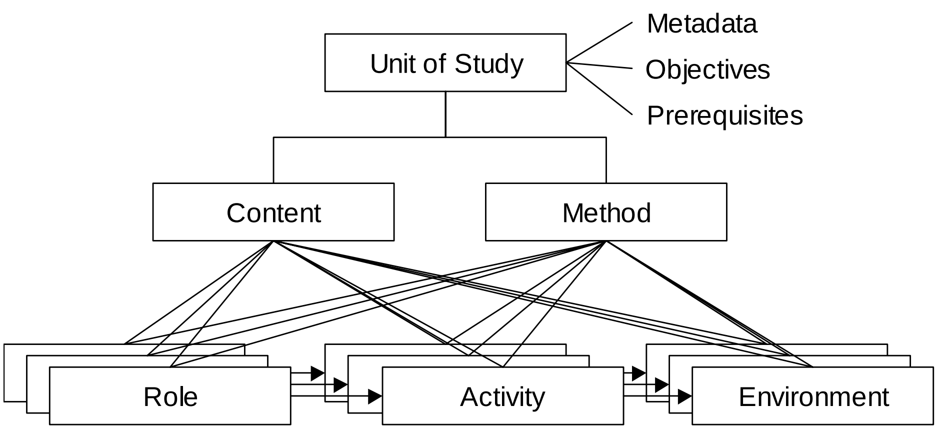

Figure 3.12 The EML structure (According to Koper 2000).

Although EML has been designed mainly as a modeling language for educational scenarios, it provides markup to specify learning content in the *Environment* as *Knowledge-objects* or *Questionnaire-objects*. Knowledge-objects may be study books, fact sheets, manuals, dictionaries, encyclopedia, etc. The source of Knowledge-Objects may be outside EML being referenced through a URI. Still, knowledge objects may also be encoded using the structured markup for Knowledge-Objects that EML provides.

Meanwhile, the development of EML has been replaced by the *IMS Learning Design* (IMS 2003) specification which provides a generic and flexible language to support a wide range of pedagogies in online learning. EML served as the basis for this specification. Nonetheless, other than EML, IMS Learning Design does not contain structured markup for learning contents.

Because EML is outdated by the IMS Learning Design specification, EML's markup for actual learning contents may be not supported any longer.

IMS Question and Test Interoperability Specification (QTI)

The *IMS Question and Test Interoperability Specification (QTI)* describes a basic structure for the representation of question and test data (IMS 2002a). It supports the exchange of questions and tests between Learning Managements Systems, content authors, and content libraries. It is an XML language that has been specially created for online-learning.

QTI provides markup for tests of the following types:

- *Logical identifiers:* Standard true/false questions, multiple-choice tests (text, images, audio), ordering tasks (text, image), connect-the-points.
- *X-Y Co-ordinates:* Image hot spots, connect-the-points.
- *Strings:* Fill-in-blank, standard short answer.
- *Numerical:* Fill-in-blank, numerical entry with slider.
- *Logical group:* Drag-and-drop (images).

In order to specify a simpler schema, QTI Lite contains a subset of elements of the full QTI specification by providing only true/false and multiple-choice questions. QTI does not clearly separate contents (i.e. the questions and answers) from presentation (the layout to present the questions and answers). Therefore, the specification is not well-suited for a media-independent representation of questionnaires, quizzes, etc.

Other Structured Markup Schemas for Learning Content

Other structured markup languages for learning contents, which are not be presented here in detail, include the *Ariadne Course Description Format (CDF)*, *PALO,* the *Targeteam TeachML*, and the *Tutorial Markup Language (TML)*. The CEN/ISSS "Survey of Educational Modeling Languages (EMLs)" provides an in-depth overview and comparison of these languages together with the *EML* and the *LMML* formats (Rawlings et al. 2002). Aimed at *"a semantic rich information model and binding, describing the content and process within 'units of learning' from a pedagogical perspective"*, the survey concludes that only *EML* and *PALO* can express pedagogical models. The other languages restrict themselves to modeling learning content and structure. Further:

> *"Each of the approaches share a strong interest in supporting reusability. However, looking into more detail … it is clear that the element/attributes that are in use in each of the EML are proprietary. Interoperability, at least at this stage, between the EMLs cannot be achieved ... In this context it is important to note that currently EML-OU (version 1.1) is the only one of the EMLs discussed which both is compatible with various international standards and follows the process and procedures to be accepted as a standard (Rawlings et al. 2002)."*

We might note that three of the five authors were employed at the Open University of the Netherlands (OUNL), where EML was developed, and one author came from the Universidad Nacional de Educación a Distancia (UNED), the origin of PALO.

However, in our opinion there is a need for a standard structured markup language for pure learning contents – be it a separate language for learning contents only or be it combined with a model which describes pedagogical processes.

In addition to the aforementioned "learning modeling" languages, other XML-based formats, which are not primarily designed for education, may be useful for the specification of learning contents.

The *Mathematical Markup Language (MathML)* is a low-level specification for describing mathematics for inclusion in web pages (W3C 1999b). As of now, there are many tools for MathML available but contemporary browsers still do not render mathematics correctly in their default configuration.

Another interesting specification of the World Wide Web Consortium (W3C) is the *Scalable Vector Graphics (SVG)* language for describing two-dimensional graphics in XML (W3C 2003). Compared to bitmap images, which are generally used for two-dimensional illustrations on the web, SVG graphics can be scaled without loss of quality. SVG can be combined further with scripting languages such as ECMAScript (also known as JavaScript) to provide dynamic and interactive illustrations. Again, there is no default support of SVG with today's browsers.

Discussion

Looking at the structured markup languages for learning content presented above, different types of languages can be identified:

- *Pedagogical modeling languages:* Languages like EML, IMS Learning Design, or PALO (see above) are used to model different types of educational scenarios, such as programmed instruction, problem based learning, etc. They provide a means to define roles, learning activities, learning environments, etc. They do not necessarily provide markup for learning content (e.g. IMS Learning Design).

- *Structured markup for learning content:* These languages provide a means to markup learning content and structure. Examples include HTML/XHTML, DocBook, LMML, etc.
- *Question and test markup languages:* Here the most important format is the IMS Question and Test Interoperability Specification (QTI). It provides a means to define multiple-choice-type tests that can be used for assessments.
- *Specialized markup languages:* These languages are used for formal notation systems which cannot be expressed by simple text, e.g. MathML for mathematics and visualizations and SVG for 2-dimensional graphics. They do not represent a complete learning content markup language but they may be combined with such languages to include specialized contents.

We may conclude that no markup schema for general learning content that could serve as a future standard data format for learning contents has yet been established. HTML/XHTML has been often used for learning material but it cannot be considered as supporting the separation of contents and presentation. Furthermore, since it is not specifically designed for learning, no specific markup is provided for educational purposes. DocBook is an established standard for hard- and software documentation. It contains a very rich set of markup elements to represent the traditional structure of books. However, it is not well-suited for multimedia learning contents. EML has been outdated by IMS Learning Design. IMS Learning Design does not provide specific markup for learning contents.

The large number of different approaches to specify a general XML language as well as the fact that no standard has yet been established illustrates the difficulty in defining a general and interoperable structured markup schema for learning content which fulfills the various needs of different knowledge domains and teaching methods.

LMML tries to overcome these problems by being solely a general framework to define specialized markup languages. LMML-based languages can then be adapted to different knowledge domains. Still, one may question whether or not such a customization will lead to a standard language which supports the exchange of interoperable and reusable learning contents.

Nonetheless, in taking a closer look at LMML, different levels can be identified. The ContentModules level defines elements, such as example, definition, exercise, illustration, etc. These elements contain the lower level MediaObjects, such as text, images, animations and StructureObjects, which provide a means to define lists and tables.

This leads to the idea that for the basic reusable components, which should consist of only a single didactic content type, e.g. an example, a definition, an exercise, an illustration, etc., a simple structured markup schema based on standard typographical elements may be sufficient. The didactic content types may be assigned to the component as a whole using metadata. Thus, such a structured markup schema may contain block elements (headings, paragraphs, annotations, lists, tables, images, multimedia elements) and inline elements (strong, emphasis, underline, superscript, subscript, links). Because of its simplicity, it is easy for content authors to understand and it is likely to be easily convertible to possible future data formats. The markup schema remains stable, even when new didactic content types are needed – new types can be simply added to the list of didactic content type metadata categories.

While a traditional set of typographical markup might be very flexible, it seems to be clear that special didactic functionality is needed, e.g. for assessments. As specialized didactic markup may reduce the flexibility of the basic structured markup schema, one approach may be to provide a separate markup schema for didactic issues, such as the IMS QTI for multiple-choice type questions.

3.6 Content Chunking

This section focuses on content chunking methods, which should support authors to divide the learning contents into modular learning objects. A large number of methodologies for the general development of e-learning applications can be found in literature. A good overview is presented by Braxton et al. (1995). Two methods which explicitly refer to the content chunking procedure are examined in more detail: the *ISDMELO* methodology and the *SCORM Best Practices Guide for Content Developers*.

ISDMELO

Baruque and Melo (2003) present the ISDMELO methodology to develop e-learning instruction using the learning object paradigm. The methodology is based on Instructional Systems Design (ISD). It claims to be grounded on sound pedagogical principles and should allow for the combination of principles from the major learning schools (behaviorist, cognitivist and constructivist).

The ISDMELO methodology is based on the general phases of ISD: *analysis*, *design*, *development*, *implementation*, and *evaluation*:

I. *Analysis:* This phase contains the *specification of the learner profile*, a *problem analysis*, a *search for existing learning objects*, and an *environmental analysis*. All data gathered in this phase should be kept as *metadata*.

II. *Design:* This phase can be divided into the design of the instructional contents and the design of a learning object interface which provide a consistent "look-and-feel". As the content design is an important aspect of the content chunking procedure, it is described in more detail below. The steps for the interface design are: *user modeling for the interface design*, *user task analysis* (focusing on how the user interacts with the learning objects), *finding a metaphor*, design of the *interface look*, design of the *interface feel*, an *evaluation of the interface prototype*, and the creation of interface specific *metadata*.

III. *Development:* The steps to produce learning objects and storing them in a repository are: *search for possible existing components*, *building the learning objects*, *quality control*, *storing the learning objects in the database*, and the generation of *technical metadata*.

IV. *Implementation:* In order to deliver the instructions to the learners, the following procedures are needed: *Integration of the learning objects into a product* (using wrappers, frames, links, or templates to be delivered via an LMS or a web site), *choosing the most adequate delivery mode* (self-paced, collaborative, or instructor-led learning), a *management plan* for the most effective delivery of instruction, *running the product*, and *tracking the learners' progress*.

V. *Evaluation:* A *formative* and a *summative evaluation* should be conducted.

An important aspect of the ISDMELO methodology is the content design which can be considered an approach to identify the single learning objects. It contains the following steps:

1. *Task analysis:* Based on what the learner should be able to do, the major learning objective established in the analysis phase should be deconstructed into sub-objectives.
2. *Content analysis:* The content analysis asks what the learners should know and will reveal - the concepts, principles, or procedures to be taught.
3. *Learning objects structure:* After defining the hierarchical tree of learning objectives, the contents should be chunked into different learning object levels. It is recommended that each item contains three to seven sub-items. In this step, Baruque and Melo refer to the concepts of *epitome* and *elaboration* of Reigheluth's Elaboration Theory (Reigeluth et al. 1983).
4. *Sequence of instruction:* The sequence in which the learning objects will be delivered should be defined.
5. *Learning object categorization:* The learning objects should be categorized. Referring to the Cisco RLO Strategy (Barritt et al. 2000, see Section 3.4), Baruque and Melo propose to use the categories *principle*, *process*, *procedure*, *concept*, and *fact*.
6. *Learning object specification:* The resulting learning objects will then be specified by their learning outcomes, content to be covered, evaluation method, example, practice, media and instructional approach.
7. *Metadata:* All data created should be kept to generate metadata records.

SCORM Best Practices Guide for Content Developers

In its "SCORM Best Practices Guide for Content Developers" (LSAL 2003), the Carnegie Mellon's Learning System Architecture Lab (LSAL) provides a detailed description of how to create learning contents for SCORM (the SCORM Content Model is described in Section 3.3). LSAL leaves open which level of granularity SCORM's Sharable Content Object (SCO) should have. The "roles" a SCO can play are: learning objectives in a lesson, segments in a lesson, lessons in a module, modules in a course, etc.

An overview of the suggested procedure is illustrated in Figure 3.13. The process identifies two roles, the instructional designer and the programmer.

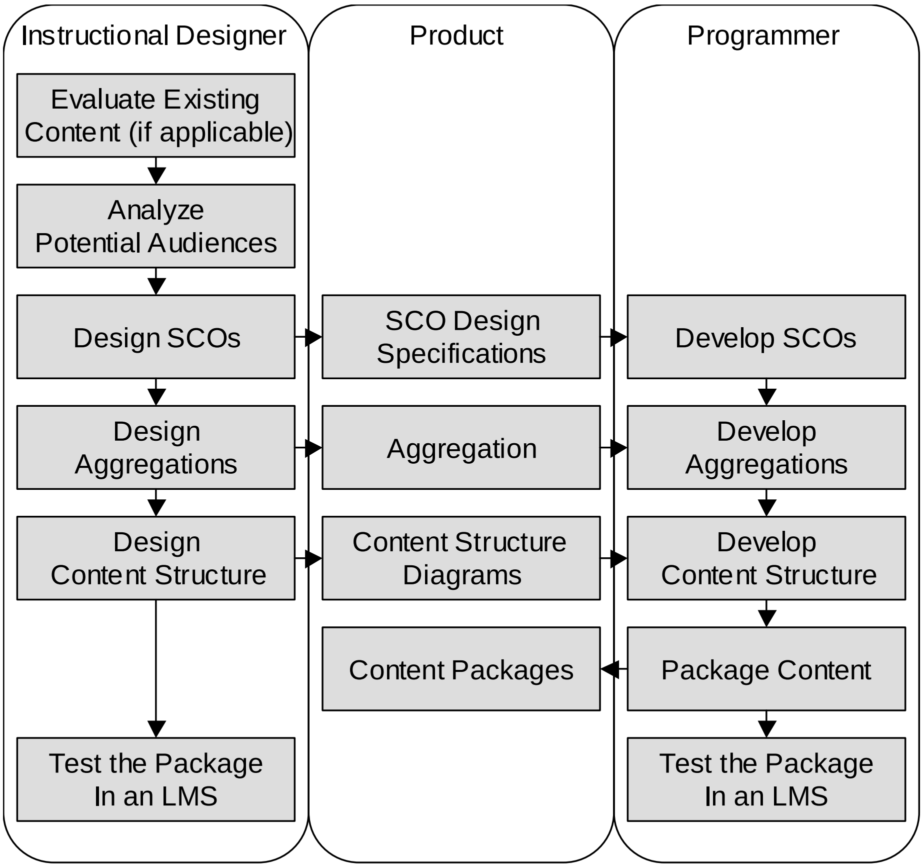

Figure 3.13 Overview of the design, development process and products for SCORM-based instructional materials (According to LSAL 2003).

The instructional designer analyzes the contents and the potential audiences and defines the learning objectives. Based on this information the contents are broken down into *SCO*s, representing instructional objectives. The SCOs are described in the *SCO Design Specifications*. In the next step, the *Aggregation* is specified, by which related SCOs are grouped into a tree structure to be delivered to the learner in the manner prescribed. The *Content Structure Diagram* then specifies the dynamic sequencing of the SCOs. Sequencing defines the order and rules in which the learner will proceed depending on the results of the learning activities, for example assessment results.

The programmer develops the *SCOs*, *Aggregations*, and the *Content Structure* based on the specifications delivered by the instructional designer. The final steps are then to put everything together, i.e. *packaging the contents*, and to *test the package in an LMS*.

Upon closer examination of the process of specifying SCOs, the LSAL guide suggests starting with an instructional strategy or with existing material. A SCO should represent an instructional objective and all the related materials and resources to support that objective. The objectives may be terminal or enabling objectives and they may be performance- or knowledge-based.

As one of the goals of SCORM is to create reusable content, an analysis of the potential audiences is conducted. LSAL suggests performing a short brainstorming session aimed at identifying three to five types of learner groups which may be interested in the subject matter to be treated. Subsequently, the content should be “divided” into individual SCOs to make it optimally reusable for the different groups. This results in a number of SCOs that are of interest to numerous individuals and organizations, while other SCOs may target only a restricted set of learner groups. A SCO may also be designed for a single target group, for example, providing a specific introduction or overview, knowing that it may not be as reusable others. In order to create SCOs in a context-neutral way, LSAL suggests designing individual SCOs for every learning objective.

Discussion

Two methods have been presented that should support content authors or instructional designers to break contents down into modular learning objects. In both methods the learning objects are based on a single objective.

The ISDMELO methodology provides a clear “top-down” procedure to identify the learning objects that are based on principles, processes, procedures, concepts, and facts. The final learning objects include information related to different didactic content types.

The LSAL SCORM Best Practice Guide is specifically related to the SCORM Content Aggregation Mode. The guide’s primary focus is for the (industrial) training community but the authors claim that the strategies presented can be easily applied to other educational communities, such as higher education. The guide does not provide a specific method to identify the learning objectives. The instructional designer must decide the most appropriate way to do this. The learning objectives may be enabling or terminal objectives and therefore the Shareable Content Objects (SCOs) may vary considerably in their granularity. We may point out that this method pays special attention to the reusability of SCOs for various learner audiences.

In both methods presented, the level of granularity proposed is based on learning objectives. While ISDMELO relates these objectives to principles, processes, procedures, concepts, and facts, the level of the learning objectives is explicitly left open by LSAL. Still, the latter contains a method for developing learning objects that are reusable for various learner audiences.

3.7 Conclusions

In order to aggregate modular learning objects coherently, they should have a similar level of granularity. In addition, a standard data format for the actual information contained in these objects is needed as this ensures a consistent presentation of contents originating from different sources.

Examination of the definitions of learning objects (Section 3.2) reveals that the size of learning objects varies considerably. However, there seems to be a common understanding that learning objects should be relatively small, self-contained units of learning content which can be aggregated into larger learning units and which are reusable in multiple instructional contexts.

In order to define a level of granularity that is more specific and is based on the structure of learning content, different classification schemas that have been found in didactic theories, literature on the writing of educational materials, and learning object standards for learning contents have been reviewed (Section 3.3). Two general levels can be found. The top level categorizes subject matter topics as concept, fact, principle, procedure, etc. The second level relates to the didactic purpose of contents, such as definitions, examples, exercises, etc. These categories may be related to one of Gagné's nine instructional events. We refer to these categories as *didactic content types*.

Beyond the learning object as the basic building block, a specification of how these building blocks can be aggregated into larger learning units, and possibly how they integrate basic media elements, is needed. This is defined by learning content component models which contain different levels of content components. They then specify how the components on lower levels can be combined to higher level units.

Based on the review of seven learning component models (Section 3.4), we may conclude that a learning component model should provide at least three layers for learning contents: a media layer, a layer which integrates the raw media to reusable basic objects, and a layer which aggregates these basic objects into larger units. In our opinion, such a model should specify how the individual electronic representations of media are related to the model, which is preferably represented by the media layer. In order to clearly separate the navigation from the contents, a level that relates to pages in a web-based environment needs to be specified. Additional levels may be needed, depending on whether a specific level should be related to single learning objectives.

We argue that the basic building blocks should be based on *didactic content types*, rather than on learning objectives, for two reasons:

- Didactic content types enable flexible adaptation of learning units to the needs of students from different scientific disciplines.
- While learning objectives vary in the breadth of coverage of subject matter, didactic content types are related to single instructional events; we anticipate promoting the development of modular contents with a similar level of granularity.

In order to provide a standard data format for the actual information, we propose the use of structured markup which separates contents and the visual presentation. Such contents can be easily integrated into varying presentation contexts using different presentation media (web-based, paper, etc.) or corporate designs. Several XML-based markup languages have been reviewed (Section 3.5). No established markup for learning content could be found which might be a candidate for future standardization. This might be due to the difficulty of defining a schema which fulfills the various needs of the different knowledge domains and teaching methods.

Based on the proposed size of the basic reusable components, which represent only a single didactic content type, we suggest that a simple structured markup schema based on standard typographical elements may be sufficient. In order to identify different didactic content types, this information may be assigned to the component as a whole using metadata. Because of its simplicity, such a schema is easy to understand by content authors and is likely to be easily convertible to possible future data formats. The markup schema remains stable, even if new didactic content types are needed – new types can be simply added to the list of didactic content type metadata categories.

It seems to be clear that special didactic functionality is needed, e.g. questions and tests for self-assessments. Specialized didactic markup might reduce the flexibility of the basic structured markup schema and therefore, we propose providing a separate markup schema for such didactic issues.

Finally, two methods have been reviewed which support content authors to divide learning content into small, modular learning objects (Section 3.6). Both methods suggest the size of learning objects is based on learning objectives. However, as we are aiming at a granularity level based on didactic content types, the methods need to be adapted accordingly. Inspired by the LSAL guidelines, we think that a new content chunking method should include a method which analyzes potential audiences and optimizes the reusability of the learning objects for these learner groups.

4 dLCMS Project

The dynamic Learning Content Management System (dLCMS) project aims to provide a tool to implement a modularization strategy combined with structured markup to enhance the reusability of learning contents. An information model for learning contents is specified which consists of a learning content component model, structured markup, and metadata. The component model defines three levels of components: Assets, Content Elements and Learning Units. Content Elements, the modular basic building blocks, are to be self-contained and a standard level of granularity is proposed. The Content Element structure is specified by a structured markup schema which integrates the learning media and provides a means to separate contents and presentation. The dLCMS functional architecture that handles and processes the modular learning contents is delineated. It consists of four main components: authoring, repository, assembly and linking, and publishing and export. The implementation of the dLCMS prototype is realized as an extension of the existing Silva open source content management system. As a whole, it provides a platform to investigate the applicability of modularization and structured markup from the point of view of learning content authors.

4.1 Objectives

The *dynamic Learning Content Management System (dLCMS)* project aims at implementing a *modularization* strategy combined with the *separation of content and presentation* to enhance the reusability of learning contents.

In particular, the design goals include:

- *Modularization:* The system should support the use of small, modular learning objects which can be flexibly combined to be reused in different educational contexts. Therefore, a *learning content component model* defines different levels of components, their properties, and the relationships of the different component levels to each other.
- *Separation of contents and presentation:* A *structured markup schema* is provided which serves as a flexible data format for the learning contents contained in the basic component layer. It should support the separation of content, presentation, and navigation in order for contents to be reused in different presentation contexts, such as didactic scenarios, presentation media (web-based, paper, etc.), or corporate designs.
- *Editing:* As many authors in a scientific environment are not computer experts, they should be able to create contents using structured markup focused on their subject matter without having to care about XML syntax

and programming languages. This implies that an easy-to-use editor needs to be provided.

- *Flexible aggregation of learning content:* Content shall be easily assembled into learning units for various instructional contexts, i.e. learning units which are well adapted to different learner target groups and which use appropriate didactic methods.
- *Centralized content management:* Centralized content management allows authors and teachers to collaboratively (re-)use learning resources.
- *Flexible graphical design:* The system should support flexible graphical styling and layout. Style and layout templates make it easy to adapt the look and feel of learning units to different corporate designs and to keep the designs up-to-date. This helps to maintain an attractive presentation of learning contents for the future.
- *Export of learning units:* Learning units which are created with the system should be interoperable with other LMSes. Therefore, learning units need to be exportable in standardized and interoperable packaging formats.

We have implemented the dLCMS prototype as an extension of the existing open source content management system *Silva* (Infrae 2005) which is described in Section 4.3.1 in more detail. The design of the dLCMS consists of an *information model* and a *functional architecture*. The information model, presented in Section 4.2, specifies the data structure of the modular learning contents. It includes the learning content component model, the structured markup, and metadata. The functional architecture, described in Section 4.3, consists of the functional components needed to handle and process this data. These components offer functionalities for editing, storage of contents, assembly of components, publishing, etc.

4.2 dLCMS Information Model

The *dLCMS Information Model* provides a standardized specification of the data structure for modular web-based learning contents to be used by the dLCMS. It aims at enhancing the reusability of web-based learning contents through a modularization strategy and an approach to separate contents from their graphical presentation and navigation. It consists of three parts.

- *dLCMS Component Model:* As a modularization strategy, it defines different levels of learning content components. The different levels specify the basic reusable building blocks, the way in which they are composed from media representations, and the way in which they are aggregated into larger units. The dLCMS component model further defines the granularity level for the reusable building blocks. It is presented in Section 4.2.1.

- *Structured markup:* XML schemas serve as a data model for the basic reusable building blocks and provide an abstraction of the visual presentation which thereby enable the separation of content and presentation. This is described in Section 4.2.2.
- *Metadata:* A set of metadata items provides a standardized way to describe the components to be used for search and retrieval. This is presented in Section 4.2.3.

4.2.1 dLCMS Component Model

The dLCMS Component Model specifies different levels of learning contents, their relationships, i.e. the way components on different levels can be aggregated into each other. Furthermore, it defines the properties of the components on the different levels. Based on our conclusions from the literature review (see Chapter 3, especially Section 3.7), we define the following requirements for the dLCMS Component Model:

- The reusable basic building blocks should be based on didactic content types rather than on learning objectives.
- The component model should provide at least three layers for learning contents: a media layer, a layer which integrates the raw media into reusable basic objects, and a layer which aggregates these basic objects into larger units.
- The individual electronic representations of media should be directly represented by a component layer.
- In order to clearly separate the navigation from the contents, a level which relates to pages in a web-based environment needs to be specified.

We also aim to achieve a clear aggregation hierarchy that is easy to understand. Therefore, components should only be aggregated into higher levels. The dLCMS Component Model should be flexible enough to suit the various needs in an academic environment. It is mainly intended for the modeling of modular learning contents which may be (re-)used in different didactic scenarios. In the future, these learning contents may be combined with educational modeling specifications, such as IMS Learning Design (IMS 2003).

As a result, we have defined the dLCMS Content Model to consist of three component types: *Assets*, *Content Elements*, and *Learning Units* (see Figure 4.1). Basically, Assets relate to the individual media resources, Content Elements represent the modular basic building blocks, and Learning Units are aggregations of Content Elements to larger learning units, such as lessons or courses. These components are described in more detail below. Because Content Elements represent the basic building blocks of learning contents and thus incorporate the main concepts of the dLCMS Component Model, they are described first.

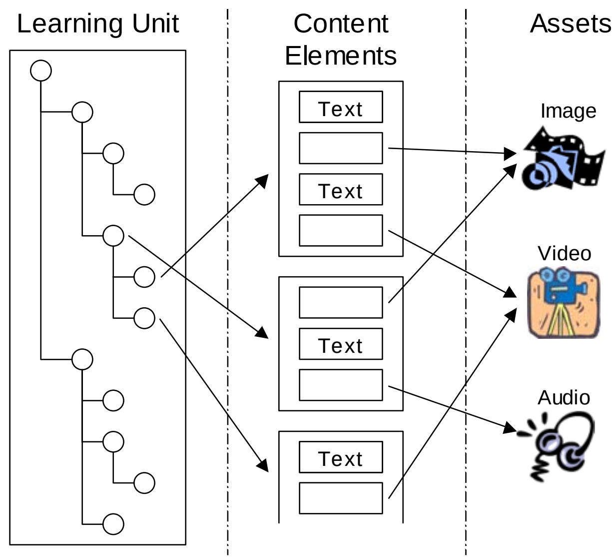

Figure 4.1 The dLCMS Component Model defines three component types: Learning Units, Content Elements, and Assets. (Text fragments are not considered to be Assets, see Section Assets below.)

Content Elements

Content Elements represent the basic building blocks that can be flexibly combined to be reused in different educational contexts. In order to coherently aggregate Content Elements into Learning Units, they should be self-contained and have a similar level of granularity.

The notion "Content Element" does not contain the word "learning". Learning is more than just the perception of pieces of information. The basic building blocks of information need to be assembled into an instructional context before learning can take place. Furthermore, we prefer the term "content" to "information". The word "content", in our eyes, relates to subject matter being contained (cp. the notion "table of contents") in a much better way than "information", which covers a broad range of meanings, such as a message, a represented pattern, sensory input, etc.

In order to be freely combined to Learning Units, a Content Element should not directly refer to other contents through a hypertext link or an explicit linguistic reference in the text. Clearly, Content Elements based on didactic content types may not be considered as providing a complete learning experience on their own. They need to be presented together with other Content Elements and might depend on prerequisites to the learners' knowledge, both of which contribute to a learning context. Therefore, we consider a Content Element to be *self-contained*, if does not have any explicit references to other contents. Still, a Content Element may have prerequisites to the learners' knowledge.

As a standard level of granularity, Content Elements should be based on *didactic content types* (see also Sections 2.2, 3.3, and 3.7) which are used to teach a single objective, such as a fact, a concept, a principle, a procedure, or a process. A didactic content type may be seen as a piece of learning content which relates to one of

Gagné's Nine Instructional Events (see Section 3.3). We argue that only Content Elements based on didactic content types are flexible enough to be reused for teaching students from different scientific disciplines. Although a level of granularity based on learning objectives may yield a complete, didactically sound learning experience, such contents cannot be easily adapted to the differing needs of specific learner groups and imposes the use of the didactic design provided. Using didactic content types, the contents can be flexibly adapted to the students' needs and the teachers' preferred didactic approach.

We propose an initial set of didactic content type categories which is presented in Table 4.1. The table shows how the didactic content types may be related to Gagné's Nine Instructional Events. The categories were based on the classification for written educational texts of Ballstaedt (see Section 3.3), the items of the LOM Learning Resource Type vocabulary which could be related to a single instructional event (see Section 3.3), and some ContentModules types defined in the LMML markup language we found to be useful (see Section 3.5). Additionally, we have included "literature" as didactic content type category.

Table 4.1 Classification of didactic content types and their possible relations to Gagné's Nine Instructional Events.

Instructional Event	Related Didactic Content Type
Gaining attention	Example[1,3], problem statement[2]
Informing learners of the objective	Learning objective[1]
Stimulating recall of prior learning	Advance organizer[1,4]
Presenting the stimulus	Expository[1], definition[3,5], narrative[1,2], instruction[1]
Providing learning guidance	Example[1,3], excursion[1], glossary[1], literature, experiment[2]
Eliciting performance	Exercise[1,2,3], self-assessment[2], simulation[2]
Providing feedback	(Feedback of self-assessments and simulations)
Assessing performance	Questionnaire[2]
Enhancing retention and transfer	Summary[1]

[1] These categories are based on the classification of Ballstaedt (1997)
[2] Based on the vocabulary of the LOM Learning Resource Type (IEEE LTSC 2002)
[3] Based on the ContentModule types of LMML (Süß 2000)
[4] Advanced organizers were introduced by Ausubel (1960) as a cognitive instructional strategy to preview the information to be learned through what the students already know.
[5] "Definition" is a special form of expository contents.

Although Table 4.1 relates the different didactic content types to the instructional events of Gangé, this should not impose the strict use of an instructional theory based on Gangé's work. Rather, this association should illustrate the granularity level at which the didactic content types are aiming. We leave it to the teachers to present the contents in a didactic manner that they think is appropriate.

Aimed at a clear aggregation hierarchy, Content Elements integrate media representing the learning contents through a well defined structure. This structure is provided by a *structured markup schema*. Different markup schemas may be used for Content Elements. We present two markup schemas, a general schema for expository contents and a separate schema for multiple-choice questions (see Section 4.2.2).

Thus, a Content Element is defined as *a small, modular unit of learning content, which:*

- *Serves as a basic building block of learning content,*
- *Can be aggregated into larger, didactically sound Learning Units,*
- *Is self-contained or self-standing (it has no references to other Content Elements but may have prerequisites to the learners' knowledge),*
- *Represents a single didactic content type which relates to a single objective (a fact, a concept, a principle, a procedure, a process),*
- *Is reusable in multiple instructional contexts (reusable for other learning objectives or target learner groups as well as reusable with different instructional designs),*
- *Contains Assets.*

Assets

Assets are media elements such as images, videos, animations, simulations, etc. They are basically binary data objects which cannot easily be divided into smaller entities. Generally, they contain pictorial or auditory information which can be static (image, graph) or dynamic (video, audio, animation). Furthermore, they can be interactive programs that are to be embedded into Content Elements.

The term "Asset" is in accordance with the SCORM and Learnativity content models (see Section 3.4) for the basic media layer.

From a conceptual point of view, textual information can also be considered to be a media element. The Content Element's structured markup schema may be seen as a model which combines textual elements and other media elements and which provides a clear aggregation hierarchy.

However, as textual and other media elements may be intermixed, there is a problem with treating text pieces as individual resources. It is unclear how to define contiguous textual resources. Looking at the structured markup schema which integrates textual media into the Content Element, one could separate the pure character data from the structural information or one could divide textual information on a paragraph level.

But if individual text resources are based on pure character data, we might end up with resources containing only a single character. For example, a markup element for superscripts, relating to the mathematical power function, might contain only a single digit which clearly does not make much sense as an individual resource. If textual information is split into individual resources on a paragraph level, a header or the items of a list may be treated as individual resources. Still, it may be doubted whether or not a header or a single list item is useful as an individual resource out of its context.

Furthermore, as the Content Elements are already based on a level of granularity, the didactic content type level, which, from our point of view, is considered to be the appropriate level for learning contents to be optimally reusable, it does not make sense to define a finer grained level for individual text fragments. Instead, we propose to relate textual information as a whole to a Content Element.

Therefore, from a technical as well as a content author's point of view, a Content Element is treated as a unit which specifies the media aggregation structure together with the textual information. The Content Elements embed other binary media resources, as they are Assets, through reference.

Learning Units

A *Learning Unit* shall be defined as an aggregation of Content Elements to be presented to the learner. Typically, a Learning Unit serves as an online lesson. Unlike Learnativity's Learning Object or Cisco's RIO (see Section 3.4), both of which are designed to teach a single learning objective, a Learning Unit may be of any size and may also be used for multiple objectives. In our opinion, this enhances the flexibility to adapt Learning Units to the teachers' needs of organizing the material.

Learning Units provide the structure to assemble Content Elements into a larger unit. In order to clearly separate the navigation from the contents in a web-based environment, the arrangement of contents into pages and the navigation between these pages are solely based on this structure. In a paper-based environment, this structure is used to assign Content Elements to chapters, sections, etc.

The Learning Unit structure is designed as a chapter-like, hierarchical structure of nodes. Each node is associated to a Content Element through reference. The Content Elements are not copied into the Learning Unit but are referred to by links.

As a starting point, we define a Content Element to be presented as a single page in a web-based environment. This should provide a clear association of the Learning Unit structure to web pages.

Because Learning Units may be any size, no further aggregation level for Learning Units is provided.

4.2.2 Structured Markup

This section presents the *structured markup* to be used as a flexible data format for the Content Elements. A structured markup schema provides a data model for the Content Element's internal structure and the integration of the media representing the contents. The markup should identify meaningful structural information rather than styling information in order to separate contents and visual presentation.

XML provides the means to create special-purpose markup languages. XML markup consists primarily of elements and attributes.

Each markup *element* has a name which identifies the nature of the content enclosed. The contents may be character data and other elements. The nesting of elements into one another creates a well defined structure. Every XML document has exactly one top element called the root element. The root element integrates the whole data in a strict hierarchical manner. Four types of elements can be distinguished depending on the contents they may contain: *simple content* (only text), *element content* (only other XML elements), *mixed content* (text mixed with other elements), or e*mpty content* (the element does not specify any contents). Empty elements are often used as references to external resources using an attribute (see below).

Furthermore, elements can include *attributes* which provide additional information that relates to the contents enclosed. Attributes are typically used to specify the type of contents more precisely, e.g. to identify the language or a specific data format of the contents, or to add processing information, such as a URL for links or the path to a binary data resource which should be included by the presentation software. Three attributes types are used by the markup schemas presented here: *CDATA* (a string of characters), *Enumeration* (a pre-defined list of names which may not contain spaces or punctuation), and *NUMBER* (a decimal number).

The structured markup schema formally describes the set of markup elements allowed, the elements' types, the way the elements can be nested (repetitions, sequences, alternatives), the attributes an element may have (required and optional attributes), and the attributes' types.

Based on our conclusion that it is difficult to define a standard schema for learning contents that integrates the various needs of different teaching methods (see Section 3.5 and 3.7), we propose providing separate schemas for the exposition of learning contents and for special didactical issues.

As the proposed size of the Content Elements represents a single didactic content type, a simple structured markup schema based on standard typographical elements, such as headers, paragraphs, lists, tables, etc. may be sufficient for expository content. The specific didactic purpose, e.g. a definition, an example, etc., can be assigned to the component as a whole by a didactic content type metadata item. Using standard typographical elements, it is anticipated that the schema is familiar to content authors. Furthermore, contents using such a schema are likely to be easily convertible to possible future data formats. The markup schema remains stable, even if new didactic content types are needed – new types can flexibly be added as metadata.

Questions and tests, as they are special didactic contents, are provided by an additional structured markup schema.

Below, two markup schemas are presented – a general schema based on typographical elements for expository contents, the *Content Element* markup schema, and a separate schema for multiple-choice questions, the *Questionnaire Content Element* markup schema.

Content Element Markup Schema

The structured markup schema for Content Elements containing expository material is based on the XML schema used for documents by Silva. This schema already provides the traditional basic typographical elements which we found to be sufficient. The schema is further supported by Silva's built-in online editor. A detailed specification of the schema can be found in Appendix C.

The elements of the markup schema can be divided into two classes: *block elements* and *inline elements*. Block elements are displayed with a paragraph break before and after the elements. Generally, they contain character data mixed with inline elements. Some block elements contain other block elements. The block elements provided are:

- *Headings:* Four different heading elements are defined: `Heading`, `SubHeading`, `ParagraphHeading`, `SubParagraphHeading`. They are used for different levels of subtitles. The textual contents may be mixed with a restricted set of inline markup, such as `Emphasis`, `Superscript` and `Subscript`.
- *Paragraphs:* Paragraph elements are the most common elements for structuring text. Next to the basic `Paragraph` element, some more specialized paragraph types are available. The `LeadParagraph` can be used to set off an introduction. Side remarks which do not belong to the main flow of text can be marked up by an `Annotation` element. These three elements may contain character data together with inline elements. The `Preformatted` element preserves line breaks, tabs, multiple spaces, and thus, the original formatting. This element is typically used to mark up program code. Citations can be set off using the `Citation` element. This element may be composed of multiple paragraphs. It also contains two sub-elements which refer to the `Author` and the `Source` of the citation.

- *Lists:* The `List` element is used to specify bullet (unordered) and numbered (ordered) lists depending on the `ListType` attribute. The list items are marked up by `ListItem` elements and contain text and inline elements. The `NestedList` element provides a more elaborate way to create lists. The `NestedListItem` contains other block elements rather than inline elements in order to integrate sub-lists, images, tables, etc. into lists. It has the same `ListType` attribute as the `List` element. The `DefinitionList` element varies from the other list types in that list items consist of two parts: a `DefinitionTerm` and a `DefinitionDescription`.
- *Tables:* The `Table` element is used to define tabular data. It is composed by two types of rows: the `RowHeading` for titles which spans across all columns, and the `Row` element, which contains the table fields. Finally, the `Field` element may contain other block elements which add content to the table.
- *Images:* The `Image` element embeds an image. The image Asset is referred to by the `Path` attribute. The `Alignment` attribute defines the position relative to the surrounding contents. Images may be positioned horizontally with text above and below, or they can "float" to one side of the contents area with subsequent text flowing along an image's side. Images can serve as hypertext links using the `Link` attribute.
- *Multimedia elements:* Multimedia Assets are embedded using Silva's `ExternalSource` element. These elements provide a way to define a specialized set of parameters and presentation methods needed for the different types of multimedia Assets. We have integrated `ExternalSource` types for Quicktime movies, Flash animations, and Java applets.

Inline elements are generally elements which are embedded in the flow of text and are represented without any obvious line breaks. From a presentation point of view, these elements are generally displayed using another font, size, font style (bold, italic, underlined) or color. However, they may also be presented with no distinction at all. Inline elements contain character data and possibly other inline elements but they never contain block elements. They are used to emphasize specialized data or add some functional behavior to text fragments, such as hyperlinks. The inline elements provided are:

- *Basic inline elements:* The basic inline elements are `Emphasis`, `Strong`, and `Underline`. Typically, `Emphasis` is presented using italics and `Strong` is presented using bold fonts. The `Superscript` and `Subscript` elements and are frequently used for footnotes, mathematical and chemical notations (e.g. "x^2", "H_2O"), etc. They have no attributes.

- *Links:* The `Link` element defines hyperlinks to other resources. The target resource is defined by the `URL` attribute.
- *Index Items:* The `IndexItem` element provides a means to define *index terms* in a Content Element, which can be used to automatically create index pages. The `IndexItem` should be inserted right after the word or the phrase to be indexed. The `name` attribute contains a normalized keyword or key phrase (e.g. singular form for nouns, nominal forms for verbs, etc.). As an empty element, the `IndexItem` is not used to highlight any words. It defines an anchor point in the contents to which an index page can provide a hypertext link.

Questionnaire Content Element Markup Schema

In order to provide a simple interactive type of Content Elements for questions and tests, the *Questionnaire Content Element* has been specified. Because of the limited development resources available, an implementation of the *IMS Question and Test Interoperability specification (QTI)* or its light weight version *QTI Lite* was not realizable (mainly because of the large effort needed to provide an online editor for such a questionnaire schema). We therefore decided to develop a new XML-schema for simple single choice and multiple-choice questions. The Questionnaire Content Element is primarily designed to be used for self-assessments. Therefore, it provides the means to specify questions, possible answers, and to indicate the correct answers. The schema allows for the specification of a "hint", which is intended to provide some clues to help the students to solve the test. Figure 4.2 shows an example of how a Questionnaire Content Element may be presented to students.

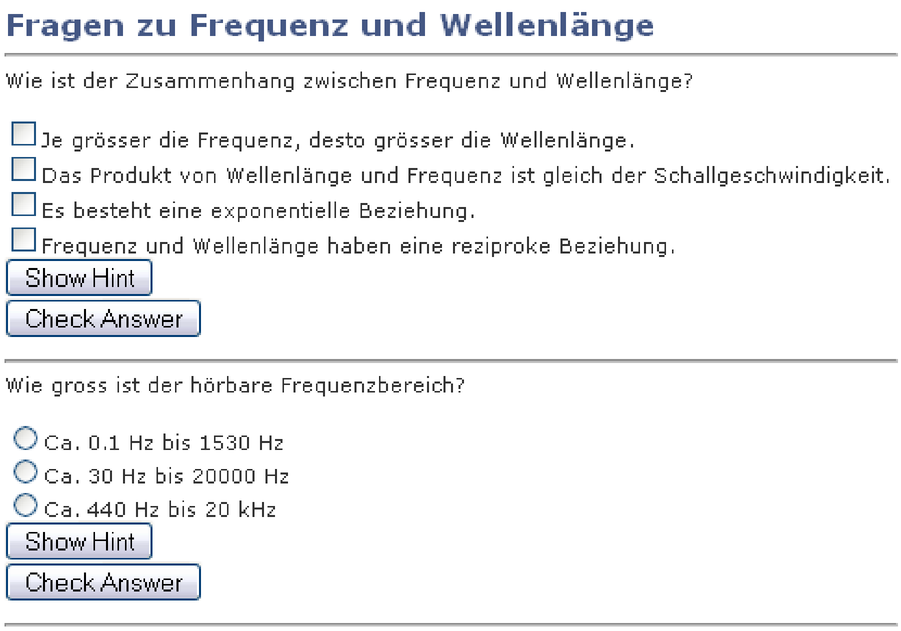

Figure 4.2 A screenshot of a Questionnaire Content Element, containing a title, a multiple-choice and a single choice question.

The proposed structured markup scheme for *Questionnaire Content Elements* defines the root node as `QuestionnaireContentElement`. It combines exactly one `Title` element and one `QuestionnaireContent` element. The title contains simple text with no other markup. The contents section is composed of a sequence of single choice and multiple-choice questions.

```
QuestionnaireContentElement ::= (Title, QustionnaireContent)
Title ::= (#PCDATA)
QuestionnnaireContent ::= ((SingleChoice | MultipleChoice)*)
```

The `SingleChoice` and the `MultipleChoice` elements have the same structure, a sequence of one question, several answers, and an optional hint.

```
SingleChoice ::= (Question, Answer*, Hint?)
MultipleChoice ::= (Question, Answer*, Hint?)
```

The `Question`, `Answer`, and `Hint` elements contain simple text with no other markup.

```
Question ::= (#PCDATA)
Hint ::= (#PCDATA)
Answer ::= (#PCDATA)
```

The `Question` and the `Hint` element do not have any attributes. The `Answer` element has an attribute to signify that the answer is correct. The schema does not ensure that single choice questions contain only one correct answer. This has to be done by the Questionnaire Content Element editor.

Attribute name	Attribute type	Description
`Correct`	*Enumerated:* correct	The attribute has only one possible value. Using the attribute signifies that this answer is correct. Omitting the attribute means the answer is incorrect.

4.2.3 Metadata

A set of *metadata* elements, associated to *Content Elements*, enables flexible search and retrieval (the detailed specification of the metadata set can be found in Appendix D). As metadata is not the main focus of this work, a minimal metadata scheme has been developed, keeping in mind that this scheme will need further improvement in the future.

The metadata set is designed to be compatible with the mandatory elements specified by the SCORM metadata information model (ADL 2001a) and the

ARIADNE Educational Metadata Recommendation (ARIADNE 2002). Both the SCORM and the ARIADNE metadata models are based on the IEEE LTSC Learning Object Metadata (LOM) Standard (IEEE LTSC 2002).

Additionally, we have included a category to the resulting metadata set to describe the *didactic content type* of a Content Element using the vocabulary defined in Table 4.1: e*xpository, definition, narrative, instruction, example, excursion, glossary, literature, summary, learning objectives, advance organizer, exercise, questionnaire, exam, self-assessment, experiment, problem statement,* and *simulation.*

Based on the work of Merrill (1983), Barrit and Lewis (2000), and Guttormsen Schär et al. (2004), we further define the *content category* metadata item which may be set to one of the following values: *fact*, *concept*, *principle*, *procedure*, or *process*.

4.3 Functional Architecture

This section describes the functional architecture and implementation of the dLCMS prototype. It aims at providing a functional tool to handle and process modular learning contents using the information model described above. We outline the functional architecture (Hackos 2002; Schluep et al. 2003) which consists of the following four primary functional components: *authoring*, *repository*, *assembly and linking*, and *publishing and export* (see Figure 4.3). The components are described as follows:

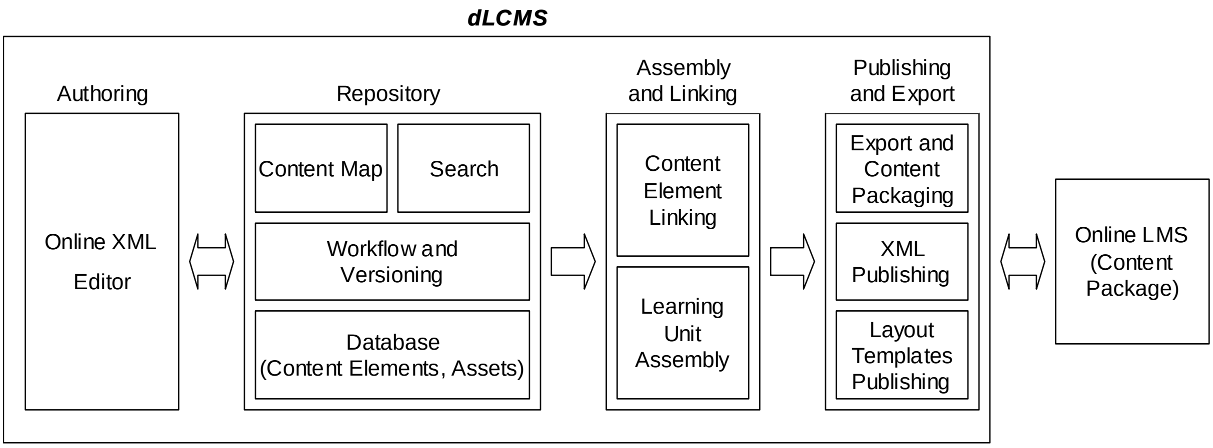

Figure 4.3 The dLCMS functional architecture.

1. *Authoring:* An online editor enables authors to create structured markup for Content Elements without worrying about programming languages and graphical design issues. No specialized editor software needs to be installed on the users' computers.
2. *Repository:* The repository stores the Content Elements and Assets. The learning resources contained can be accessed using the content map or using full-text or metadata search. The content map provides the means to

organize the contents. The repository further includes workflow and version management. It allows one to collaboratively (re-)use the learning resources.

3. *Assembly and linking:* This stage provides the means to assemble the Content Elements into the Learning Unit and to build up a hierarchical, chapter-like structure. This structure is the basis for the navigation in a web-based environment or for the division into chapters and sections using paper based media. The linking associates this structure with the Content Elements.
4. *Publishing and export:* The XML and layout template publishing stages apply the style and layout specifications to the Learning Units and allow a flexible graphical design. This stage further contains the functionalities to export Learning Units using standardized packaging formats, such as the IMS Content Packaging or the SCORM formats (ADL 2001a; IMS 2001a).

The next section provides a concise overview of the technology used and the implementation of the dLCMS prototype.

4.3.1 Technology and dLCMS Implementation

The implementation of the dLCMS prototype is designed to be an extension of the open source content management system *Silva* (Infrae 2005). The main reasons for choosing Silva as a basis for the implementation were:

- Silva offers an integrated online XML editor.
- The open source software is easily extendable and adaptable.
- Silva is being implemented as the university's standard web content management software.

Silva is a completely browser-based publication system used to create, manage, and publish contents for the web. Silva allows users to enter new documents as well as to edit existing documents using a simple web interface. Content is stored in XML. Silva supports workflow and version management. It can be extended by adding new types of content objects.

Silva is based on the Zope web application framework. Zope (Latteier et al. 2003) is an open source web server used to build content management systems, intranets, portals, and custom applications. Zope is written in Python (Python Software Foundation 2005), an object-oriented scripting language. Zope itself is highly object-oriented. Data and logic are stored in one or more objects using a hierarchical object structure – a Zope web application is composed of objects each of which may contain other objects. Zope has an integrated transactional object database which stores content and custom data, as well as templates, logic, and other complex services, for example, a search engine. Zope also features a tightly integrated security model that manages access rights and authorization.

The dLCMS extends the Silva system by defining new object types, which can be used together with the existing Silva objects (see Figure 4.4). The Silva/Zope system provides basic functions, such as database, search, XML editing and publishing.

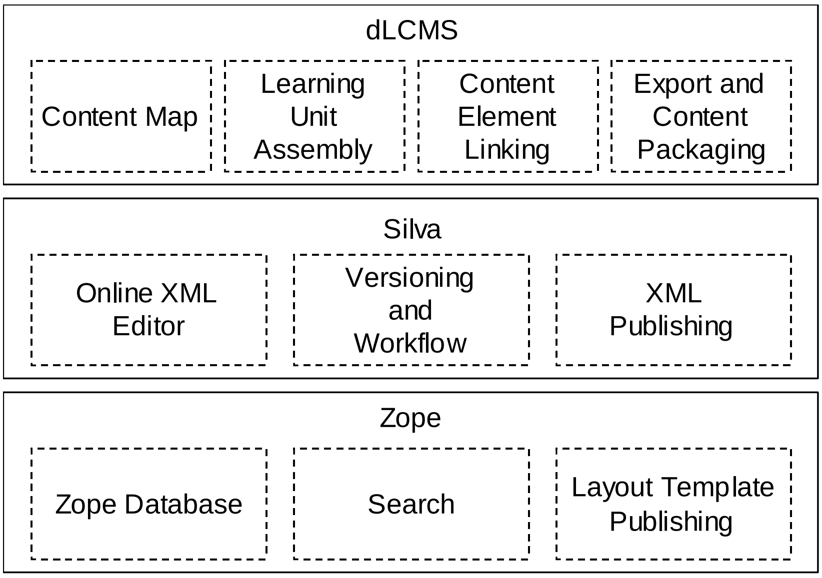

Figure 4.4 dLCMS is an extension of the Silva content management system which is based on the Zope web application framework.

The following sections describe the functional components in more detail: authoring, repository, assembly and linking, publishing and export, as well as the implementation.

Authoring

The online editor supports the creation of structured markup according to the schema presented in Section 4.2.2. As most authors of learning material are not computer specialists, they should not be forced to learn XML. The editor guarantees *valid* XML. Valid XML means that the structured markup schema is correctly applied according to the schema's set of elements and rules.

Silva provides two editors: the *Kupu* editor, and the Silva *forms editor*. With dLCMS, both editors may be used to edit expository Content Elements, while Questionnaire Content Elements can be edited only with the forms editor.

Kupu (Kupu 2005) is an open source, browser-based application, using JavaScript technology. It is cross-browser compatible, running on Internet Explorer, Mozilla, Firefox, etc. Kupu applies interaction techniques used by many popular editors, such as Word, OpenOffice: cursor positioning and text selection with pointing devices, drag and drop of selected text or other elements. A toolbar at the top contains controls to assign markup to selected text, such as paragraph types, headers, lists, and basic inline elements. It supports copy and paste operations.

Contents in HTML format can be copied from other applications into the Kupu editor. In this case, the contents are converted to the structured markup schema provided.

Therefore, contents from other HTML resources can be copied and pasted directly into Content Elements. Contents can also be copied from other formats if the application from which content is copied provides HTML as a format for copy and paste interactions. Therefore, authors can copy contents from Web-Browsers, Microsoft Word, OpenOffice, etc. into the dLCMS editor.

A screenshot of the Kupu editor displaying a Content Element is shown in Figure 4.5, and the numbers are explained below.

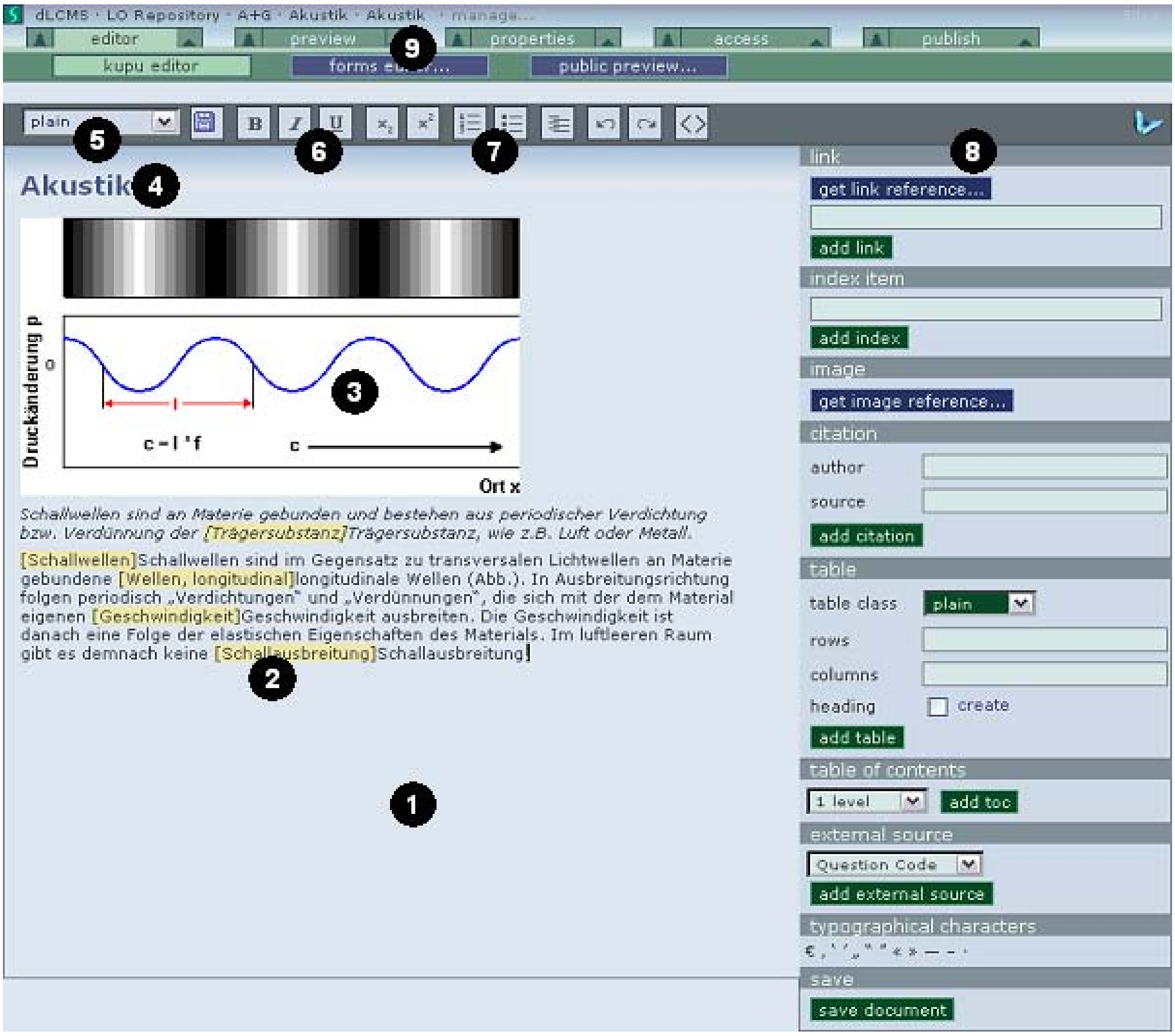

Figure 4.5 The Kupu online editor displaying a Content Element. The numbered items are explained in the text.

1. *Editing area:* Text can be entered and edited using keyboard or pointing devices. It allows text selection, copy and paste, drag and drop, etc.
2. *Highlighted index item:* The normalized keyword or key phrase of an `IndexItem` is displayed and highlighted.
3. *Inserted image:* Inserted images are displayed in the editing area and can be moved using drag and drop.
4. *Title:* The Content Element `Title` can be edited directly in the editing area.

5. *Selector for headings and paragraphs:* Special paragraph or heading types can be applied to paragraphs.
6. *Buttons for basic inline elements:* Selected text can be marked up as `Emphasis`, `Strong`, `Underline`, `Subscript`, or `Superscript`.
7. *Buttons for lists:* Selected paragraph or heading elements can be marked up as list elements.
8. *Sidebar for elements with attributes:* `Link`, `IndexItem`, `Image`, `Citation`, `Table`, or `ExternalSource` elements can be added and edited through this sidebar. It provides input fields for the corresponding attributes.
9. *Management tabs and navigation:* The management tabs provide access to other functions beyond content editing, such as *preview*, *metadata editing* (*properties* tab), *access rights*, and *publishing*.

The Silva forms editor uses HTML forms to edit the XML element's contents. Since it is based on standard HTML, the editor is cross-browser compatible. It can be adapted to XML schemas which are quite different from HTML, such as the Questionnaire Content Element markup schema. Still, the configuration is very complex and requires in-depth knowledge of the editors' architecture. Compared to Kupu, the forms editor requires many clicks to insert, edit, delete and move elements. It does neither support drag and drop between elements nor copy and paste of marked up contents.

A screenshot of the Silva forms editor displaying a Questionnaire Content Element is shown in Figure 4.6 and the numbers are explained below:

1. *Title:* The title of the Questionnaire Content Element can be edited by clicking on the dark arrow at the left side.
2. *Insert XML element:* New elements can be inserted by clicking on the light arrows at the left side of the screen.
3. *Moving and deleting XML elements:* These controls are used to delete elements or to move them up and down.
4. *Preview of a multiple-choice question:* Not all XML elements can be edited at the same time. These elements are previewed showing the questions, answers, and the checkboxes and buttons.
5. *Edit view of a single choice question:* The edit view of an element provides HTML forms to edit the text of the question, the answers, and the hint. Check boxes are provided to select the correct answer.
6. *Save:* These buttons are used to save the data entered. “Save” leaves the editing forms open, “save + exit” closes the forms, “save + insert” closes the forms and inserts a new multiple-choice question element.

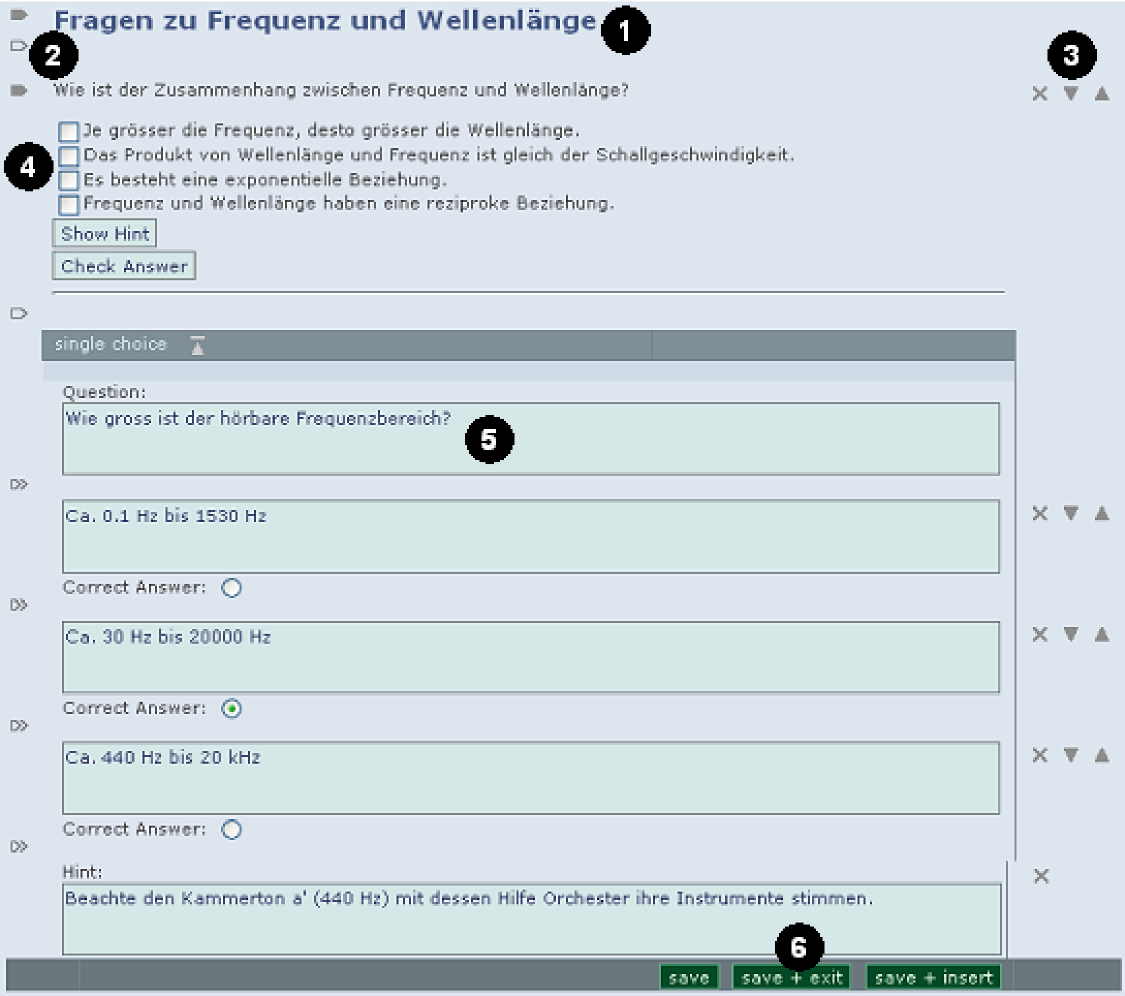

Figure 4.6 The Silva forms editor displaying a Questionnaire Content Element, of which a single choice question can be edited. The numbered items are explained above.

Repository

The main purpose of the *repository* is the central storage of Content Elements and Assets, making them accessible for collaborative use. As this work focuses mainly on modularization and structured markup, our development goal for the repository was to provide basic functionality for the centralized storage of learning resources. Therefore, the implementation of the repository does not make use of more advanced techniques of information retrieval. Thus, further research and development is needed in order to provide more sophisticated functions for the sharing of learning contents.

To enable navigation and to provide an overview of the contents contained in the repository, the learning resources are organized using a *content map*. The map is built up by the users.

The implementation is based on two object types: the base repository object and the subject folder object. The repository and the subject folder may contain further subject folders as well as Content Elements and Assets. The repository object serves as the root container containing all contents and providing an entry point to the users. The subject folders provide a means to organize contents into topics ("subjects") and subtopics in a hierarchical manner, thereby establishing a *content map*.

The repository is the basis for search and retrieval functions. Full-text search as well as search based on metadata are supported. Search is based on the functionality provided by Silva and Zope.

The Content Elements contained in the repository can inherit certain metadata items from their containing repository and subject folders. These metadata items are: *discipline*, *subdiscipline*, *context*, *institution*, *cost*, *restrictions*, *rights* (see Appendix D), which are likely to apply to most of the Content Elements inside a single subject folder or even inside a whole repository. This relieves the authors from having to specify all metadata items for every single Content Element.

A screenshot of a repository view is presented in Figure 4.7. The numbers are explained below:

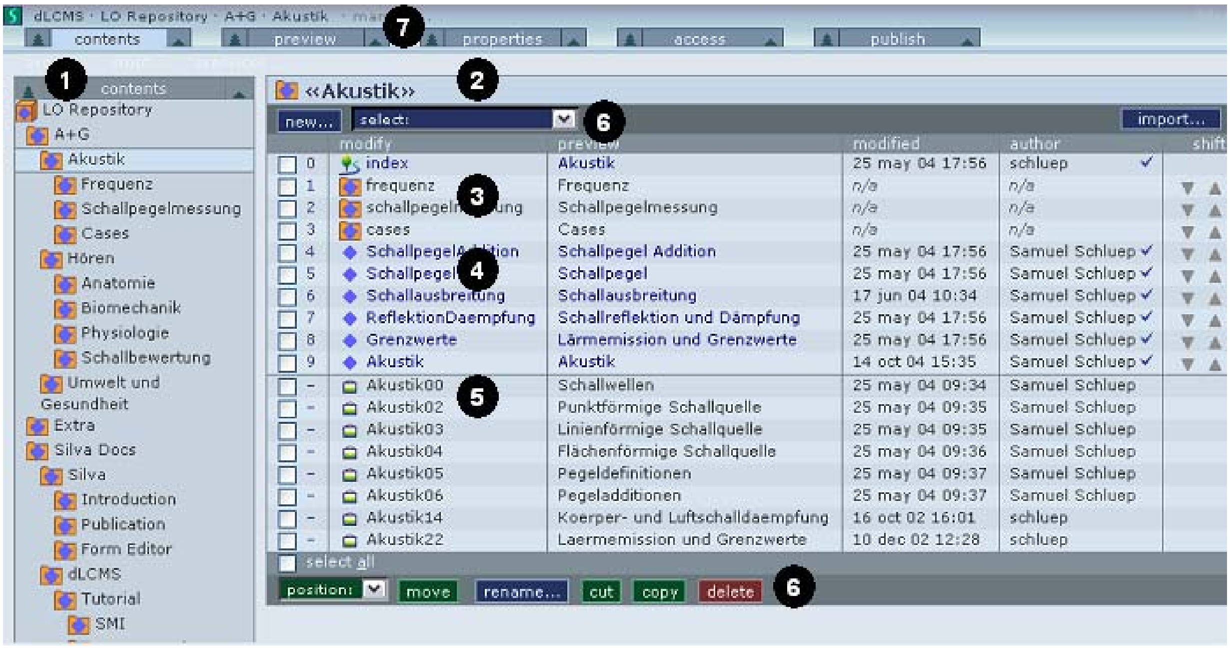

Figure 4.7 The repository view. The numbered items are explained below.

1. *Sidebar overview and navigation:* The sidebar contains the overview of subject folders, i.e. of the subject matter contained in the repository.
2. *List of contents:* On the right side, the list of contents of a repository object or a subject folder object is displayed. The listed objects may be subject folders (3), Content Elements (4), or Assets (5).
3. *Subject folders:* Subject folders are used to organize the contents into topics and sub-topics.
4. *Content Elements:* Content Elements are stored inside a subject folder.

5. *Assets:* Assets, images and other files are stored inside a subject folder and are listed separately at the bottom of the list.
6. *Management functions:* The controls provide functions to manage the contents, such as creating new contents and subject folders, and moving, copying, renaming or deleting existing contents.
7. *Management tabs:* The management tabs provide access to this content view and to other functions for preview, metadata editing, access rights, and publishing.

Assembly and Linking

The teachers can define the Learning Unit structure with the help of the *lesson* and the *lesson element* objects. The lesson object contains a whole Learning Unit that can be exported. Lesson element objects are added to create a hierarchical, chapter-like structure.

Each lesson and lesson element is associated with exactly one Content Element by reference. This reference is a string containing a path to the requested resource in the repository.

A screenshot of the lesson object view is presented in Figure 4.8. Lesson element objects have a similar view. The numbers are explained below:

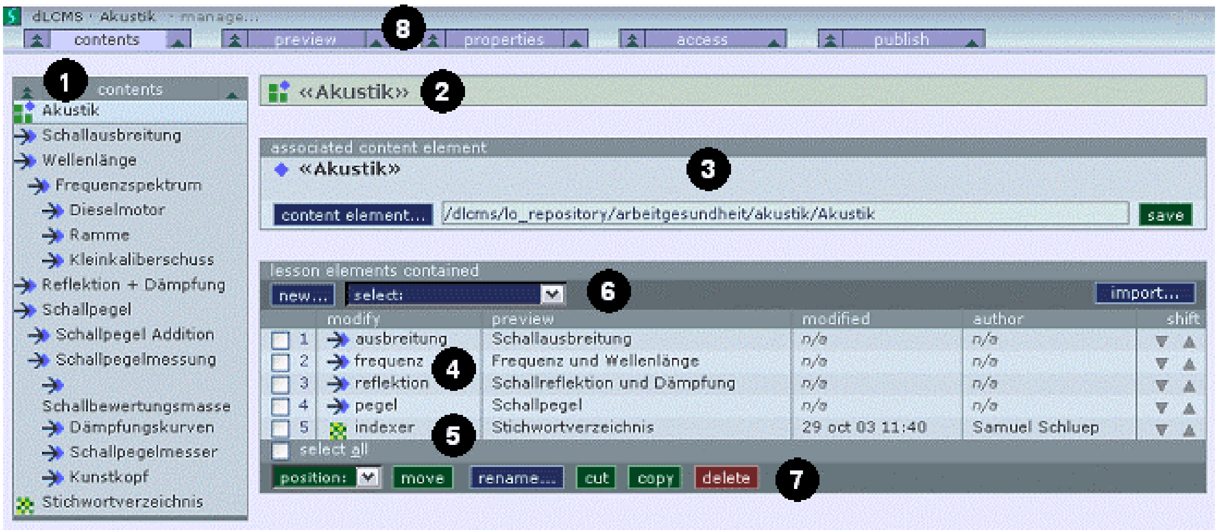

Figure 4.8 The lesson object view. The numbered items are explained below.

1. *Learning Unit structure:* The left sidebar provides an overview of the complete Learning Unit structure.
2. *Learning Unit title:* The right side contains the contents of a lesson or lesson element object: At the top, the title is displayed, which is the title of the associated Content Element.
3. *Associated Content Element:* This section shows the path to the associated Content Element. Content Elements can be selected using the "content

element....” button. This button opens a chooser window by which the user can select a Content Element. The chooser window provides two methods to find and select a Content Element: the user can browse the repository or the user can search for Content Elements using full-text or metadata search.

4. *List of subsections:* A list of *lesson elements* contains the direct subsections relative to the object shown.
5. *Index page:* Next to *lesson elements*, *index pages* can be added to Learning Units. These objects automatically create an index page. They scan all Content Elements associated to the Learning Unit for `IndexItem` markup elements and build up a page containing the keywords or key phrases with links to the containing Content Elements.
6. *Control to add new subsections:* New *lesson elements* or *index pages* can be added to the current object.
7. *Management functions:* The Learning Unit can be edited by moving, copying, renaming, or deleting lesson elements or index pages.
8. *Management tabs:* The management tabs provide access to other functions, such as preview, metadata editing, access rights, and publishing.

Publishing and Export

The publishing stage applies the layout and styling defined in layout templates to the Learning Unit pages. The rendering system provided by Silva and Zope does this in two steps (see Figure 4.9):

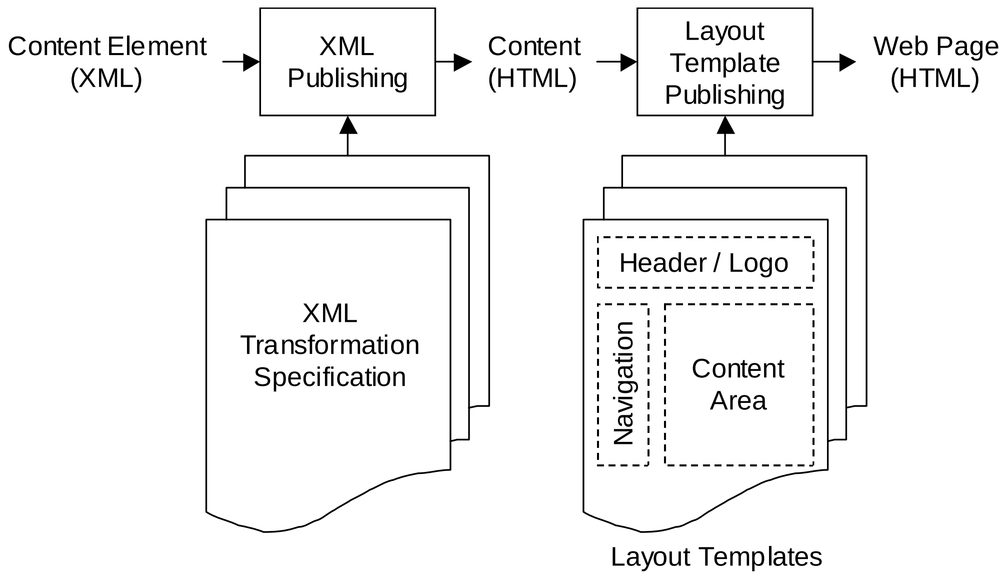

Figure 4.9 The two publishing steps for Content Elements: in the first step, the Content Element’s XML is transformed to HTML; then, the content is embedded into the web page layout which also defines the headers, footers and navigation.

1. The Content Element's XML data is converted to HTML by the Silva XML publishing engine.
2. The HTML content is inserted into the content area, defined by the layout template. The layout template defines the page frame with its headers, footers and navigation using the Zope Page Template technology.

The Content Element's XML data is transformed to HTML by the Silva XML publishing engine. The transformation of the XML elements is defined by the dLCMS and Silva XML transformation specifications. The original XML elements can be clearly identified by the name and the class attribute of the HTML element generated. Therefore, the final presentation style of the contents can be customized using Cascading Style Sheets (CSS).

Using the Zope Page Template technology, the system allows for easy customization of the page frame's layout. The look and feel of a Learning Unit can be adapted to specific corporate designs. Zope Page Templates define the basic HTML frame, i.e. headers, footers, and logos, as well as areas for dynamic contents. Dynamic HTML code is generated using Zope's Template Attribute Language (TAL). TAL attributes are used to create the navigation based on the Learning Unit structure and to insert the Content Element's data into the content area.

Content packaging, as specified by the IMS Content Packaging (IMS 2001a) or the SCORM specifications (ADL 2001a), is used to deploy Learning Units to external learning management systems (LMS). These packages are basically ZIP files containing a manifest file and the physical files, called *resources*, which are needed to display the Learning Units in a browser. The manifest file describes the Learning Unit structure, the resources, and how the resources are related to each other. All links referring to resources in the package must be relative. The Learning Unit export function packages the Learning Unit pages and the associated Assets to a ZIP archive and creates the appropriate manifest file. Three different output formats, which differ slightly, can be selected: IMS Content Packaging version 1.1.3, SCORM 1.2, or SCORM 1.3.

4.4 Résumé

The dLCMS prototype provides a tool which implements a *modularization* strategy combined with the *separation of content and presentation*.

The modularization strategy is defined by the *dLCMS component model*, consisting of *Learning Units*, *Content Elements*, and *Assets*. Content Elements are the basic building blocks of learning material which can be flexibly assembled into *Learning Units* to be used for various educational contexts. Content Elements should be self-contained and their granularity should be based on what we call *didactic content types*. A Didactic content type may be seen as a piece of learning content which relates to a didactic step used to teach a topic, such as a definition, an example, an exercise. Using didactic content types, we can flexibly combine components with a high potential for reuse together with elements which apply more specifically to the needs of students of a particular scientific discipline. We further anticipate that this will promote the development of Content Elements with a similar level of granularity.

The Content Elements' structure is defined by *structured markup schemas* which integrate the learning media and provide an abstraction of the visual presentation, thereby separating contents and presentation. The dLCMS includes separate schemas for the exposition of learning contents and for multiple-choice questions. The markup schema of expository content is based on the approach of providing a simple set of elements based on traditional typographical elements, such as headings, paragraphs, emphasis, etc., which we expect to be familiar to content authors. Contents using this schema are likely to be easily convertible to possible future data formats. The markup schema remains stable, even if new didactic content types are needed – new types can flexibly be assigned to Content Elements using metadata. The markup schema for questions and tests offers a structure for simple choice and multiple-choice questions.

The functional architecture consists of four general functional entities: the *online editor* which provides an easy-to-use tool for authors to create valid structured markup, the *repository* for collaborative (re-)use of learning resources, the *assembly and linking* stage which allows for flexible aggregation of Content Elements into Learning Units, and the *publishing and export* stage enabling flexible graphical design and export of Learning Units as interoperable, standards-compliant content packages.

The dLCMS provides a basis to investigate the applicability of modularization and structured markup from the point of view of learning content authors in an academic environment.

5 Learning Unit Development Guidelines

This chapter aims at providing a chunking method which supports learning content authors to create Content Elements representing didactic content types. The Learning Unit Development Guidelines presented in this chapter are based on the general phases of Instructional Systems Design (ISD) and have been specially developed for the dLCMS project with its approach to base the granularity level of Content Elements on didactic content types. Special attention was given to the content chunking process. It provides a three step procedure: (1) First, the content should be divided into topics and subtopics, each of which is related to a single concept, fact, procedure, process, and principle. (2) Then, the didactic content types to be used should be assigned to every topic or subtopic. For every single didactic content type, a separate Content Element should be created. (3) At last three to five other potential learner groups should be identified. The Content Elements specified by that point should be analyzed for reuse with the potential learner groups.

5.1 Objectives

Learning objects are a new way of thinking about learning content. Authors of learning resources may need guidance to adapt their thinking about learning material, which traditionally had been whole courses or lecture notes, with respect to modularized contents.

Primarily, 4.2.1the *Learning Unit Development Guidelines* aim to provide a *content chunking method* which supports content authors to divide learning contents into self-contained Content Elements which are defined by the dLCMS Component Model (see 4.2.1). In particular, the guidelines should assist authors to create Content Elements which, as a standard granularity level, represent single *didactic content types*.

The proposed procedure should not impose the strict use of a single instructional design theory. The procedure was developed under the assumption that the authors are the experts with respect to the contents to be taught and the appropriate didactics. It was further assumed that the authors have experience with traditional forms of teaching. Therefore, the procedure should be flexible enough to support their approaches to lesson planning.

In examining the literature, a large number of procedures for creating e-learning applications are related to Instructional Systems Design (ISD). The process of the various ISD models can be summarized into five general phases: *analysis – design – development – implementation – evaluation* (Braxton et al. 1995; Kerres 1998; Issing 2002).

Therefore, the Learning Unit Development Guidelines are also based on ISD. Special attention is paid to the content chunking process, i.e. breaking learning contents down into modular Content Elements.

Secondary, the Learning Unit Development Guidelines were also intended to provide a structure which clearly identifies the different phases of the investigation of the learning unit development process. Research questions and results can be directly related to the different development phases.

5.2 Development Phases

In order to clearly support the process of chunking learning contents into Content Elements and assembling Content Elements into Learning Units, the *design* and the *development* phase of the general ISD model are extended. The design phase is divided into a *Learning Unit Concept* and a *Content Chunking* phase. The development phase contains a *Content Development* and a *Learning Unit Assembly* phase. Thus, our learning content development process consists of seven phases: *Learning Unit Analysis*, *Learning Unit Concept*, *Content Chunking*, *Content Development*, *Learning Unit Assembly*, *Teaching*, and *Evaluation* (see Figure 5.1):

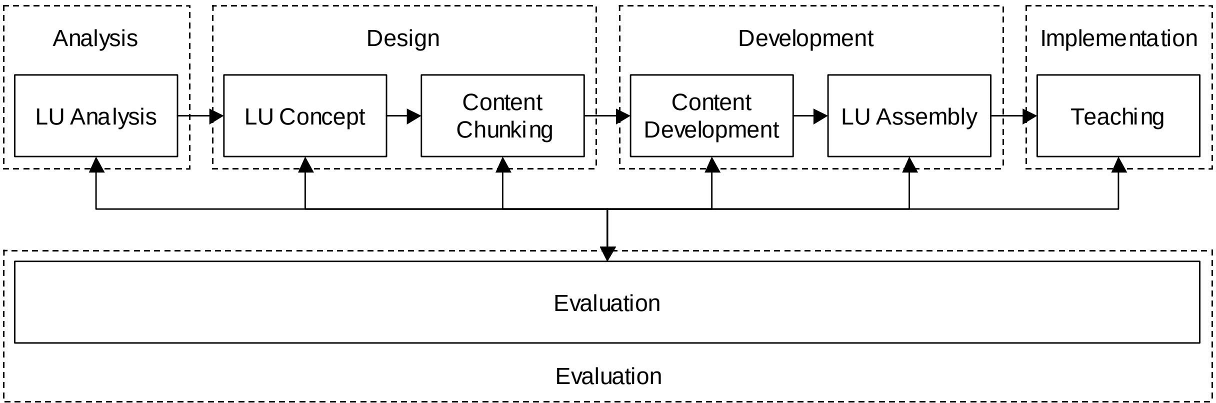

Figure 5.1 Based on the general model of ISD, we divide the learning Content Development process for modular contents into seven phases: Learning Unit (LU) Analysis, LU Concept, Content Chunking, LU assembly, teaching, and evaluation.

1. *Learning Unit Analysis:* The analysis phase includes an analysis of the subject to be taught, the major learning objective, the learner profile and the organizational environment.
2. *Learning Unit Concept:* The specification of detailed learning objectives, the selection of learning content, didactic strategies and methods to be used are specified in this phase.
3. *Content Chunking:* In this step, the contents are chunked into small, modular Content Elements which represent didactic content types.

4. *Content Development:* This phase involves the production of the specified Content Elements, including the media which is integrated into the Content Elements, e.g. images, animations, videos, etc.
5. *Assembly of Content Elements:* The modularized Content Elements are assembled into the final Learning Unit.
6. *Teaching:* Teaching refers to the deployment and the actual delivery of the final Learning Unit to students. This phase was named “teaching” instead of “implementation” because programmers often use the term “implementation” for the process of writing software code, which is analogous to the Content Development phase specified here.
7. *Evaluation:* The evaluation guides all phases in parallel.

To support learning content authors to create modularized content, we have worked out guidelines for the first three steps: *Learning Unit Analysis*, *Learning Unit Concept*, and *Content Chunking*. The development of these guidelines is based on existing work which describes instructional design procedures (Kerres 1998; Issing 2002; Baruque et al. 2003; LSAL 2003; Schüpbach et al. 2003).

The development guidelines do not cover the Content Development and Learning Unit Assembly phase because we anticipated that the dLCMS would sufficiently support the authors as it already provides the design of the graphical layout and the navigation. It was not clear how the new Learning Unit would be deployed at the time the guidelines were compiled. Therefore, the guidelines do not cover the teaching phase. We may provide further guidelines for the Content Development Learning Unit Assembly and Teaching phases in the future, which will be based on the experiences using the dLCMS.

In conjunction with the guidelines, we have developed the *Learning Unit Analysis* and *Learning Unit Concept Forms* which may be used to write down the results of these phases. The guidelines can be found in Appendix A.

In the following section, an overview of Learning Unit Analysis, Learning Unit Concept, and the Content Chunking phases is presented.

Learning Unit Analysis

The analysis phase is important for the specification of a learning unit which should be well adapted to the learners’ needs. Looking at the available resources, it should also provide assurance that the project is realizable. It generally includes an analysis of the subject to be taught, the learner profile and the organizational environment.

Subject matter: The general idea of the learning unit to be created should be specified in this section. This includes the subject matter to be taught, the major learning objective, and some information on the coverage of contents. It should define the general didactical setting: how the e-learning unit is to be integrated into regular class (e.g. stand-alone unit, blended learning), the final qualification for learners (e.g. credit points, certificate), and how long students will be expected to learn using the unit. The suitability of the subject matter for e-learning and modularization should be analyzed. For the analysis of the subject matter's suitability for e-learning, the criteria of Schulmeister (2003) are presented. As a support for judging the suitability for modularization, the guidelines contain the Content Element definition (see Section 4.2.1). An analysis of existing learning contents which may be reused is suggested. Such content may be reused directly or it may serve as a basis of materials to be re-edited for the new learning unit.

Learner profile: The analysis of learner characteristics is important for the choice of learning methods. The learner profile analysis contains the educational level (e.g. undergraduate, graduate or post-graduate), knowledge and professional prerequisites, age, language of the students, and the ratio of men and women. The guidelines propose to analyze learner motivation and learning habits (e.g. if the students are accustomed to self-administered learning). The learner profile also contains expected previous e-learning and computer experiences and analyzes the computer equipment to which students have access.

Environment: The analysis of the environment should identify the resources that are available for the learning unit development. The guidelines include available e-learning platforms, development software (as the dLCMS is a development tool, it belongs to this group), financial resources, and personnel resources.

Learning Unit Concept

The Learning Unit Concept phase should provide a clear conceptual specification of a final Learning Unit. This includes the detailed learning objectives, the learning contents to be presented, and the general didactic strategy.

Therefore, this phase starts with a task analysis of the skills to be taught and/or a content analysis of the subject matter. Based on the results of this analysis, the major learning objective, specified in the Learning Unit Analysis phase, should be deconstructed into more detailed sub-objectives yielding the topics to be taught. Then, the overall didactic strategy and methods should be specified. The guidelines propose the following steps:

1. *Task analysis / analysis of the knowledge domain:* Depending on the type of instruction, skills for a specific task (what the learner should be able to do) or knowledge on specific subject matter (what the learner should know), different approaches are suggested. In the case of skills, the task should be deconstructed into sub-tasks performing a task analysis. For the education of general knowledge, the knowledge domain should be analyzed reflecting the theme to be taught and its logical structure. This results in an overview of sub-topics, their possible structures, and their relevance.
2. *Learning objectives:* Based on the results of the last step, the detailed learning objectives should be formulated.

3. *Learning contents:* This step defines the contents to be used to teach the learning objectives. As a result, the topics and their sequence are briefly described.

4. *Didactic strategies and methods:* The overall didactic strategies and methods to be used with the Learning Unit shall then be specified. For example, whether the learner should be guided step-by-step through the learning unit (instruction-led expository design) or whether the contents can be explored by the student using a hyperlink structure (learner-centered explorative design). As an alternative, one may refer to specific didactic methods, such as training, tutorial, simulation, game, problem-oriented learning, mastery learning and others. Then, the didactic methods should be worked out in more detail, specifying how to present the subject matter, how to involve the learners, and what media should be used.

5. *Graphical structure of the e-learning unit:* The learning unit may be represented graphically to allow for the visualization of its structure and navigation.

Content Chunking

The *Content Chunking* phase is the most important step to modularize the contents into modular building blocks. It provides a method to break down the content, which has been specified in the Learning Unit Concept phase, into Content Elements. Relating to the Content Elements definition (see 4.2.1), this procedure should ensure that the final Content Elements represent single didactic content types which are related to single concepts, facts, procedures, processes, or principles. Furthermore, the Content Chunking method provides an approach to create Content Elements which are reusable for different potential audiences.

The Content Chunking method consists of the following steps:

1. *Breaking contents down into single objectives:* The content should be divided into topics and subtopics, each of which is related to a single objective. To help to identify single objectives, the items should be labeled as *concepts, facts procedure, processes,* and *principles* (content categories). This step is based on the works of various authors (Barritt et al. 2000; South et al. 2000; Baruque et al. 2003).

2. *Specifying the didactic content types:* Following our approach to base Content Elements on didactic content types, every topic or subtopic specified in the first step should be assigned the didactic content type to be used for teaching. Every didactic content type should represent a separate Content Element. The proposed list of didactic content types is described in Section 4.2.1.

3. *Analysis for reuse with other learner groups:* Then, three to five other potential learner groups for the subject matter should be identified. Each Content Element specified so far should be analyzed to determine if it can be potentially reused for the other learner groups identified. If only some pieces of a Content Element seem to fit the needs of a potential learner group, one should consider dividing the Content Element so that the pieces can be reused individually.

As a tool to perform the Content Chunking procedure, we propose using a *Content Chunking Spreadsheet* (e.g. an Excel spreadsheet, see Table 5.1). The lines represent the Content Elements. From left to right, columns are labeled as “topic/subtopic”, “content category” (concept, fact, procedure, process, and principle), “didactic content type”, and “target learner group”. Additional columns are used to indicate the potential reusability of a Content Element for other learner groups. The Content Chunking Spreadsheet allows one to continuously refine the content chunking process. New Content Elements can be added by inserting a new line. The spreadsheet provides a concise view of the evolving learning unit structure. The use of the spreadsheet is also flexible enough to jump back to earlier steps when needed. An example of a Content Chunking Spreadsheet is available as an Excel file.

Table 5.1 Example of the Chunking Spreadsheet.

Topic/Subtopic	Content Category	Didactic Content Type	Target Learner Group	Other Potential Learners			
				Group 1	Group 2	Group 3	Group 4
Topic 1	Concept	Definition	X	X	X	X	X
Topic 1	Concept	Example	X	X		X	
Topic 1	Concept	Questionnaire	X		X		
Topic 2	Procedure	Instruction	X		X	X	X
Topic 2	Procedure	Exercise	X		X		X
…	…	…	…	…	…	…	…

5.3 Résumé

The *Learning Unit Development Guidelines* presented in this chapter aim at providing a *content chunking method* which supports content authors to divide learning contents into self-contained Content Elements.

The procedure is based on the general phases of Instructional Systems Design (ISD). It has been developed under the assumption that the authors are the experts with respect to the contents to be taught and the appropriate didactics. Therefore, the procedure should be flexible enough to represent the author’s approach to lesson planning and it should not impose the strict use of a single instructional design theory.

Special attention was given to the content chunking process to ensure that the final Content Elements represent a didactic content type and that they are reusable for different potential audiences.

Furthermore, the seven steps of the Learning Unit Development Guidelines – Learning Unit Analysis, Learning Unit Concept, Content Chunking, Learning Unit Assembly, Teaching, and Evaluation – provide a means to investigate the development process in a structured manner.

6 Pilot Projects Study

This chapter presents an evaluation of the dLCMS and the Learning Unit Development Guidelines, focusing on modularization and structured markup from a learning content author's point of view. Authors from three different scientific domains, as well as one author working in the ICT services department of an academic environment, used the dLCMS to create a web-based learning unit for the education of students or university personnel. Formative evaluation served as a theoretical foundation and the investigation was guided by the principles of contextual inquiry. The findings showed that the proposed steps to modularize content described in the guidelines did not work well. The assignment of the content categories, concept, fact, procedure, process, and principle, was difficult and the analysis for other potential learner groups did not have any effect on the modularization structure. However, the participants perceived the guidelines as helpful to improve the didactic quality of the learning unit. Generally, the participants could create learning contents which corresponded to their expectations. In a few cases, the participants suggested that it should be possible to combine several Content Elements on a single page. Missing markup elements concerned mainly specialized markup for literature and glossaries. Additionally, the markup schema for multiple-choice-like questions did not suit the authors' expectations. No author perceived the structured markup as constraining their creativity.

6.1 Objectives

In this chapter, we present an evaluation of the dLCMS modularization strategy and structured markup schemas, described in Chapter 4, as well as the chunking method presented in Chapter 5. The evaluation focuses the points of view of learning content authors in an academic environment. The study primarily aims at investigating the questions of whether authors understand the modularization strategy, whether the chunking method supports the authors to create modularized contents, and whether the structured markup schemas are suitable for the authors' needs.

As the dLCMS and the content chunking method were considered to be still in a prototyping phase, this study followed a qualitative approach using observations and interviews investigating only a small number of participants. The observations and interviews were conducted at the work place of the authors, i.e. in their normal working context, while the authors applied the dLCMS to create learning material for their teaching needs. This method should allow one to gain a better understanding of the content author's perspective - including their goals, work practices, and work context. The study aimed to be open to discovering issues and problems not previously considered.

As a means of focusing our research more clearly, the next sections present our research questions and hypotheses. Then, our research method, describing the participants, the setting, and the procedure are subsequently presented. Section 6.3 contains a summary of the relevant results for each of the participants (the detailed results can be found in Appendix B). In Section 6.4, we discuss the results and present our conclusions in Section 6.5.

Content Chunking

As modularized Content Elements are a new way of thinking about learning content, authors may need guidance to adapt their thinking about learning material, which traditionally had been whole courses or lecture notes. This leads to the following research question:

RQ1: How can authors be supported to create modularized content?

The Learning Unit Development Guidelines, presented in Chapter 5, give special attention to how authors can be supported to divide learning contents into small modular Content Elements. The guidelines provide a step-by-step procedure using three steps: (1) breaking the content down into single objectives, (2) specifying the didactic content types to be used, and finally (3) analyzing the elements for reuse with other potential learner groups. The following hypotheses are formulated in order to evaluate the content author’s ability to perform these steps:

H1.1: Authors are able to identify single objectives assigning the content categories, “concept”, “fact”, “procedure”, “process”, or “principle” to the contents.

H1.2: Authors are able to identify single didactic content types.

H1.3: Authors are able to analyze the contents for reuse with other potential learner groups.

Modularization

A general interest of this study is to investigate the author’s understanding of the modularization concept:

RQ2: Do authors understand the concept of modularization?

Based on the Content Element’s definition (Section 4.2.1), the following hypotheses were formulated to investigate this question:

H2.1: Authors can create Content Elements which are self-contained (self-contained Content Elements have no references to other units but may have prerequisites with respect to the learners’ knowledge).

H2.2: Authors can create Content Elements which represent a single didactic content type.

H2.3: Authors can create Content Elements which relate to a single objective (a fact, a concept, a principle, a procedure, a process).

These hypotheses do not make any assumptions about how the authors would chunk learning content into small pieces. Even if the authors were provided the guidelines to create the modular units, they would be free to modularize the content in a way that was found to be most comfortable. This should provide insights into the user's intuitive understanding of modularization and it may reveal alternatives to the proposed content chunking method as a whole or for specific steps.

The Content Element definition further specifies that Content Elements "can be aggregated into larger, didactically sound Learning Units" which relates to the next research question:

RQ3: Can small, self-contained Content Elements be aggregated into didactically coherent learning units?

This research question contains two aspects of Learning Units: The ability to *aggregate* Content Elements into Learning Units and the final Learning Units being *didactically coherent*. This is expressed by the following two hypotheses:

H3.1: Authors can aggregate Content Elements into larger Learning Units.

H3.2: Modularization influences the didactic quality of the final Learning Unit from an author's point of view.

In this study, we focus on the author's point of view of the final Learning Unit's didactic quality. In Chapter 7, a study is presented which investigates the student's point of view.

Clearly, authors would only be willing to apply this new paradigm of learning content production if they believe that this new way of content handling would pay-off:

RQ4: Does modularization pay-off from an author's point of view?

Our initial hypothesis will simply be formulated as follows:

H4.1: Modularization pays-off from an author's point of view.

Other factors might also influence modularization of learning content. One such factor could be the subject matter. Not all types of subject matter may be suited to be divided into small, self-standing building blocks:

RQ5: Which subject matter is suited for modularization?

Although this study cannot profoundly investigate which types of subject matter are suited for modularization, we may, nevertheless, consider this research question. Here, too, the initial hypothesis is straightforward:

H5.1: Is any subject matter suited for modularization?

Structured markup

The structured markup schema for learning contents, provided by the dLCMS, has been based on the assumption that a simple structured markup schema with traditional typographical elements, such as headings, paragraphs, emphasis, etc., may be sufficient for different kinds of learning contents. This assumption should be verified.

RQ6: Is a simple structured markup schema, based on traditional typographical elements, sufficient for learning contents?

This study aims at an evaluation of the proposed markup schema from an author's point of view. The initial hypothesis is formulated as follows:

H6.1: The structured markup schema, based on traditional typographical elements, is sufficient to design Content Elements which are in accordance with the authors' expectations.

In order to keep the basic markup schema for learning contents simple and flexible, the dLCMS provides special didactic functionality, i.e. markup for questions and tests, as separate Content Elements. In order to evaluate this approach, we ask if authors need didactic markup to be directly integrated into the basic markup schema or if such specialized contents can be integrated as separate components.

RQ7: What kinds of specialized didactic markup are needed and can they be provided through separate markup schemas for separated components?

According to the approach taken, the following initial hypothesis is used to investigate this issue:

H7.1: Specialized didactic markup can be provided as separate components.

The idea that structured markup helps authors to focus on contents without having to worry about formatting and graphical design, leads to the last research question. On the one hand, structured markup may be perceived as an aid. On the other hand, a structured markup schema restricts possibilities to compose learning contents and may be perceived as a constraint to creativity.

RQ8: Do authors perceive structured markup as an aid or as a constraint to creativity?

Our hypothesis is formulated as follows:

H8.1: Authors do not perceive structured markup as constraining creativity.

6.2 Method

6.2.1 Participants

The participants of the study were employed at the Swiss Federal Institute of Technology and were all involved in the development of e-learning material. They come from the domains of natural sciences, engineering sciences, and social sciences, as well as from the ICT services department. Table 6.1 gives an overview of the participating authors.

Table 6.1 Overview of the authors participating in the study.

	Author 1	Author 2	Author 3	Author 4
Domain	Natural sciences, medicine	Social sciences, history	ICT services, software training	Engineering sciences, ergonomics
Tasks	Education; research	Implementation of e-learning content; teaching assistance	ICT training; support for software systems	Research on and development of e-learning material
Duration of employment	4 years	3 years	13 years	7 years
Subject matter experience[1]	20 (- 30) %	60 (- 70) %	100 %	60 (- 70) %
E-learning know-how[1]	90 %	90 %	5 %	60 %
Previous e-learning authoring experience	3 years; conception of an e-learning project; implementation of a proof of concept	3 years; coauthoring, search for and editing of images for e-learning; some authoring of texts	None	5 years; development of interactive electronic learning material
Previous LMS authoring experience	Evaluation of different LMS products	Course development with OLAT	Development of an exercise using WebCT	None
Previous CMS authoring experience	Silva	Silva	Silva	Silva
Previous LCMS authoring experience	MySSQL, had a look at dLCMS and IBT-Server	Had a look at dLCMS	Had a look at dLCMS	dLCMS

[1] The authors were asked to rate their “e-learning know-how” and “subject matter experience” using a scale from 0% to 100%, where 0% means no experience or know-how and 100% stands for “expert in this field”.

6.2.2 Task

The participants' task was the development of a learning unit which should teach a topic from their knowledge domain. They were free to choose any specific subject. The learning unit was to be developed by applying modularization and structured markup using the dLCMS tool. The learning unit was to be designed for web-based learning and it could be an exposition of the subject matter and/or an exercise. The participants were free to choose the didactic strategy and methods which they believed would best suit their purposes. The duration of the learning unit was to correspond with a traditional lesson or lecture (45 – 120 minutes).

6.2.3 Setting and Procedure

Formative evaluation served as a theoretical foundation for this study. As the development of the dLCMS and the Learning Unit Development Guidelines were still works in progress, the results of the study should be used to guide future research and development. Aimed at participative development, this approach allows one to observe and interact with real users working on actual e-learning projects in their everyday environments. The approach should reveal the advantages as well as the problems of the proposed concepts as applied to actual e-learning content; it should enable the participants to actively influence the future development.

The following points are important characteristics of a formative evaluation and apply to this study (Clases et al. 2001):

a) The data gathering focuses primarily on the clarification of the goals accompanying the process and the illumination of problems which arise during the application of the process.

b) The role of the evaluator is an interactive one.

c) In general, qualitative methods are used.

d) For credible participation, an agreement with the developers and participants is needed, as well as appreciation and confidence on both sides.

The participant's development of a learning unit using the dLCMS tool was conducted together with a two-person evaluation team at each participant's work place. The evaluation team consisted of a coach and an observer. The coach led the interaction with the participant and supported the authors where needed. The observer recorded observations of the participant's performance and the conversation between the participant and the coach during the interaction by note taking.

As the Learning Unit Development Guidelines and the dLCMS were works in progress, it was expected that not everything would necessarily be clear and comprehensible right from the start. The coach was to try to get an understanding of the participant's plans and intensions, as well as an understanding of the experiences and problems encountered during the learning unit development. The interaction and data gathering were guided by the principles of *contextual inquiry* (Holtzblatt et al. 1993):

1. *Context:* ”The best way to understand the work is to talk to people in their actual work environment. Design information is present in its richest form when we speak with people during ongoing work or using work artifacts (Holtzblatt et al. 1993).”
2. *Partnership:* Users (in our case the content authors) are acknowledged to be the experts of their work. In order to have an effective dialogue with the users, it is important to establish partnership. Two techniques support this: sharing the control of the conversation and creating a shared meaning between participants and investigators.
3. *Focus:* Focus directs the conversation with the users. On the one hand, the focus is fixed by the initial goals and the questions of a study. On the other hand, the focus should be expanded or shifted if the goals and assumptions are not appropriate to the actual situation.

The participants were explicitly asked to voice their comments or questions loudly. They were also encouraged to take steps other than the ones proposed by the guidelines if they wanted to do so and to explain why they would do so.

In order to investigate the development process in a structured manner, the development process was divided into distinct phases as proposed by the Learning Unit Development Guidelines: Learning Unit Analysis, Learning Unit Concept, Content Chunking, Content Development, Learning Unit Assembly, Teaching, and Evaluation. Focusing mainly on the authoring process, this study did not cover the Teaching phase. Furthermore, this study was intended to serve as an Evaluation phase that should be conducted in parallel to the other phases.

The following procedure was used to evaluate the different phases of the learning unit development process:

1. Kick-off Meeting
2. Analysis and Concept Workshop
3. Chunking and Content Development Workshop
4. Continuing autonomous development

Kick-off Meeting

The *Kick-off Meeting* was held together with all participants. They were informed about the concepts of modularization and structured markup, the dLCMS tool, the goals of this study, and the procedure planned. The participant’s task, i.e. the development of a learning unit (see Section 6.2.2), was explained and discussed.

There was an opportunity to discuss the concepts, the participant's task, as well as the goals of the study.

Analysis and Concept Workshop

The main goal of the *Learning Unit Analysis* and *Learning Unit Concept* phases from a research point of view was to get an understanding of the author's plans for the final learning unit. This was to provide a basis with which to compare the outcomes of the whole development with the author's intentions.

The *Analysis and Concept Workshop* was held about one month after the Kick-off Meeting with each participant individually. The duration was approximately two hours. The participants received the guidelines and forms for the Learning Unit Analysis and Learning Unit Concept a few days before the workshop took place so that they had the opportunity to take a look at the guidelines and forms in advance.

The workshop was conducted at each participant's work place. After an introduction, saying hello and giving an overview of the goals and the procedure of the workshop, we proceeded through the following steps:

1. *Semi-structured interview on the participant's data (personal data, work place, computer and e-learning experience) and their expectations regarding participation in this study as a pilot project.*

2. *An opportunity to discuss open questions about the guidelines.*

3. *Contextual inquiry while carrying out the Learning Unit Analysis and Learning Unit Concept step-by-step, based on the guidelines.*

   The participants were to conduct an analysis and create a concept of the learning unit they intended to develop. It was proposed that they follow the guidelines and write their results to the *Learning Unit Analysis* and *Learning Unit Concept Form*s.

   In addition to the judgment of the subject matter's suitability for e-learning and modularization, as proposed by the guidelines, the participants were asked to rate the suitability for e-learning and the suitability for modularization, respectively, on a scale from 0% to 100% (0% meaning unsuitable, 100% meaning very suitable). The participants were also asked to give an example of a Content Element.

   The observer was to record observations and conversation by note taking. At the end, we collected the completed Learning Unit Analysis and Learning Unit Concept Forms, as well as other artifacts, which documented the plans and intensions of the participant. These artifacts could have been existing online courses, learning material that the participants wanted to use, previously existing concepts, etc.

4. *Final debriefing to reduce the stress and to discuss the general impressions of the workshop.*

Chunking and Development Workshop

The *Chunking and Development Workshop* was conducted a few weeks after the *Analysis and Concept Workshop*. The time in between should have enabled the authors to collect their learning materials that they wanted to integrate into the new learning unit. The duration of the workshop was to be four to six hours. Both parts of the workshop, Content Chunking and Content Development, were allotted approximately the same time duration.

Again, the workshop was carried out with each participant individually. The participants received a copy of the Content Chunking guidelines and the Content Chunking Spreadsheet example a few days in advance via e-mail.

The workshop took place at each participant's work place. After saying hello, giving an overview of the goals and the procedure of the workshop, we proceeded through the following steps:

1. *Contextual inquiry while carrying out the Content Chunking process step-by-step, based on the guidelines.*

   Beginning with the Learning Unit Concept, the participants were to chunk the learning content into Content Elements conceptually. As a result, the learning unit was to be created as a list of Content Elements, describing the topic and didactic content type of each Content Element as well as the structure of the learning unit. It was proposed that they follow the Content Chunking guidelines and use the Content Chunking Spreadsheet to perform this process.

   The focus for the evaluation of the Content Chunking phase was guided by the research questions RQ1 (H1.1 - H1.3), RQ2 (H2.1 - H2.4), RQ3 (H3.2), RQ4 (H4.1), and RQ5 (H5.1).

   In addition to providing support where needed, the coach of the evaluation team aimed to create an understanding of the participant's comprehension of modularization.

   The results of the contextual inquiry were recorded by note taking. At the end, the Content Chunking Spreadsheet, containing the modularization result, was to be collected.

2. *Semi-structured interview about the author's experiences of the chunking procedure.*

   At the end of the chunking process, a semi-structured interview was conducted. The interview questions, IQ1 to IQ7, were formulated as follows:

   IQ1: How did you create your Content Elements? Did you use the guidelines or did you use some other method?

IQ2: Is the subject matter of your learning unit suited for e-learning?

IQ3: Is the subject matter of your learning unit suited for modularization?

IQ4: How did the modularization process influence your thinking about the learning unit compared with your habitual way of development?

IQ5: What are the effects of modularization on the learning unit itself, compared to your habitual way of development?

IQ6: Is modularization inspiring and motivating or rather, is it laborious?

IQ7: Does modularization pay-off?

The participants' answers were recorded by note taking.

3. *Contextual inquiry of the Content Development and Learning Unit Assembly process.*

   Using the dLCMS tool, the participants were to create the Content Elements for their learning unit using the dLCMS. The Content Elements were then to be assembled into a Learning Unit.

   The evaluation goals of this phase and the initial focus of the contextual inquiry were based on the following questions: RQ3 (H3.1 - H3.2), RQ6 (H6.1), RQ7 (H7.1), RQ8 (H8.1).

   The results of the contextual inquiry were to be collected by note taking. The final Content Elements and the Learning Unit which had been developed with the dLCMS remained in the system for further analysis at a later time.

4. *Semi-structured interview about the authors' experiences using the dLCMS tool and structured markup.*

   The semi-structured interview questions (IQ8 - IQ14) at the end of the development phase focused on the research questions RQ6, RQ7, and RQ8:

IQ8: How do you judge the dLCMS tool in general?

IQ9: What was the most difficult aspect to create Content Elements with the tool?

IQ10: Using the set of markup elements (e.g. 'paragraph', 'heading', 'list', 'table' etc.), could you do what you wanted to do?

IQ11: Did the set constrain your creativity?

IQ12: Was the provided set of elements sufficient or did it contain too many elements?

IQ13: Were there elements missing and if so, which ones?

IQ14: Which problems related to the dLCMS should be solved?

The participants' answers were recorded by note taking.

5. *Final debriefing to reduce the stress and to discuss the general impressions of the workshop.*

Continuing Autonomous Development

Clearly, the workshops did not provide enough time to develop a Learning Unit covering the contents which corresponds to a traditional lesson of 45 to 120 minutes. It was suggested that the participants continue the development of Content Elements and the Learning Unit using the dLCMS tool on their own. Support for the participants was provided by e-mail, which should also serve as a means for further data collection.

6.2.4 Data Analysis

The data gathered can be grouped into two main parts which were analyzed separately:

1. *Workshop and e-mail support data:* This data consists of the notes taken during the workshops, i.e. observations and interviews, e-mail data to support the autonomous development of the learning unit, and the data contained in the Learning Unit Analysis and Learning Unit Concept Forms, which were completed by the participants.
2. *Final Content Elements:* The other source of data to be analyzed was the final Learning Units and Content Elements that the participants had developed.

Analysis of Workshop and E-mail Support Data

The workshop and e-mail support data provided a rich source of information about the participants' plans and intensions, their understanding of the dLCMS modularization and structured markup concepts, as well as their experiences and problems encountered during the learning unit development using the Content Chunking method and the dLCMS tool. Because this information was spread over several data sources, i.e. observation notes, results of the semi-structured interviews, e-mails, and the Learning Unit Analysis and Learning Unit Concept Forms, the data was analyzed if it did contain information which related to one of the research questions of hypotheses (see 6.1). The pieces of text that were identified as containing relevant information were copied and grouped into categories that related to the research questions and hypotheses.

Analysis of the Final Content Elements

The analysis of the final Content Elements intended to examine whether or not the participants were able to create Content Elements according to the definition presented in Section 4.2.1. The Content Elements were analyzed according to the following criteria: being self-contained (hypothesis H2.1), representing a single didactic content type (hypothesis H2.2), relating to a single objective, a concept, fact, procedure, process, or principle (hypothesis H2.3).

Since the dLCMS Component Model was applied for the first time, the analysis aimed to investigate whether, in principle, authors understood the concept of Content Elements. Therefore, the analysis was not intended to be an in-depth semantic analysis of every single sentence. Rather, we tried to identify key statements of the Content Elements. The analysis was carried out by the author of this work who tried to be as objective as possible using the definitions provided. The detailed procedures of the Content Element analysis were performed as follows:

Self-Contained

Content Elements were rated to be "self-contained" if they had no explicit reference to other contents. References could be either hypertext links or explicit linguistic reference in the text, i.e. words or sentences referring to other contents (see Section 4.2.1).

Single Didactic Content type

In order to analyze whether a Content Element represents a single didactic content type, the following steps were performed:

1. Identify the fragments of the Content Element which represent a didactic content type.
2. Assign the didactic content type categories from the list presented in Section 4.2.1 to the fragments. If no category in the list suits the contents, propose a new category which suits the contents.
3. Count the number of fragments identified. If only one fragment is identified, the Content Element is judged to represent a single didactic content type.

Single Objective

To check if a Content Element is related to a single objective the following procedure was carried out:

1. Identify the fragments of the Content Elements which relate to different objectives.
2. Assign one of the content categories, concept, fact, procedure, process, or principle to each fragment using the definitions presented below.
3. Count the number of fragments relating to different objectives. If only one fragment is identified, the Content Element is judged to be related to a single objective.

The definitions of the content categories used for this analysis were based on the work of Barritt and Lewis (2000). However, Barritt and Lewis' definition of the *principle* category is, in our opinion, much too oriented towards industrial training. It was therefore expanded using Merrill's definition of a principle (Merrill 1983). As a result, the following definitions were used for the analysis:

- ***Concept:*** *A concept is used to teach a group of objects, symbols, ideas, or events. Concepts are designated by a single word or term, share a common feature, and vary on irrelevant features.*
- ***Fact:*** *A fact is used to teach unique, specific, one-of-a-kind pieces of information. Facts are presented as statements, data, or pictures of specific objects.*
- ***Procedure:*** *A procedure is used to teach a performance.*
  - *A procedure is a sequential set of steps to be followed by one individual to accomplish a task or make decisions.*
  - *A procedure lists directions for procedural tasks.*
  - *Actions within a procedure must be done the same way each time (within a given situation).*
- ***Process:*** *A process is used when you need to teach how a system works. This is helpful in supporting underlying job tasks, providing motivation, and ensuring overall quality of job performance. A process can be defined as:*
  - *A flow of events that describes how something works*
  - *Not a task to be done by one person*
  - *Many persons or organizations are involved*
  - *Mechanical, business, or scientific*
- ***Principle:*** *Principles are explanations or predictions of why things happen in the world. They are cause-and-effect, correctional, or constraint relationships. Principles are also used when a task requires judgment or when guidelines must be applied to a situation.*

6.3 Results

This section presents a summary of the relevant results for each participant separately, each of which contains two parts:

- *Profile of the participant:* This part contains the motivation and expectations of the author with respect to taking part in this study. Then, the participant's previous experiences as a learning content author are summarized. Finally, the agreement on the scope of participation between the author and the evaluation team is described.
- *Summary of results:* This part presents the relevant results obtained from the analysis of the workshop and e-mail data and the analysis of the final Content Elements. The results are ordered according to the development

phases: Content Chunking, Content Development, and Learning Unit Assembly. Then, the results of the Content Element analysis follow.

No results of the Analysis and Concept Workshop are presented here because the data gathered for these phases did not turn out to be useful to investigate the research questions that are formulated in Section 6.1. The detailed results of all development phases and the results of the final Content Element analysis can be found in Appendix B.

6.3.1 Author 1

Promoting the dLCMS project was the main motivation of Author 1 in participating in this study. As a benefit, the participant hoped to gain new ideas for the didactic realization of e-learning applications, e.g. for the sequencing and the presentation of learning units.

The participant was a research assistant. She was already developing an e-learning course on refraction (the procedure to measure the correction needed for eyeglasses and contact lenses) for ophthalmologists and eventually also for opticians. At the beginning of her project, she had almost no previous knowledge on the subject matter.

She had already developed a concept for the course and implemented an example of a chapter with a proprietary system. This system used open source database and web server software. It allowed simple HTML-editing and the presentation of the e-learning course via a web browser. The system and the course were stored on a CD-ROM, which could be installed on a server or handed out to students directly. This would allow ophthalmologists to use the course without web access, which was anticipated to be the preferred way for ophthalmologists to work with e-learning applications.

The participant agreed to implement selected parts of her existing contents using the dLCMS as a test. She decided to participate in the *Analysis and Concept Workshop* as well as in the *Chunking and Development Workshop*. She did not intend to use the learning unit to be created with the dLCMS to teach real students.

Summary of Results

Content Chunking

Already having contents, which were divided into sections, she started to apply the chunking guidelines to these sections in a somewhat reverse engineering manner. In the first step, *breaking contents down into single objectives*, she had difficulties in assigning the given set of content categories to the existing sections. She stated that the content categories were difficult to understand and that the given set of categories did not seem to suit all of the contents contained in the existing sections. In the second step, *specifying the didactic content types*, some of the terms given were not clear (*advanced organizer* and *self-assessment*). Most of the existing sections were divided into two didactic content types using *expository* and *glossary*. The other didactic content types used were *test*, *definition*, *example*, and *animation*. For the third step, *analysis of other potential learner groups*, the participant could not identify any group other than the target group.

While the participant did not think that the subject matter was suited for e-learning, she rated the subject matter as suitable for modularization.

The participant stated that the content chunking procedure was helpful. It inspired her to divide the learning unit into smaller sections and in the future she would try to structure the contents more clearly with less exploratory text as compared to the learning contents she had already developed. In her opinion, modularization would yield to a better didactic structure as it would allow learners to navigate more easily to areas of interest and to what they think is important. She judged that a clear chunking method would pay-off because contents need to be modularized for a web-based learning application anyway.

Content Development

Generally, the structured markup schema provided was sufficient. Copying the contents from the existing online course, the participant was not even aware of the restricted set of structured markup elements. The only markup elements that she was missing were glossary items, “expert comments” (a summary of the things to remember, which were presented in the original system using synthesized speech), and a more elaborate content type for multiple-choice questions. Therefore, it can be suggested, that she did not perceive content structuring as constraining creativity.

Learning Unit Assembly

The Content Elements created could be aggregated into a larger Learning Unit but the possibility for assigning a separate title to Learning Units was missing.

Analysis of the Final Content Elements

In the end, six Content Elements were created, all of which were clearly self-contained. However, analyzing the didactic content types and the single objective was difficult. Many Content Elements did not clearly point out key statements. Rather, they were a mixture of several fragments of information which were somehow related to each other. It was not clear if such an element should be rated as one “extensive” expository element relating to one single objective or if the various fragments should be looked at as separate pieces of learning contents. From our point of view, this lack of clarity affected the didactic quality of the Content Elements adversely. Therefore, we decided to rate each fragment individually. The problems to create clearly structured Content Elements might be due to the fact that the participant was not a subject matter expert and that the original source of the contents, as they were lecture notes and books, did not contain clearly structured contents.

In the end, only two Content Elements could be clearly rated to represent a single didactic content type representing a single objective. One Content Element did relate to a single objective but it contained two didactic content types. The remaining elements represented neither a single didactic content type nor a single objective. Figure 6.1 shows the one Content Element representing two didactic content types, expository text and an example used to teach the procedure of the “crossed cylinder technique”.

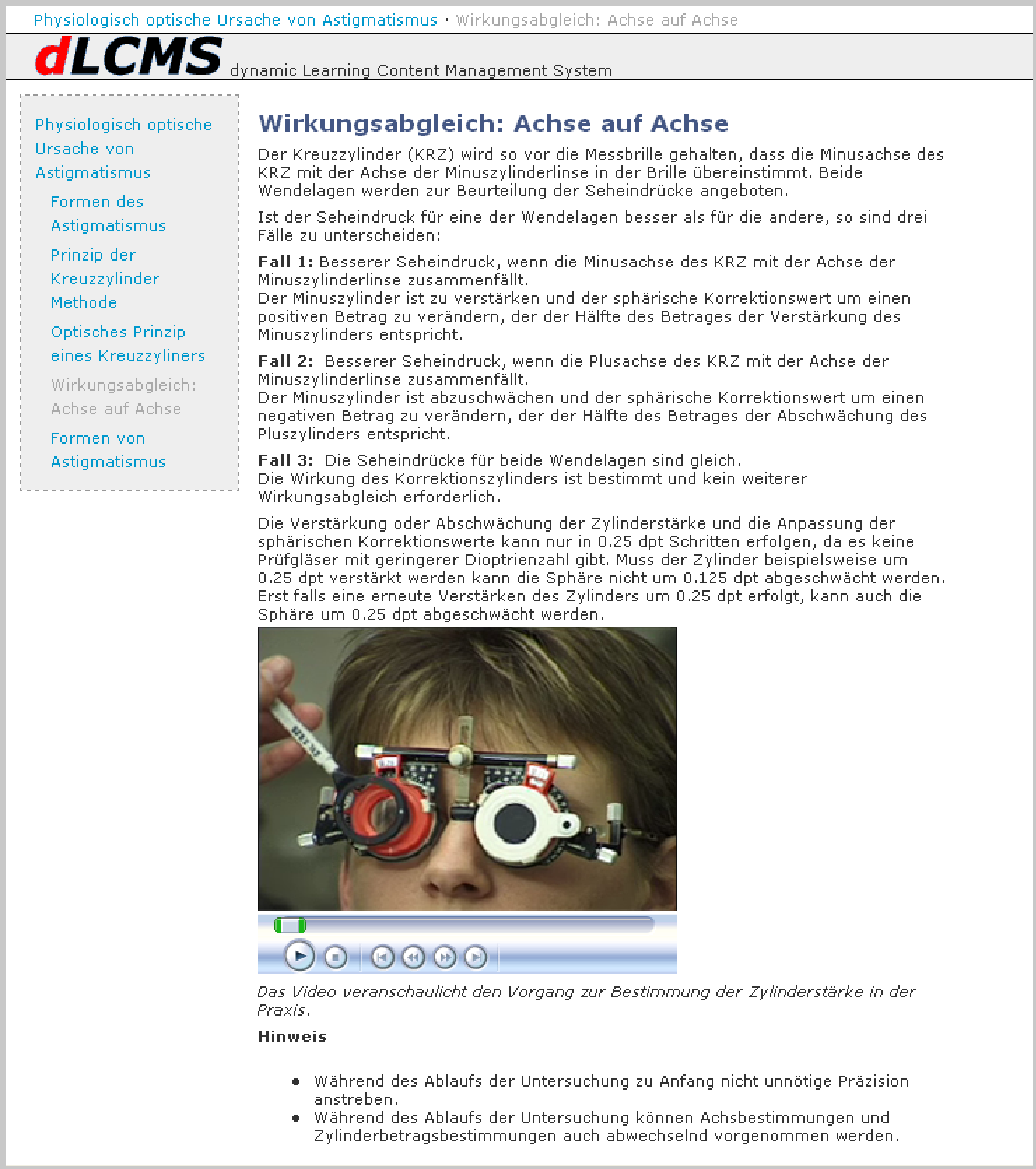

Figure 6.1 Screenshot of a Content Element created by author 1 (CE1.5). It describes the steps and decisions for the “crossed cylinder technique” and contains a video showing a practical example of the procedure.

6.3.2 Author 2

The main motivation for Author 2 to participate in the study was to get to know alternative products for the development of web-based learning. The participant wanted to benefit from new technology being offered at the university. She had had positive experiences using non-commercial products, especially in using the open source OLAT learning platform of the University of Zürich. To her, the advantages of non-commercial products offered the possibility of actively influencing the development of the system (suggesting new features) and not having to pay for new features. As a benefit of participation in this study, she hoped to see alternatives for the implementation of web-based learning and to gather new ideas.

The participant was a member of a development team, which just had completed a web-based course project called “Webclass”. It covered the “History of Technology”. The Webclass team had developed four different modules, each of which was a semester course. The courses were created for the OLAT learning platform using Macromedia Dreamweaver as a HTML development tool. The participant had been involved in the development of the original material but another person had been responsible for the concept of that lesson.

The participant proposed to transfer a page from the existing contents of the Webclass course “History of Energy” to the dLCMS in order to evaluate the system. She agreed to participate in the *Analysis and Concept Workshop* and the *Chunking and Development Workshop*. She did not intend to use the resulting learning unit to teach real students because the existing Webclass courses had already been successfully implemented.

Summary of Results

Content Chunking

The selected contents, the page of the Webclass project to be transferred, already contained small sections. These sections were the basis for the content chunking performed in this study. An example of such a section which contains a title, some text, and an image, is shown in Figure 6.2

The participant started to apply the steps of the Content Chunking guidelines to these existing sections. In the first step, *breaking contents down into single objectives*, the participant was not able to assign the content categories to the existing sections. The content categories simply did not seem to fit the contents. The second step, *specifying the didactic content types*, did not have any impact on the chunking results. In the third step, *analysis for reuse with other learner groups*, all Content Elements were considered to be reusable for all potential learner groups identified. Therefore, the Content Elements specified until then were not divided any further. Thus, this step did not have any effect on the chunking results.

Figure 6.2 A section of the existing Webclass e-learning application. The section contains a title, some text, and an image. (Webclass-Team, ETH Technikgeschichte)

Although it was easy to create modularized Content Elements in this case, the participant rated the subject matter to be 50% suitable for modularization. Nonetheless, she rated the subject matter to be suitable for e-learning.

To her, the applied content chunking procedure did not influence the didactic quality of the learning unit created – the learning unit did not change significantly, except that each Content Element was presented on a separate page. She was not able to judge which was better – having all Content Elements on one page or presenting each element on a separate page. Her general estimation was that the Content Chunking procedure would probably yield a good didactic structure but that the extra effort would only pay-off if the contents were to be reused by other authors and if modularization was already considered at the beginning of an e-learning project.

Content Development

She could not transfer all content as expected into the dLCMS, specifically, the “material box” (a box containing links to bibliographic information and further resources) which should not be displayed as a separate Content Element. In general, she stated that the set of structured markup elements was sufficient but she thought that specialized markup was needed for reference information (such as bibliographies), the positioning of images next to text, and inline images. As she did not create new content, she could not judge if structured markup constrained creativity.

Learning Unit Assembly

The Content Elements could be assembled into Learning Units but the participant was not sure if each Content Element should be presented on a separate page. In the original online course, all content was presented on the same page. In her opinion, the Learning Unit should have a separate overall title which is different from the Content Element titles.

Analysis of the Final Content Elements

The participant created seven Content Elements. Two elements were not self-contained. One of these elements contained a linguistic reference to a preceding element using the word “also”. We may note that the contents had been originally edited to be on one page and were simply copied into several Content Elements for this study. The other Content Element not rated to be self-contained was a collection of links to further information resources.

Three Content Elements combined an expository and one or two examples. However, the expository and the examples were tightly coupled to each other so that these Content Elements appeared to be a single content entity. Still, to be precise, we rated them as not representing a single didactic content type. The remaining elements represented a single didactic content type. All Content Elements were clearly related to a single objective.

We may note that during the analysis, we assigned the new didactic content type category “introduction” to one of the Content Elements. In our opinion, none of the categories provided suited the contents.

Other Results

Generally, the participant had doubts as to whether her team would want to share the content to be collaboratively used with other parties because much effort was needed to create the contents. At the least, it should be guaranteed that each Content Element contains the name of the authors. Another problem of sharing contents was the copyrights on some contents which were owned by other parties. Therefore, the participant stated that the system must provide a password authentication in order to restrict the access to such items. This was formulated as a general requirement and not as a critique of the dLCMS which provided password authentication at that time.

6.3.3 Author 3

The main motivation for Author 3 to participate in this study was the development of an online tutorial. The tutorial would teach the use of the web content management system “Silva”, which was being implemented at the university. In the past, she had organized several classroom trainings on this topic. Now she wanted to make an online tutorial available for autonomous learners who were unable or did not want to attend a classroom course. As a benefit of participation in the study, she hoped for support to create her online tutorial. She further expected to become acquainted with the work of the other pilot projects as examples of e-learning.

From the participant's point of view, conducting the proposed *Learning Unit Analysis* and the development of a *Learning Unit Concept* was a waste of time. She had already worked out a general concept mentally. She wanted to work with the dLCMS first and explore the possibilities of the system. She preferred "learning by doing". However, she took a look at the *Learning Unit Analysis* and *Learning Unit Concept* guidelines and rated them as helpful.

She also did not want to take part in an *Analysis and Concept Workshop* but she did agree to be interviewed, so as to allow the study to understand her plans about the learning unit to be created. The plan for the duration of the interview was for one hour. To maintain the participant's confidence in the evaluation team, care was taken to ensure that the interview did not appear to be a hidden workshop. Therefore, the Learning Unit Analysis and Learning Unit Concept data was not gathered systematically following the structure of the guidelines. The results presented as *Learning Unit Analysis* and *Learning Unit Concept* were based on this interview.

For the same reasons, the participant did not wish to participate in the *Chunking and Development Workshop.* It was decided that she would develop her learning unit autonomously, and that she would write down any problems she encountered. The coach offered her e-mail support for this phase. She agreed to this procedure and to the interview at the end of the development process. The results presented as *Content Chunking* and *Content Development* were based on the participant's support requests via e-mail, her notes of problems, and the concluding interview.

Summary of Results

Content Chunking

The participant did not follow the chunking guidelines. She divided the contents into Content Elements using her own method. This method could not be investigated in more detail because no Chunking and Development Workshop was conducted in this case. She stated that the subject matter was well-suited for modularization as well as for e-learning.

She stated that modularization would help to create contents of which learners could easily perceive the logical structure. Therefore, modularization would pay-off.

Content Development

Most of the Content Elements created by the participant contained only a single multimedia element. These multimedia elements were designed as animated screenshots with balloons commenting on the changes of the user interface between two versions of the Silva software (see Figure 6.3). Only one Content Element used structured markup intensively. It was copied from an existing source. Therefore, the participant did not have to deal with structured markup very much and no detailed results were obtained concerning missing elements of the structured markup schema or of whether the participant perceived structured markup as an aid or as constraining creativity. In any event, she was able to create learning content which corresponded to her expectations.

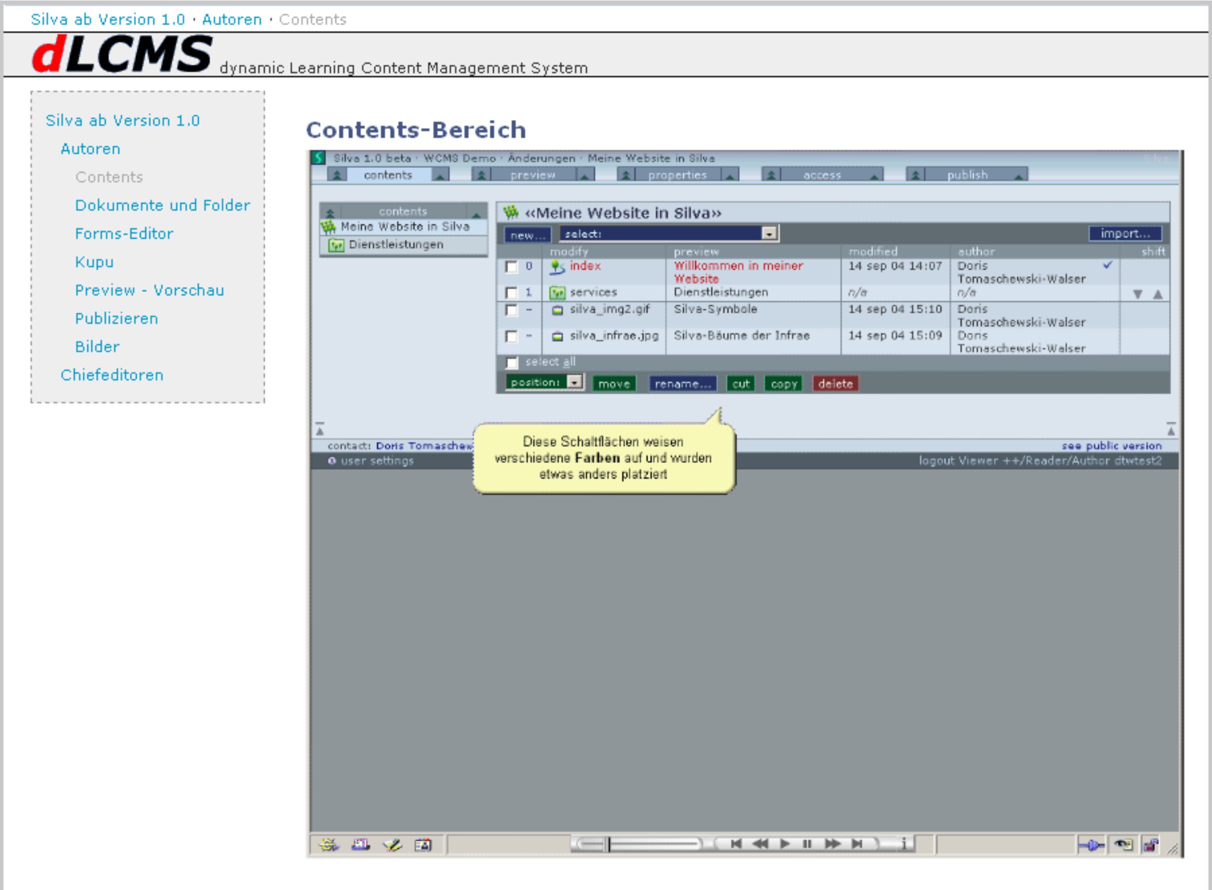

Figure 6.3 A Content Element developed by author 3. The Content Element shows an animated screenshot using balloons to point out what is new in the contents view of a Silva 1.0 Folder.

Learning Unit Assembly

The Content Elements created could be aggregated into a larger learning unit. To provide an overview of the contents in the Learning Unit, the participant wanted to use automatically generated tables of contents (AutoTOC). Although the dLCMS was not designed to provide AutoTOCs, the coach showed her a work-around to create them anyway.

Analysis of the Final Content Elements

In the end, the participant created thirteen Content Elements. Four elements represented tables of contents (TOC) with links to the other Content Elements. They provided an overview of the Learning Unit and contained links to the other Content Elements. By its nature, a TOC references other elements and is therefore neither self-contained nor relating to a single objective. Therefore, the TOC Content Elements were not analyzed any further. However, these elements show the need for an element type which presents an overview of the Learning Unit and its sub-sections.

Of the remaining nine Content Elements, eight were self-contained. The one element not rated to be self-contained referred to a resource on the web containing multimedia contents.

Eight Content Elements represented a single didactic content type. They were either expository, specifying the changes of a particular user interface view, or instructions, teaching new operating procedures. One Content Element combined expository contents and instructions and therefore, did not represent a single didactic content type.

Five Content Elements were clearly related to a single objective, three presented two clearly distinguishable procedures. One Content Element contained eleven clearly distinguishable sub-tasks of the general editing procedure. It had originally been created by other authors and was copied as a whole into the dLCMS. All elements relating to a single objective also represented a single didactic content type.

6.3.4 Author 4

The Author 4 was a developer of the dLCMS and the author of this work. His main motivation was to gain experiences throughout the entire development cycle of learning contents, from the Learning Unit Analysis to teaching students. He hoped to gain a better understanding of the advantages and the problems of the system and its concepts.

The participant was a research assistant working the field of man machine interaction. He decided to develop a learning unit on the subject of “Usability Evaluation”. The learning unit was to be applied for post-graduate students in the field of ergonomics.

Summary of Results

Content Chunking

The participant followed the chunking guidelines to modularize his contents. In the first step, *breaking contents down into single objectives*, he was not sure which content categories should be used to categorize an overview of usability methods or usability principle definitions. During the second step, *specifying the didactic content types*, he added a new didactic content type category “overview”. In the third step, *analysis of other potential learner groups*, the participant identified four other potential learner groups. However, without having detailed learning objectives for these groups, he reported that he was not able to analyze which contents would be reusable for these groups.

The participant rated the subject matter to be suited for modularization, stating that the topic could easily be divided into concepts and principles. Nevertheless, he judged the subject matter to be only partly suited for e-learning. He stated that human behavior in the field of human computer interaction cannot be easily modeled and simulated and therefore, it is difficult to develop interactive online learning activities.

Content Development

He was able to create a Learning Unit which corresponded to his expectations but he "misused" some of the markup elements to set off definitions, examples, and bibliographic information. Furthermore, he was not sure how to set off titles in the top row of tables and how to markup table captions. The dLCMS questionnaire content type did not match his expectations. Therefore, he integrated multiple-choice self-assessments using "raw" HTML and JavaScript code, which he considered to be a hack. Apart from this, he did not perceive the structured markup schema as constraining creativity.

Learning Unit Assembly

The participant started to assemble the Learning Unit in parallel to the development of the Content Elements. In order to see the evolving lesson, he frequently switched back and forth between the repository, where Content Elements were edited, and the lesson view, where he immediately added the new Content Elements to the Learning Unit. After the Content Chunking procedure, the participant estimated that some Content Elements based on didactic content types were too small (see Figure 6.4). Therefore, he added explanations or examples to these elements. He thought it should be possible to combine several Content Elements onto one page. Generally, he stated that modularization did not change the way he was thinking about the learning unit but he added that using modular contents resulted in a very structured learning unit. For him, modularization did pay-off.

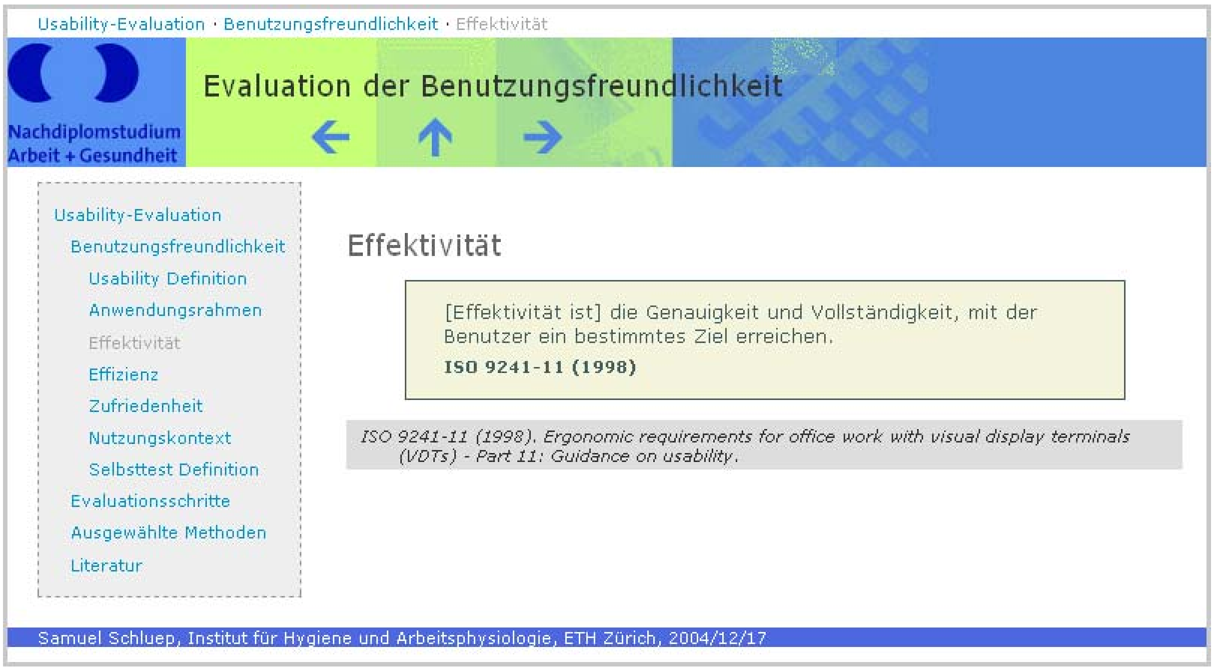

Figure 6.4 The first version of a Content Element of Author 4 containing only a single definition. The participant thought that the page should present more information than just the definition.

Analysis of the Final Content Elements

In the end, the participant created 58 Content Elements.

Two Content Elements were created as an overview and thus referred to Contents Elements contained in the Learning Unit. These were the only elements which were not rated to be self-contained by the final Content Element analysis. For these elements, none of the didactic content types or content categories seemed to be appropriate. Therefore, the category “overview” was assigned as a new didactic content type as well as a new content category.

Eighteen Content Elements did not represent a single didactic content type. Most of these elements contained a definition and either an explanation or an example illustrating the definitions. One Content Element contained a very short expository text and an example.

Fifty-three Content Elements related to a single objective. The Content Elements not rated as relating to a single objective were the two overview elements described above – two elements which contained a list of concepts or principles, and the last element containing the bibliography for the whole Learning Unit.

Thirty-five Content Elements fulfilled both criteria: representing a single didactic content type and being related to a single objective.

6.4 Discussion

This section discusses the results obtained with respect to each of the research hypotheses formulated in Section 6.1. The discussion is grouped into three parts: *content chunking*, *modularization*, and *structured markup*.

Content Chunking

H1.1: Authors are able to identify single objectives by assigning the content categories, “concept”, “fact”, “procedure”, “process”, or “principle”, to the contents.

The findings do not support this hypothesis. The main problem concerned the understanding of the *content categories* “concept”, “fact”, “procedure”, “process”, or “principle” and their applicability to the contents available.

Author 2 was not able to assign the given set of content categories to her contents at all. Author 1 and Author 4 had difficulties applying the content categories to the contents given – for some contents none the categories given seemed to fit.

As an explanation, one might argue that the given set of content categories is too small to contain sufficient items to suit all contents. In the Cisco content model, next to the content categories on the RIO level, two additional categories can be found on the RLO level: “overview” and “summary” (Barritt et al. 2000, see Section 3.4). The classification of “Information Types” in Horn’s Structured Writing methodology contains the two additional categories “structure” and “classification” (Horn 1993, see Section 3.3). However, extending the classification scheme will not help to better understand the previously given categories.

The theories and guidelines of Cisco, Horn, and Merrill (Merrill 1983; 1987, see Section 3.3) use classifications, which are related to the content categories used here, to analyze the learning objectives on an abstract level. Based on this analysis, these methodologies will then specify the types of Content Items (Cisco), Information Blocks (Horn) or presentation forms (Merrill) to be combined to create optimal understandable learning contents. Therefore, it may be argued that the use of content categories will help to modularize the contents if a learning unit is designed from scratch, but that it is problematic to assign content categories to already existing contents, as Author 1 and Author 2 did. This contradicts the experiences of Author 4, where the learning unit was created from scratch but where problems using the content categories occurred as well.

We may conclude that the content categories are rather abstract and difficult to understand. Authors certainly need better support to use this classification if it should be used at all.

H1.2: Authors are able to identify single didactic content types.

The findings suggest that this hypothesis can be supported. Generally, the *didactic content types* were easily understood by the participants.

Although Author 1 and Author 2 did not understand all categories at first sight, this may be interpreted as a language problem. The didactic content types were presented in English while the participants’ native language was German. The term “advanced organizer” is perhaps a special case. It is a specialized notion of didactics and might need more profound knowledge of didactics to be understood than the knowledge that the participants had. Generally, the guidelines should contain a better explanation of the didactic content type classification.

The experiences of Author 4, where a new category “overview” was added, indicates that further research is needed to provide a sufficient and comprehensible set of didactic content type categories.

H1.3: Authors are able to analyze the contents for reuse with other potential learner groups.

The findings do not support this hypothesis. In all three cases where the Content Chunking guidelines were applied, the analysis of the contents for reuse with other potential learner groups did not have any effect on the final modularization structure.

The difficulties in analyzing content for other potential learner groups encountered were:

- The subject matter was very specialized and could only be used for a very limited audience of subject matter specialists. No learner group other than the original target group could be identified (Author 1).
- The subject matter can be described as "general education". It did not intend to teach skills and knowledge needed for specific professions, and therefore, all contents are useful for all learner groups identified (Author 2).
- Detailed learning objectives for the other groups identified were missing. Without such information it is difficult or even impossible to determine what content is important for any other learner group (Author 4).

Furthermore, it may also be argued that the analysis for reuse with other potential groups does not make much sense if it is applied to Content Elements which have been modularized to the *didactic content type* granularity level. This level is already very finely grained.

We may conclude that the *analysis for reuse with other learner groups*, at least on the didactic content type level, is not useful.

Modularization

H2.1: Authors can create Content Elements which are self-contained (self-contained Content Elements have no references to other units but may have prerequisites to the learners' knowledge).

The findings support this hypothesis. In all cases, most of the Content Elements created were rated to be self-contained by the analysis of the final Content Elements.

Anyhow, there is a need for elements which are not self-contained in order to present an overview of a lesson. This can be seen as a way to provide a learning context which goes beyond simple aggregation. Author 4 created two such overviews that explicitly referred to the contents contained in the Learning Unit. Author 3 used simple tables of contents to provide such an overview.

Another type of Content Element, which was not considered to be self-contained, concerns lists of reference information, such as bibliographic information and references to external web-sites containing further material. These Content Elements contained links to other resources. Although they were not rated to be self-contained, it is a characteristic of such information to provide links to other resources. Therefore, the existence of such Content Elements does not mean that authors are not able to create self-contained Content Elements.

H2.2: Authors can create Content Elements which represent a single didactic content type.

This hypothesis cannot be rejected but the results of the Content Element analysis suggest that authors need better support to create Content Elements which represent a single didactic content type. Figure 6.5 shows a summary of the results of the Content Element analysis. The rate of Content Elements which represent a didactic content type for each author was between approximately 30% and 90%.

In two cases, the Content Elements were simply created by copying existing contents (Author 1 and Author 2). The results of the Content Chunking procedure, as described in the Learning Unit Development Guidelines, were not applied to the implementation of the final Content Elements. Therefore, the approach to create Content Elements based on didactic content types was not consequently followed.

Another problem is the small size of the materials which represent a single didactic content type. For example, definitions often consist only of one or two sentences. The dLCMS Component Model specifies Content Elements to be presented as a single web-page. However, the authors perceive pages containing only a few sentences as being too small. In their opinion, it should be possible to combine several Content Elements representing a didactic content type onto one page; for example, by adding an example or an explanation next to a definition (Author 2 and Author 4). This might indicate that the dLCMS Component Model must be re-thought in order to allow for the aggregation of several Content Elements onto a single page.

During the analysis of the Content Elements, we found that the list of didactic content type categories needs to be extended. Specifically, the categories “overview” and “introduction” seemed to be missing. Furthermore, the type “expository” is not very specific and can easily be applied to different kinds of contents and granularity levels. It can be applied to small pieces of contents as well as large ones that cover several objectives (see Author 3, where more Content Elements are representing a single didactic content type than are relating to a single objective). These findings suggest that further research is needed to develop a sufficient and comprehensible set of didactic content type categories.

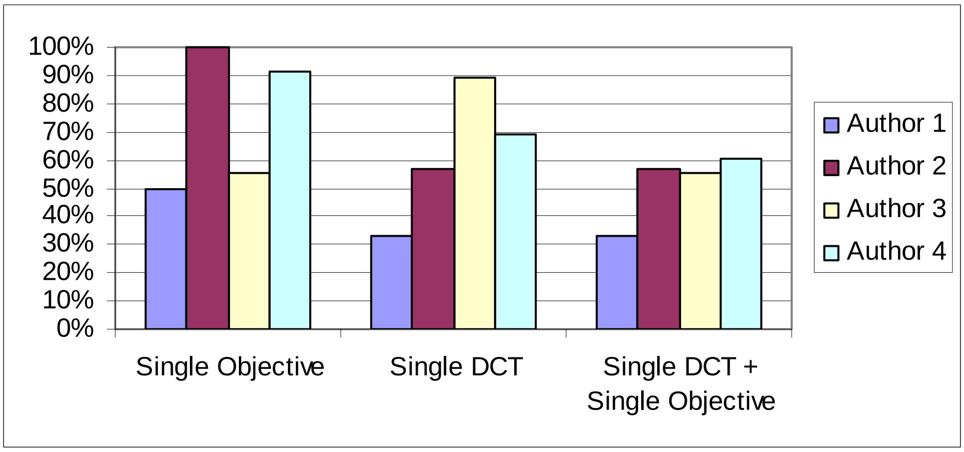

Figure 6.5 A summary of the results of the final Content Elements analysis. It presents the percentage of Content Elements of each author which (1) relate to a single objective, (2) represent a single didactic content type (DCT), and (3) fulfills both criteria.

H2.3: Authors can create Content Elements which relate to a single objective (a fact, a concept, a principle, a procedure, a process).

The results of the final Content Element analysis suggest that this hypothesis may be upheld but more research is needed to understand how this can have an impact for the dLCMS and content creation. At least half of the Content Elements of each author were found to be related to a single objective and could be classified as either a fact, a concept, a principle, a procedure, a process (see Figure 6.5).

The problems encountered by Author 1 in creating Content Elements that are related to a single objective might be due to the fact that the participant only wanted to transfer the existing contents into the dLCMS as they were. She did not consider revising the contents according to the results of Content Chunking procedure. Another problem may be caused by the fact that the participant, as she was not a subject matter expert, had to edit web-based contents based on original material which did not contain clear structure and she did not have enough background knowledge to redefine the structure.

Author 3 did not apply the guidelines. Nevertheless, four of seven Content Elements which the participant created from scratch related to a single objective. One further element was clearly divided into two parts, each part having its own title and relating to a single objective. The remaining two Content Elements that she had designed on her own were related to two objectives. This suggests that the participant had an intuitive understanding of the modularization concept.

Generally, there seems to be a need to create Content Elements which consist of an overview. These elements can either be an overview of the topics presented in the Learning Unit or contain a list of terms which may or may not have a short definition.

The results of the evaluation also suggest that better support for content authors is needed and that the Learning Unit Development Guidelines should be improved.

H3.1: Authors can aggregate Content Elements into larger Learning Units.

The findings support this hypothesis. All participants were able to aggregate the Content Elements that they had created into a Learning Unit. This is not very surprising because the Content Elements were designed for the learning unit to be created. It would be interesting to see if users will still be able to aggregate Content Elements when they start to reuse contents from different sources.

One problem identified concerned the overall Learning Unit title. The system automatically used the title of the Content Element associated to the Learning Unit top node as the overall title. The findings suggest that there is a need to specify a separate title.

An important observation made was the development of Content Elements and their aggregation into a Learning Unit done in parallel (Author 4). Therefore, the Content Elements were specifically designed for the context of the learning unit.

Nevertheless, the Content Elements were rated to be self-contained and related to a single objective. Still, further research will have to show if these elements will be reusable in other instructional contexts.

H3.2: Modularization influences the didactic quality of the final learning unit from an author's point of view.

The results suggest that modularization has a positive effect on the didactic quality of the learning unit. All participants stated that the content chunking process would yield well structured learning units, providing a clear overview of the subject matter to the learners. Modularization would help to define the learning steps that the learner could grasp immediately and to identify the logical sequence of these steps (Author 3). The analysis of the contents using the content categories and didactic content type categories inspired one participant to structure the contents more clearly (Author 1).

This result was somewhat surprising because the didactic usefulness of learning objects is discussed controversially in the literature. Some authors doubt if a learning context can be established solely by aggregating self-contained chunks of information to a whole (Lambe 2002).

The Content Elements created for the present study were all designed for a specific learning unit. Thus, it is not so surprising that the aggregated Content Elements provided coherent learning context from the authors' points of view. Further research will have to show if it is possible to aggregate Content Elements from various sources for different instructional contexts that provide coherent learning experience from an author's, as well as a student's point of view.

It may be argued that the authors' positive rating of the didactic quality is not an effect of modularization (i.e. an effect of the development of small self-contained Content Elements) but that the rating might actually be influenced by the fact that a structured learning content development methodology was applied. The application of a design methodology, for modularized or non-modularized contents, will generally have a positive influence on the didactic quality. On the other hand, the need for a systematic analysis of the contents, in order to design small, self-contained Content Elements, fosters the development of contents which are presented as small, logical learning steps. Currently, it is not possible to distinguish between cause and effect. Further research should look into this issue.

H4.1: Modularization pays-off from an author's point of view.

The findings support this hypothesis. All participants stated that modularization would pay-off. Learning contents for any web-based training, i.e. even if the contents are not designed to be reusable, modular Content Elements, need to be divided into single web-pages anyway. A clear methodology that analyzes contents from a didactic perspective was therefore perceived to be helpful.

Obviously, applying the content chunking method to an already existing web-based course was an extra effort. Accordingly, Author 2 stated that it would pay-off provided that it is done from the beginning of an e-learning project.

H5.1: Any subject matter is suited for modularization.

Based on the results of the study, this hypothesis may be upheld. Subject matter from three major scientific domains, natural sciences, social sciences, and engineering sciences, as well as subject matter from the university's ICT service department could successfully be modularized.

Having investigated only four cases, it is clear that this result cannot be generalized. The selection of the pilot projects for this study may be biased because of the subject matter being suited for modularization.

However, we would like to point out that even subject matter from the discipline of history could be modularized. This was not evident at first sight, which is emphasized by the participant's rating of the subject matter to be only 50% suitable for modularization. Interestingly, she did not even change her rating after the successful implementation of the material as small modular Content Elements.

Structured Markup

H6.1: The structured markup schema, based on traditional typographical elements, is sufficient to design Content Elements which are in accordance with the authors' expectations.

In general, the findings suggest that this hypothesis may be upheld. The participants were able to implement most of their contents using the structured markup schema provided to create Content Elements which were in accordance with their expectations. Interestingly, Author 1 was not even aware of using a special set of structured markup. This may indicate that the given markup schema was intuitive.

However, the findings suggest that the schema needs to be improved. In the participants view, markup was missing for references to bibliographic and further learning resources (Author 2 and Author 4), glossary entries (Author 1), and table captions (case 4). These elements are also used in traditional typography, for example, in books. Therefore, a structured markup schema including such elements can still be considered to be based on traditional typography.

There seems to be a need to position text next to images (Author 1 and Author 2). The participants were able to do this using layout tables but the use of layout tables is generally considered bad practice because it violates the principle of separating contents from presentation. Further research is needed to analyze in which situations content authors demand such positioning features and how such features can be provided without violating the principle of separating contents and presentation.

The markup schema provides an element for headings in a table (`RowHeading`), which spans the whole table width, but no standard header is available for headers in table cells. For Author 4, it was not clear how to markup headings for single table columns. In the end, the participant used two methods to markup table column headings: in some cases he used `Emphasis` markup, and in others `SubHeading` was applied to mark up table column headers.

Author 2 reported that inline images were missing. She intended to use these elements as icons to visualize different types of references. Icons may be considered a graphical design element. Therefore, it may be discussed whether or not the markup schema should provide inline images. To markup different types of links, a “reference type” attribute might be a more appropriate solution that is in accordance with the principle of separating contents and navigation.

Author 4 was missing special markup for definitions and examples. The definitions and examples were combined onto a single page. If several Content Elements could be combined onto a single page, as the participant suggested, no further markup elements would be needed. The corresponding didactic content type, e.g. “example” or “definition”, would be assigned to the Content Elements using metadata. Such metadata could also be used to set off special Content Elements graphically.

The “expert comment” of Author 1 was a summary of the points a student should remember. They were presented in the original system using synthesized speech but the participant stated that it would be sufficient to paste the text into the Content Element. We may therefore conclude that the structured markup schema can do without this special type of element.

H7.1: Specialized didactic markup can be provided as separate components.

This hypothesis may be upheld. In two cases multiple-choice questions were created as separate Content Elements (Author 1 and Author 4). No other specialized didactic markup was needed by any of the authors. Therefore, we may conclude that special didactic markup can be provided using a separate markup schema.

However, the Questionnaire Content Element provided did not suit the participants’ expectations. This indicates that a more elaborate questionnaire content type is needed. The Questionnaire Content Element markup schema and its implementation did not suit the participants’ expectations for different reasons:

- The labels of the buttons to submit the answer or to show the hint were in English and could not be easily adapted to German.
- The “hint” button was always displayed, even if no text for a hint was specified.
- Author 1 wanted the answers to be ordered randomly.
- Next to single choice and multiple-choice questions, Author 4 wanted to use textual “fill-in-blanks” and “standard short answers”.

Another special element needed by Author 3, although not specifically a didactic one, was an automatically generated table of contents (AutoTOC). This element should provide an overview of the contents of a Learning Unit containing links to the Content Elements. As the participant did not add any information next to the table of contents, such elements may be provided as separate components too.

H8.1: Authors do not perceive structured markup as constraining creativity.

This hypothesis cannot be rejected. No author explicitly stated that structured markup was perceived as constraining creativity. However, three authors were not able to answer the semi-structured interview question IQ 11: “Did the set constrain your creativity?” Further research is needed to provide a data basis to investigate this hypothesis more clearly.

Author 1 was not aware of using a restricted set of markup elements and could not answer question IQ11. However, being unaware of the structured markup may be interpreted to understand that creativity was not constrained.

Author 2 stated that she did not create new contents and therefore no creativity was needed.

Each Content Element that Author 3 had developed from scratch contained only a multimedia element. Therefore, no other markup elements were needed. Still, as a matter of fact, creativity was not constrained in this case – Author 3 created the Content Elements that she wanted to create.

Author 4 did not perceive structured content as constraining creativity but as he was a developer of the dLCMS, we caution that this result should be interpreted with care.

6.5 Conclusions

Content Chunking

RQ1: How can authors be supported to create modularized content?

Evaluating the content chunking method provided by the Learning Unit Development Guidelines, the participants had problems identifying single objectives by assigning content categories “concept”, “fact”, “procedure”, “process”, or “principle” to the contents. Furthermore, the analysis for reuse with other potential learner groups did not have an effect on the content chunking results.

The difficulty to apply the content categories to the contents given may indicate that the definition of Content Elements relating to a single fact, concept, principle, procedure, or process must be re-thought. Either one may want to extend the list of content categories with additional categories, which better suit the available contents. Or, one might consider revising the Content Element definition. For example, Content Elements could be based on the didactic content types only (definition, exercise, self-assessment, etc.). In our opinion, the second solution seems to be more adequate because the content categories seem to be very abstract and difficult to understand. The findings suggest that content authors in an academic environment are more familiar with the didactic content type categorization.

The analysis of the contents for reuse with other potential learner groups as a last step in the Content Chunking procedure does not seem to be useful. Content Elements based on didactic content types are already so small that it does make sense to divide them any further. In addition, some subject matter is either very specialized, thus applicable only to a specific target group, or it can be described as “general education”, applicable as a whole to a wide range of audiences. Identifying which contents suit another learner group has also been found to be difficult without a detailed analysis of the learning objectives for that group.

Modularization

RQ2: Do authors understand the concept of modularization?

In general, the participants were able to create self-contained Content Elements which represented a didactic content type and are related to a single objective.

Due to the nature of some of the information presented, e.g. reference material or an overview of a learning unit, it is clear that not all contents can be self-contained.

The problems encountered to create Content Elements representing didactic content types indicate that better support is needed, i.e. the Learning Unit Development Guidelines need to be improved.

Interestingly, content authors seem to have an intuitive understanding of modularization that relates to the traditional structuring of text into chapters and sections. This is emphasized by one author being able to create self-contained Content Elements without using the guidelines. Further research may want to investigate this intuitive understanding of modularization more thoroughly.

RQ3: Can small, self-contained Content Elements be aggregated into didactically coherent learning units?

Content Elements could be aggregated into larger Learning Units. Furthermore, the authors reported that, from their points of view, the aggregation of modularized contents would result in a well structured learning unit. It remains to be shown that such learning units will also be rated as didactically coherent from the point of view of students (see Chapter 7).

In this study, the modularized contents were specifically designed for specific learning units. It is therefore not surprising that these elements could be assembled into coherent learning units. It will be interesting to see if coherent learning units can also be composed using modularized Content Elements from different sources.

The dLCMS component model may need to be adapted to provide a possibility for combining multiple Content Elements onto a single page.

RQ4: Does modularization pay-off from an author's point of view?

It can be concluded that modularization does pay-off from an author's point of view. Contents for web-based learning need to be divided into single web pages anyway. A clear methodology to do this is perceived as helpful. The structured analysis of the learning contents from a didactic perspective fosters a clear structuring which is beneficial for the didactic quality.

RQ5: Which subject matter is suited for modularization?

Subject matter from different scientific domains, including social sciences and trainings, offered by the administration may be suitable to be successfully modularized. However, further research is needed to generalize these findings.

Structured Markup

RQ6: Is a simple structured markup schema, based on traditional typographical elements, sufficient for learning contents?

It may be concluded that a simple structured markup schema, based on traditional typographical elements, is sufficient for learning contents. However, the schema provided needs to be improved. This includes markup for references to bibliographic and external information, glossary entries, and table captions, as they are elements which are used in traditional typography as well.

The use of structured markup is aimed at the separation of contents and presentation. However, there is a need to specify layout issues, such as positioning text next to images. How to solve this issue without violating the principle of separating contents and presentation remains unclear but we are confident that such a solution can be found.

The problems encountered to set-off definitions or examples visually could be solved by a modified component model. If Content Elements could be combined onto a single page then such elements may be specified as separate components. The type of content would then be assigned using metadata. However, creating smaller components may cause an extra effort for the content authors, who would have to edit an increasing number of separate components. The potential acceptance of such a solution needs to be evaluated.

RQ7: What kinds of specialized didactic markup are needed and can they be provided through separate markup schemas for separated components?

Generally, the approach to provide a separate markup schema for specialized didactic functionality, in order to keep the basic structured markup schema simple and flexible, is practicable. The didactically specialized types of contents used in this study consisted of questions and tests and automatically generated tables of content. In all cases these contents were implemented as separate Content Elements.

Using an approach to integrate several Content Elements onto a single page, as discussed above, pure learning contents and interactive questions can be arranged right next to each other.

RQ8: Do authors perceive structured markup as an aid or as constraint to creativity?

Although the data provided by this study does not give a clear answer to this question, we can at least say that none of the participants explicitly perceived the structured markup as a constraint to creativity.

Summary

The results of the study suggest that the proposed steps to modularize content described in the guidelines did not work well. The assignment of content categories was difficult and the analysis for other potential learner groups did not have any effect on the modularization structure. Regardless, the participants reported that the guidelines would improve the didactic quality of the learning unit, having a structuring effect on the planning of the learning unit and the singular elements.

Generally, the participants were able to create modular, self-standing Content Elements, suggesting that they did understand the concept of modularization. These Content Elements could be aggregated into larger Learning Units which corresponded with the authors' expectations. In a few cases, the participants stated that it should be possible to combine several Content Elements on a single page.

Markup elements reported as missing concerned mainly specialized markup for literature and glossaries. Furthermore, the markup schema provided for multiple-choice questions was not satisfying. No author perceived the structured markup as constraining creativity.

7 Student Evaluation of a Learning Unit

This chapter presents a student evaluation of a learning unit created in the pilot project study described above. It investigated whether students perceived learning units that are based on modular Content Elements as didactically coherent. The learning unit was an introduction to usability evaluation. The learning unit was used to teach students of a post-graduate study in ergonomics. A questionnaire, containing 17 items on the previous computer and e-learning experience and on the didactic quality of the learning unit, was handed out to the students after they had worked with the learning unit. The results of the investigation were analyzed using descriptive statistics. As a result, the students were able to easily detect the logical relationship between the pages. Therefore, it may be concluded that it is possible to aggregate self-contained Content Elements into a larger coherent learning unit. The results further suggest that it is possible to offer good didactic quality provided that such a learning unit makes use of the advantages that e-learning offers, e.g. the use of multimedia and elaborate interactive elements, and the potential to learn “anytime” and “anywhere”. Further, modularized contents may improve the comprehensibility of the contents and yield a clear structuring of the subject matter. As the investigation looked at only one learning unit, which was specially developed for this instructional context by a single author for a specific target learner group, further research is needed to generalize these findings.

7.1 Objectives

In order to investigate the research question *RQ3*, “*Can small, self-contained Content Elements be aggregated into didactically coherent learning units?*” (see Section 6.1), this chapter presents a study which examined a learning unit from a student’s perspective. Therefore, the hypothesis was formulated as follows:

H3.3: Students perceive learning units, which are based on modular Content Elements, as didactically coherent.

A learning unit on “Usability Evaluation”, which was created by Author 4 in the pilot project study (see Chapter 6), was deployed to a LMS. The learning unit was then used for the regular education of ergonomists. The didactic quality and the coherence of the learning unit were evaluated by the students using a questionnaire with 16 items.

7.2 Method

Fourteen students of a post-graduate study in ergonomics worked with the learning unit created by Author 4 (see Section 6.3.4).

The students were between 27 and 51 years old and the mean age was 40. The group consisted of 12 women and two men. They work in the fields of occupational medicine (6), occupational hygiene (2), ergonomics (1), consulting (1), administration (1), and sports (1). The fields of work of two students are unknown.

The subject of the learning unit was an introduction to usability evaluation and it was used to teach the basics. The learning unit contained 58 pages which were structured into four main sections, “Usability”, “Steps of an Evaluation”, “Selected Methods”, and “Literature”. The “Steps of an Evaluation” section contained five sub-sections and the “Selected Methods” section contained two sub-sections. The learning unit had seven multiple-choice type self-assessments and contained 17 illustrations. The learning unit was written in German.

The learning unit was packaged using the IMS Content Packaging Format and was imported into the OLAT[1] learning management system of the University of Zürich. It was then embedded into an OLAT-course, which consisted of a general entry page, an inscription page, and the content package. The navigation tree on the left side of the screen was generated by the OLAT system and the “back”, “top”, and “next” buttons were integrated into the learning unit package. The navigation of the content package was not embedded into the overall OLAT navigation.

Because e-learning was applied for the first time for this group of students, the students were taken to a computer room and they were introduced to the e-learning system. The students had 90 minutes of time to work through the learning unit individually in the computer room. A lecturer was present to provide support when needed. After working through the web-based learning unit, the students performed a simple usability test in two groups, as a practical training in the classroom.

A questionnaire was handed out to the students immediately after the practical training in the classroom. The questionnaire contained four questions concerning the student’s age, profession, previous computer and e-learning experience, and 12 closed-ended questions on the didactic quality of the learning unit. The detailed questions and answers are presented below in more detail. The questionnaire contained a field for further comments at the end. The questionnaire was written in German.

1 The OLAT (Online Learning and Training) system is an open-source learning management system (LMS) developed by the University of Zürich.

The results of the investigation were analyzed using descriptive statistics.

Student's Profession

In what field do you work?

Possible answers were "occupational medicine", "occupational hygiene", "ergonomics", and "others". The questionnaire provided a field to specify a more detailed answer for "others".

Student's Previous Computer Experience

How do you judge your computer experience?

Possible answers were:

- No or almost no experiences with computers.
- I rarely use computers.
- I use the computer regularly for office tasks, e-mail and internet.
- I have already designed web-pages and developed small programs.

Student's Previous E-Learning Experience

Have you had experiences with e-learning applications previously?

Possible answers were:

- No previous experience.
- I did take a look at e-learning applications but never worked with one.
- I have worked once or twice with an e-learning application.
- I have already worked with e-learning applications often.
- I have already developed e-learning applications.

Satisfaction

Did you enjoy working with the e-learning application?

The possible answers were "no, not at all", "mostly no", "rather no", "rather yes", "mostly yes", and "yes, very much".

Interesting Contents

Were the contents of the e-learning application interesting to you?

The possible answers were "no, not at all", "mostly no", "rather no", "rather yes", "mostly yes", and "yes, very much".

Comprehensibility

How do you judge the comprehensibility of the e-learning application?

The possible answers were "unusable", "insufficient", "sufficient", "satisfactory", "good", and "very good".

Self-Assessments

How do you judge the self-assessments of the e-learning application?

The possible answers were "unusable", "insufficient", "sufficient", "satisfactory", "good", and "very good".

Graphical design

How do you judge the graphical design of the e-learning application?

The possible answers were "unusable", "insufficient", "sufficient", "satisfactory", "good", and "very good".

Navigation

How do you judge the navigation of the e-learning application?

The possible answers were "unusable", "insufficient", "sufficient", "satisfactory", "good", and "very good".

Didactic Structuring

How do you judge the didactic structuring of the e-learning application?

The possible answers were "unusable", "insufficient", "sufficient", "satisfactory", "good", and "very good".

Logical Relationship of Pages

In your opinion, is there a clear logical relationship between the individual pages?

Possible answers were (the notions in brackets will be used as a short-cut for the answers in the results section – they were not included in the questionnaire):

- The relationship was clear at first sight [clear at first sight].
- For some pages the relationship was not clear [mostly clear].
- For many pages the relationship was not clear [unclear for many pages].
- The relationship was clear only for a few pages [mostly unclear].
- The relationship was unclear – the learning unit seemed to be a conglomerate of pages [totally unclear].

Overall Length of the Learning Unit

> *How do you judge the length of the e-learning application compared to the time available?*

The possible answers were “too long”, “rather too long”, “just right”, “rather too short”, and “too short”.

Length of the Individual Pages

> *How do you judge the length of the individual pages?*

The possible answers were “too long”, “rather too long”, “just right”, “rather too short”, and “too short”.

Number of Interactive Pages

> *How do you judge the number of interactive pages (self-assessments) in the e-learning application?*

The possible answers were “too many”, “rather too many”, “just right”, “rather too few”, and ”too few”.

Number of Illustrations

> *How do you judge the number of illustrations in the e-learning application?*

The possible answers were “too many”, “rather too many”, “just right”, “rather too few”, and ”too few”.

7.3 Results

Thirteen of the fourteen questionnaires were returned.

Students’ Previous Computer Experiences

Two students answered that they “had already designed web-pages and developed small programs”. The rest of students “use the computer regularly for office tasks, e-mail and internet”.

Students’ Previous E-Learning Experiences

Table 7.1 shows the detailed results. Most students did not have much previous e-learning experience. Only one had often worked with e-learning applications before and one was an e-learning application developer.

Table 7.1 The students' previous e-learning experiences.

Answer	Number of Answers
No previous experience	1
Did take a look at e-learning applications, but never worked with one	5
Had worked once or twice with an e-learning application	5
Had already worked with e-learning applications often	1
Had already developed e-learning applications	1

Satisfaction

Most of the students answered "rather yes" (5) or "mostly yes" (4). One student enjoyed working with the learning unit very much and two students answered "rather no" (see also Figure 7.1). One student did not answer this question.

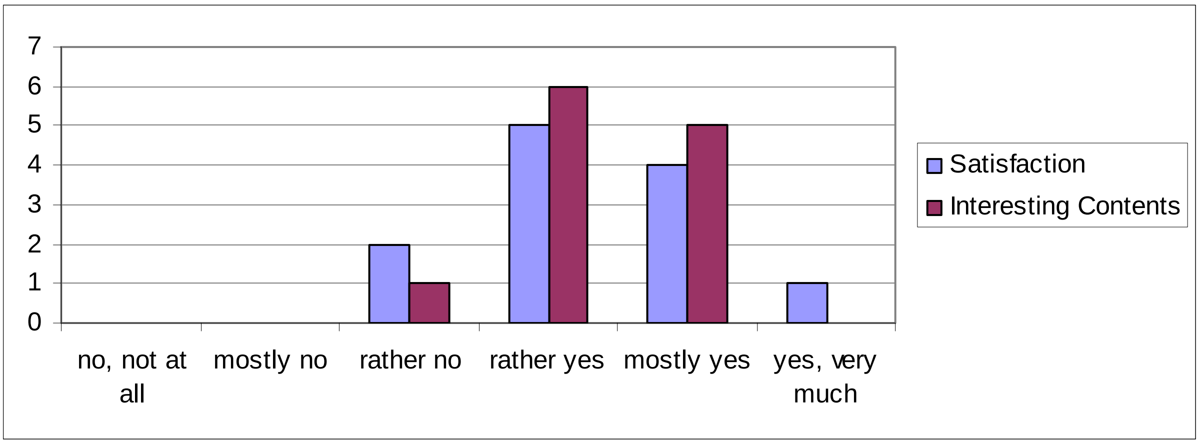

Figure 7.1 Number of answers of the questions concerning "satisfaction" and "interesting contents".

Interesting Contents

Figure 7.1 shows the detailed results of this question. Most of the students answered "rather yes" (6) and "mostly yes" (5). One student rated the contents as rather uninteresting and one student did not answer this question.

Comprehensibility

Most of the students rated the comprehensibility of the contents (7) as "good" and three found it to be "very good". Only two rated the comprehensibility as "satisfactory" or "sufficient" (see Figure 7.2). One student did not answer this question. Generally, the comprehensibility can be rated as good.

Self-Assessments

The quality of the self-assessments was rated as "satisfactory" (5) and "good" (6). Only one student rated it as "very good" and one student did not answer this question (see also Figure 7.2).

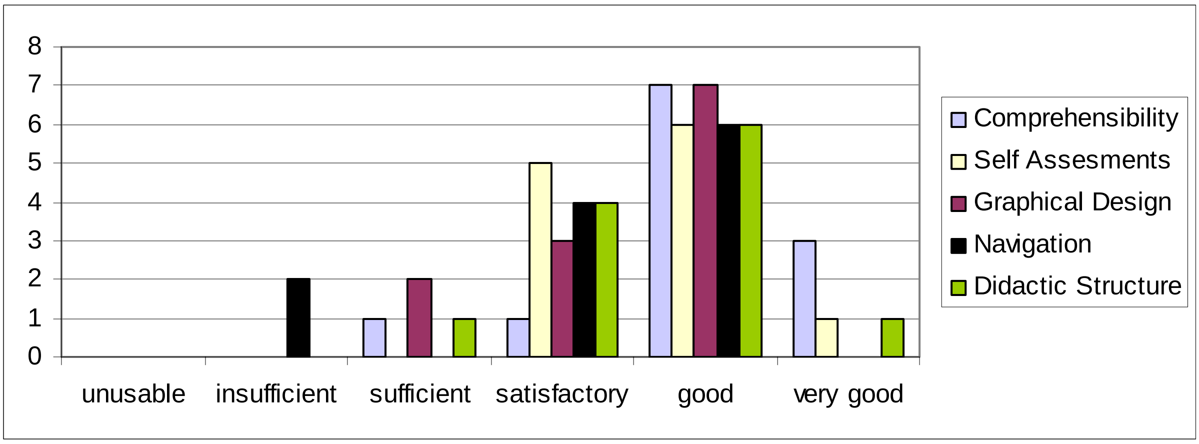

Figure 7.2 Number of answers of the questions concerning “comprehensibility”, “self-assessments”, “graphical design”, ”navigation”, and “didactic structuring”.

Graphical Design

Most of the students rated the graphical design (7) as “good” and three found it to be “satisfactory”. Only two rated the comprehensibility as “sufficient” (see also Figure 7.2). One student did not answer this question.

Navigation

Half of the students rated the navigation as “good” (6), four students found it to be “satisfactory” and two judged it to be “insufficient” (see also Figure 7.2). One student did not answer the question.

Didactic Structuring

Most of the students found the didactic structuring to be “good” (6) and “satisfactory” (4). One student rated it to be “very good” and one as “sufficient” (see also Figure 7.2). One student did not answer this question.

Logical Relationship of Pages

The students found that there was a clear relationship between the pages. Seven students answered that the relationship context was clear at first sight, five answered that for some pages the relationship was not clear (see Figure 7.3). One student did not provide an answer.

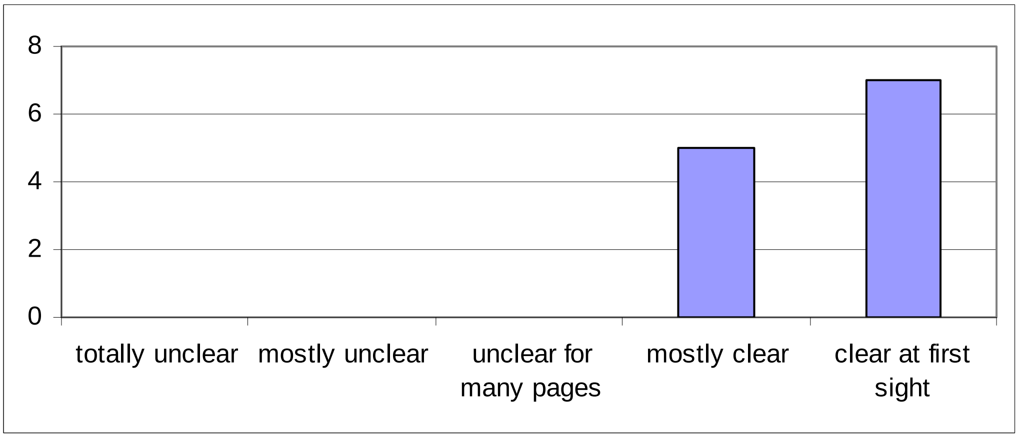

Figure 7.3 Number of answers of the question concerning the "logical relationship of pages".

Overall Length of the Learning Unit

Most of the students rated the overall length of the learning unit to be just right (9). Three answered that it was "rather too long" and one judged it as "rather too short" (see Figure 7.4).

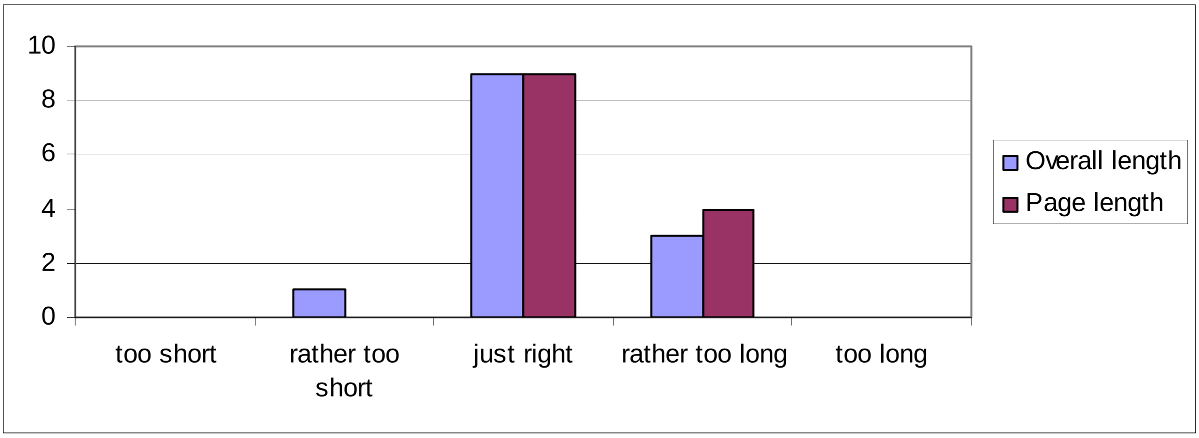

Figure 7.4 Number of answers of the question concerning the "overall length of the learning unit" and the "length of the individual pages".

Length of the Individual Pages

Most of the students rated the length of the individual pages to be just right (9). Four answered that the pages were "rather too long" (see Figure 7.4). The length of the individual pages tended to be too long.

Number of Interactive Pages

Most of students (10) rated the number of interactive pages, i.e. the multiple-choice tests, as "just right". For two students there were "rather too few" and for one there were "rather too many" self-tests (see Figure 7.5).

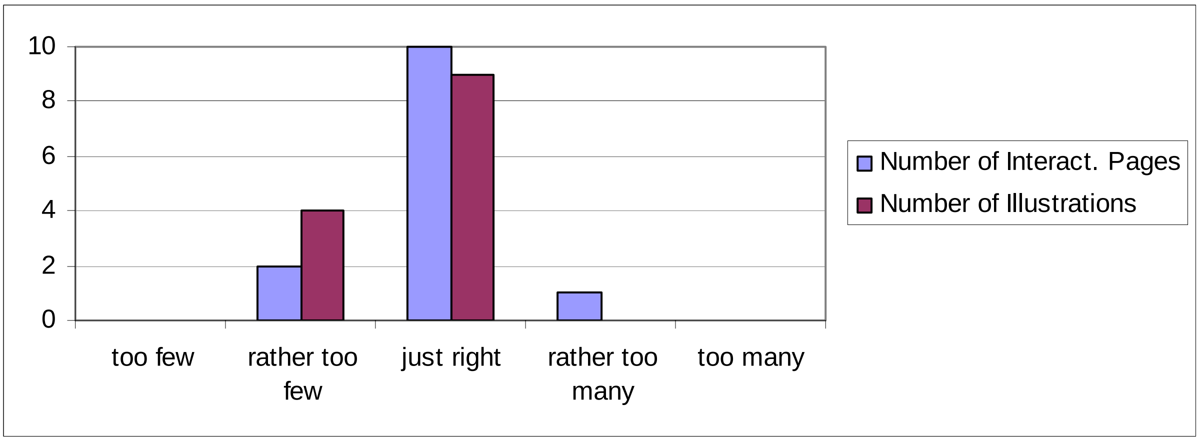

Figure 7.5 Number of answers to the question concerning the "number of interactive pages" and the "number of illustrations".

Number of Illustrations

A large number of students (9) rated the number of illustrations as "just right". For four students, there were "rather too few" illustrations (see Figure 7.5). There was a tendency for too few illustrations.

Further remarks

The following notes were added to the field for further remarks:

- The well defined structure of each page resulted in a high clarity but provided little variety. The examples could have been used to provide more variety.
- One page is easy to read but longer texts are difficult to read if they are not printed on paper.
- In my opinion, the "next" button should be at the bottom of the page (so one does not have to scroll back to the top of the page each time).
- It is cumbersome to scroll the page up and down in order to read all of the contents. The reception of the information is interrupted because the navigation bar has to be clicked. Ultimately, this would be better with a "scrolling mouse".
- The sentences were too long and too complicated. Ultimately, the sentences should be taken apart and made shorter. The text would be clearer and more comprehensible.
- E-learning applications should be used at home; it is a pity to use the time in the classroom for e-learning.
- I would print out the "page" at home, and read it on paper because this is more comfortable for me and provides a better overview.

7.4 Discussion

In general, the overall didactic quality of the learning unit can be rated as “satisfactory” to “good”. The e-learning unit contained much text and provided only simple multiple-choice type questions as interactive elements. No multimedia elements, such as videos, animations or interactive simulations were used because only minimal development resources were available. Apart from the hypertext navigation, multimedia elements would provide a clear added-value as compared to paper-based learning materials. Without these elements, it is not surprising that the learning unit did not obtain higher marks.

The students’ additional remarks underline that they would prefer to read the contents on paper and that the number of illustrations tended to be rather too small. Even though there is no indication that the number of interactive pages was too small, the response may be influenced by the fact that the students were thinking of interactive pages as multiple-choice questions like the ones offered by the learning unit. They may not have wanted more self-assessments of this type, which admittedly were not very interesting. However, they may like to have more complex interactive elements, such as simulations.

Further, the learning unit cannot benefit from the advantages of e-learning, which offers the potential to learn “anywhere” and “anytime”. This is emphasized by some remarks stating that some students would prefer to have a personal interaction with a teacher in classroom courses and use e-learning at home. Since e-learning was applied for the fist time in this case, it was important to provide an introduction to the technical system and to support the students during their first steps. The practical experiences during the introduction of the system have shown that the support was needed and therefore, that the procedure was appropriate.

In the context of this work, it may be pointed out that the comprehensibility of the contents can be deemed to be “good”. This may be interpreted as an effect of the content chunking procedure. The comprehensibility of the contents may benefit from the need to clearly structure the contents and to present the single topics in a concise manner. This may be emphasized by the comment of one of the students, saying that the well defined structure resulted in a high clarity.

Apparently, the students could easily detect the logical relationship between the single pages, even when they were designed as self-contained Content Elements. It may be concluded that is possible to provide a clear learning context by aggregating self-contained Content Elements. However, it should be noted that the learning contents were developed from scratch specifically for this learning unit. Further research is needed to show if self-contained Content Elements coming from different sources can also be combined into coherent learning units.

The length of the individual pages was rated as tending to be rather too long. This may be a sign that students do not have a problem with short and concise Content

Elements but the results indicate that this rating was due to the need to scroll the page up after having read the contents in order to proceed to the next page. The “previous” and “next” buttons were only available in the title bar. The students were working with 15-inch screens and only about half of the height was available for the presentation of the contents. Much vertical space was used by the Windows task bar, the browser frame, the LMS frame and the learning unit title bar (see Figure 7.6). Other results might have been obtained if there had been “previous” and “back” buttons at the bottom of the page. Similarly, using the “Full Screen” browser mode would allow for better use of the available screen area. In the future, we will explicitly suggest to the students to use the browser’s “Full Screen” mode.

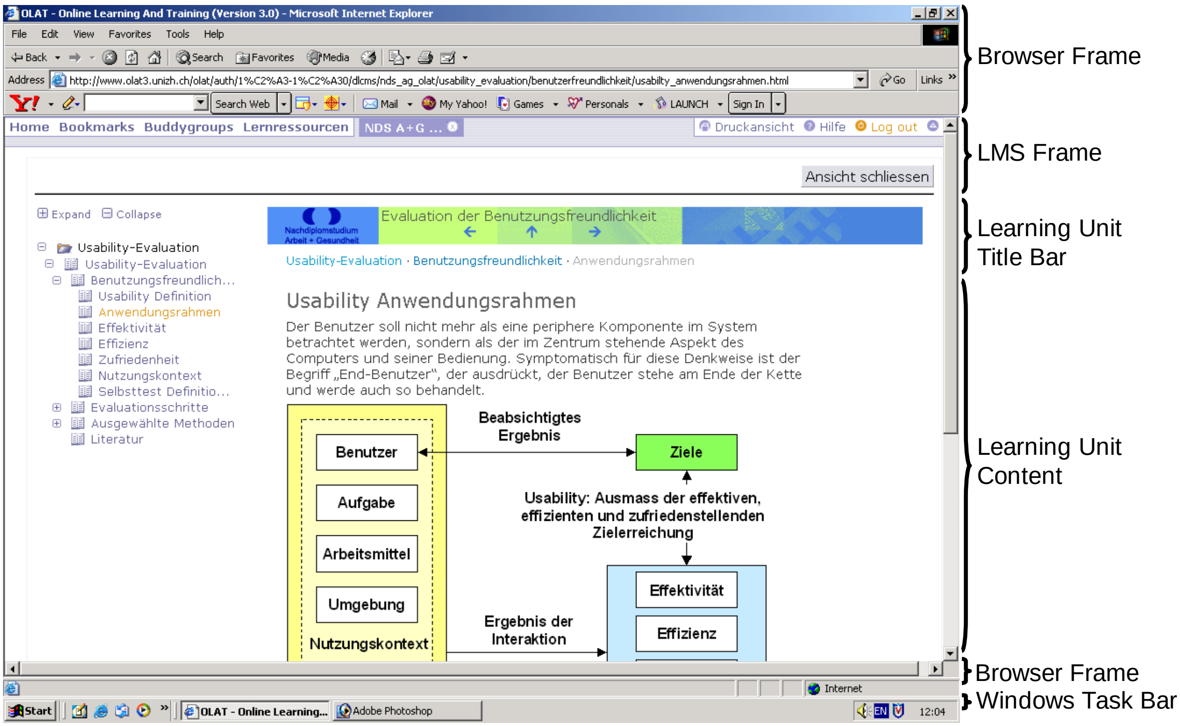

Figure 7.6 The vertical space used by the Windows task bar, browser frame, LMS frame, learning unit title bar on a 15-inch screen. Only about half of the vertical space is available for the learning contents.

Two persons rated the navigation as “insufficient”. This may be due to navigation provided by the OLAT system. Using the navigation tree on the left side (see Figure 7.6), students could only switch to pages inside the content package. In order to return to the entry page of the whole course, the button “*Ansicht schliessen*” (“close view”) in the top right corner had to be clicked. This is admittedly not very intuitive.

7.5 Conclusions

This chapter presented a student evaluation of learning unit created with the dLCMS. The evaluation was aimed at investigating the student’s perception of the didactic quality of a learning unit, which was composed of small, self-contained Content Elements. More precisely, the evaluation should provide an answer to the question of whether students perceive learning units, which are based on modular Content Elements, as didactically coherent.

As the students were able to easily detect the logical relationship between the pages, i.e. the Content Elements, it may be concluded that it is possible to aggregate self-contained Content Elements into a larger coherent learning unit.

The results further suggest that it is possible to provide good didactic quality, provided that such a learning unit makes use of the advantages that e-learning can offer, e.g. the use of multimedia and elaborate interactive elements, and the potential to learn “anytime” and “anywhere”.

Furthermore, modularized contents may yield a good comprehensibility of the contents and a clear structuring of the subject matter. The extra effort for learning content authors seems to pay-off, resulting in a better didactic presentation of the material.

However, the investigation examined only one learning unit, which was specially developed for this educational context by a single author, for a specific target learner group. In order to generalize these findings, more research is needed which includes learning contents from various sources and authors that has been originally designed for different learner audiences.

8 Summary and Conclusions

The concept of *learning objects* has been introduced in the e-learning field to enhance the reusability of learning content. There seems to be a common understanding that learning objects should be relatively small, self-contained units of learning content which can be aggregated into larger learning units and which are reusable in multiple instructional contexts.

Despite the many standardization activities, there are no established specifications for the structure and granularity of learning objects. Today, they come in a variety of data formats. Most of them are individually designed and styled, and navigational and user interface controls are directly integrated into the objects. This prevents the presentation of aggregated learning objects to learners in a coherent way.

As an approach to improve the possibility to aggregate learning objects from different origins, the basic building blocks should have similar granularity and should separate contents from their graphical presentation.

Learning content component models contain different, more or less well defined, levels of granularity and they specify how the objects can be aggregated into higher level units. Therefore, learning content component models allow one to define different levels of granularity more precisely. In our work, *modularization* is understood as the concept of a modular component model as well as the content chunking process to create such modular learning contents.

Structured markup implies an abstraction of the visual presentation from the learning content. Well defined structuring schemes together with appropriate data formats, such as *XML*-based languages, provide the basis for machine processing in order to adapt the learning contents to various presentation needs, thus enabling the separation of content, presentation, and navigation.

As learning content authors are a key factor for the successful application of these concepts, we investigated modularization and structured markup from the point of view of authors in an academic environment.

8.1 Modularization

To investigate learning object component models, a flexible learning content component model has been developed and implemented, which consists of three levels: *Assets* for “raw media”, *Content Elements* as the basic building blocks of learning contents, and *Learning Units* representing aggregated Content Elements.

In our work, we proposed basing the granularity of Content Elements on *didactic content types*, e.g. definitions, examples, exercises, self-assessments, etc., rather than on learning objectives as proposed by other learning content component models. Didactic content types provide a level of granularity which allows teachers to adapt learning units to the needs of students from different disciplines. In general, this approach could be successfully applied. However, Content Elements focusing on didactic content types, for example, a single definition, can be very small. Therefore, one might need to combine several Content Elements onto a single page.

Based on the results of the qualitative evaluation together with learning content authors in an academic environment, it can be concluded that, in general, the authors are able to create modular, self-standing Content Elements. The authors seem to have an intuitive understanding of modularization. However, the findings suggest that the authors in charge of creating web-based learning contents often are neither subject matter nor didactic experts. The task to create didactically sound web-based learning material using existing materials, such as lecture notes and books, is challenging. Therefore, these authors welcome a clear method which supports creating such Content Elements.

We may also conclude that self-standing Content Elements can be aggregated into larger didactic coherent Learning Units from an author's as well a student's point of view. Modularization seems to foster the didactic quality of the learning unit yielding a good comprehensibility of the contents and a clear structuring of the subject matter. It is clear that good web-based learning materials should make use of the advantages that e-learning can offer as it provides an added value to paper-based learning media. Therefore, a learning unit should integrate multimedia and elaborate interactive elements where appropriate, and offer the potential to learn "anytime" and "anywhere". Given these advantages, the extra effort of modularization is perceived to pay-off.

Subject matter from different scientific domains, including social sciences, as well as training offered by the administration could be successfully modularized.

8.2 Content Chunking

In order to support authors to modularize learning contents, the methods which we have found in the literature generally base the granularity of their basic building blocks on a learning objective level.

To support the chunking of learning content into didactic content types, a chunking methodology has been developed which provides a three step procedure. This procedure ensures that the final Content Elements are based on didactic content types, which, combined with other Content Elements, are used to teach single objectives and are reusable for different potential audiences. First, the content is broken down into topics and subtopics, each of which is related to a single fact, concept, principle, procedure, or process. Then, the author assigns the didactic content types that he or she wants to use to every topic or subtopic. Finally, the

Content Elements specified thus far are analyzed for reuse with the potential learner groups.

The content categories (fact, concept, principle, procedure, or process) have been found to be abstract and are therefore difficult to comprehend. As a consequence, authors need better support to use this type of classification if it is to be used at all.

The analysis of the contents for reuse with other potential learner groups as the last step of the Content Chunking procedure does not seem to be useful. Content Elements based on didactic content types are already very small.

However, the findings suggest that content authors in an academic environment are familiar with the didactic content types (definitions, examples, exercises, etc.). As an approach to improve the chunking method provided, one may want to consider using only one step that identifies the didactic content types directly.

8.3 Structured Markup for Learning Content

Different types of markup languages for educational material have been developed and have even become standards. These types include pedagogical modeling languages, structured markup schemas for learning content, question and test markup languages, and other specialized markup languages.

Although many XML-schemas have been developed for the e-learning domain, there is still no structured markup schema for pure learning content in sight that provides a true separation of learning contents and presentation and which may be a true candidate for future standardization.

We have proposed using a simple structured markup schema for learning contents that are based on traditional typographical elements, such as headings, paragraphs, emphasis, etc. Such a schema is anticipated to be familiar to learning content authors and it may be easily translated to potential future standards for learning contents.

In order to keep the basic schema for learning content simple and flexible, the use of a separate structured markup schema for special didactic functionality, such as questions and tests, has been proposed here.

Based on the results of the qualitative evaluation, together with content authors, it can be concluded that the approach taken is feasible. However, the markup schemas provided need further improvement. The basic schema needs to be enhanced by markup for references and glossaries. One problem found relates to the layout of text blocks together with other media elements. Although structured markup should separate contents from presentation, and thereby from layout issues, we found that in some cases authors explicitly want to be able to position text next to images.

The approach to provide a separate markup schema for multiple-choice questions is practicable. Expository contents and multiple-choice type questions were created as separate Content Elements in our study. However, a more elaborate schema for questions and tests is needed and it may be based on the IMS QTI specification (IMS 2002a).

8.4 Further Research

On the path towards a modularization strategy combined with structured markup, which may enhance reusability and interoperability of learning material in the future, our work examined the first steps to create such learning material. The concepts developed were initially evaluated by a limited number of authors. As it was understood to be inductive research in order to generate hypotheses, the feasibility of some of the initial concepts proposed were shown, while other concepts were revealed to be problematic. Areas for further research which became apparent include the following:

- The development and evaluation of a learning content component model, which allows one to combine Content Elements on a single page.
- Research focusing on a better understanding of how content authors structure text into sections normally.
- The development of a sufficient and comprehensible set of *didactic content type* categories.
- The development and evaluation of improved Content Chunking guidelines.
- The development of methods to specify layout issues without violating the principle of separating contents and presentation.

It should be pointed out that all modular learning contents were specially developed for a specific educational context by a single author and for a specific target learner group. In order to generalize the findings, further research is needed, which includes different learning contents from various sources and authors that were originally designed for different learner audiences.

The work presented here provides a fundamental framework for the creation and use of small, modular learning objects, that aims at enhancing the reusability of e-learning contents. Further research will have to show that this goal, the reuse of learning materials in various educational contexts, can be actually achieved.

Appendix A: Guidelines for the Development of Learning Units

Learning objects are a new way of thinking about learning content. Authors of learning resources might need guidance to adapt their thinking about learning material, which traditionally had been whole courses or lecture notes. These Guidelines aim at supporting content authors to divide learning contents into self-contained Content Elements, according to the dLCMS Component Model. In particular, the guidelines should assist the authors to create Content Elements which, as a standard granularity level, represent single didactic content types, e.g. definitions, examples, exercises, self-assessments, etc.

The following procedure is based on classical instructional design theories and literature of traditional lesson planning. It should not impose the strict use of a single instructional design theory for the realization of the planned learning unit. It assumes that the teachers and creators of the learning unit are the experts on the contents to be taught and the appropriate didactics. It is further assumed that they have experience with traditional forms of teaching.

Generally, the following development phases can be distinguished to produce learning units with modular and self-contained Content Elements:

1. *Learning Unit Analysis:* Specification of subject matter, major learning objective and analysis of student characteristics, and technological, personal and financial resources.
2. *Learning Unit Concept:* Specification of detailed learning objectives and subject matter, didactic methods and structure of the e-learning unit.
3. *Content Chunking:* Chunking of learning content into small, modular and reusable Content Elements.
4. *Learning Unit Assembly:* The modularized Content Elements will be assembled to the final Learning Unit.
5. *Content Development:* Development of Content Element and Learning Units with the dLCMS.
6. *Teaching:* Teaching refers to the deployment and actual delivery of the final Learning Unit to students.
7. *Evaluation:* The evaluation may guide all phases in parallel.

The following guidelines provide a detailed description for the first three steps: Learning Unit Analysis, Learning Unit Concept, and Content Chunking.

Although the guidelines suggest using a well defined sequence of phases, content authors shall be encouraged to use their familiar approach when defining learning units and to change the order of steps according to their usual procedures. For example, sometimes it may be easier to analyze the content first and define the learning objectives afterwards. In other cases one wants to jump back and forth between some fields. In any case, it is strongly recommended that one deal with all the steps of these guidelines.

In conjunction with these guidelines, a set of forms is provided as an aid for working out the specification for the learning unit. The numbering of items in the forms corresponds to the numbering of the steps listed below.

1 Learning Unit Analysis

The analysis phase is important for the specification of a learning unit adapted to the learners' needs as well as for making the project realizable. It generally includes an analysis of the subject to be taught, the learner profile and the organizational environment.

1.1 Subject Matter

1.1.1 Subject matter to be taught

Here the main title of the subject matter to be taught should be specified.

> Example: Introduction to Usability Evaluation

1.1.2 Major learning objective of e-learning unit, coverage of contents (depth / breadth)

The major learning objective describes generally what the learners should know after working through the learning unit.

> Example: After the learning unit, the students should be able to conduct a simple usability evaluation of a system for a specified user and task.

1.1.3 Integration of e-learning unit into regular class

How will the learning unit be used? Is it a stand-alone course or is it related to some classroom teaching? If it is related to class room teaching, the relationship should be described in more detail. For example, the e-learning unit may be used to teach the basics, while practical examples will be covered and discussed in the classroom. Another possibility would be to teach the basics as a lecture and use the e-learning unit for exercises.

> Example: The e-learning unit will teach the basics of usability evaluation. Selected examples and their discussion will be treated in the classroom and an example of a usability test will be practiced in groups in the classroom.

1.1.4 Final qualification for learner (certificate, credit points etc.)

What are the final qualifications for learners attending to the learning unit? Will they get a certificate or credit points? Will the learning unit be a requirement to get credit points together with some other lessons?

> Example: The learning unit is part of a two day course of which 2/3 have to be attended in order to earn a credit point.

1.1.5 Duration of e-learning unit (how long will the target group be expected to learn?)

The duration specified in the curriculum is important in order to determine the size of the learning unit. The student should be able to complete the learning unit within its time limits.

> Example: Duration 45 minutes.

1.1.6 Is the subject matter suitable for e-learning

For certain subject matters, e-learning is not suitable. The limits of e-learning are reached if:

1. Learning relies on the quality and directness of the process of the discourse (e.g. consulting, communication training).
2. Facial expressions, gestures, sound of voice and emotions play an important role and can only be transmitted insufficiently with digital audio and video (e.g. therapy, consulting, communication training, teaching of foreign languages).
3. Manual and other practical skills are important for learning (e.g. medicine, arts, chemistry).
4. A direct view of objects is important, and cannot be transmitted sufficiently using multimedia (e.g. medicine, biology, physics).
5. Costly multimedia teaching applications are needed, the financing and technical realization of which cannot be guaranteed.
6. Large digitalization of existing data sources would be needed which cannot be financed (literature science, history).
7. Learning environments need to be developed, which need to be well adapted to a specific context and situation.

But some subject matters are very well-suited to e-learning:

1. Virtual companies, business simulation games, stock market games, planning games, project planning.
2. Virtual laboratory, which help to reduce capacitive bottlenecks.
3. Models and simulations in architecture, mathematics, natural sciences, medicine, and economics.
4. Processes, which involve computers anyway (e.g. models in computer science or machines).
5. Problem-oriented learning, where sufficient clients or real objects (e.g. patients, animal experiments) for real-life practice are unavailable or where would it be too dangerous for beginners.
6. The preparation or wrap-up of learning, where real-world exploration is essential (e.g. field trips).
7. The storage and transmission of learning situations, when adequate situations are missing (e.g. bed-side teaching in medicine).

Example: Computer user interfaces can be presented and simulated easily in an e-learning unit.

1.1.7 Is the subject matter suitable to be chunked into small, modular and reusable Content Elements

The dLCMS uses the concept of basic building blocks, called “Content Elements”. A “Content Element” is defined as a small, modular unit of learning content, which:

- Serves as basic building block of learning content.
- Can be aggregated into larger, didactically sound learning units (creating a learning context).
- Is self-contained or self-standing (has no references to other Content Elements, e.g. links, but may have prerequisites to the learners’ knowledge).
- Represents a single didactic content type which relates to a single objective (a fact, a concept, a principle, a procedure, a process).
- Is reusable in multiple instructional contexts (reusable for other learning objectives or target learner groups as well as reusable with different instructional designs).
- May contain “Assets” (media elements).

Example: The subject matter can be chunked into usability concepts, usability principles and usability procedures, which are self-contained and therefore reusable in various educational contexts.

1.1.8 Existing learning content available for reuse

Are there already Content Elements for the chosen subject matter available which can be reused? Are there other kinds electronic learning content, e.g. existing Assets (images, videos, animations, simulations, etc.), existing e-learning applications or documents (slides, lecture notes), which provide material for the development of new Content Elements?

Example:
Content Elements: not available yet.
Assets: Plan of a usability laboratory, screen shots of selected user interfaces
Other Documents:

- "usability_pdf": collection of slide presentations and lecture notes on usability
- "ETH 04 Übung Nr2.doc": exercise for a user interface evaluation
- "8_VNET5_Inspection_Methods_v2-1.ppt": slide presentation of usability inspection methods

1.2 Learner profile

Familiarity with a learner profile helps one design a learning unit that is well adapted to the learner. It has been proven that consideration given to learners' characteristics in choosing learning methods has a strong influence on learning success.

1.2.1 Educational level (beginners or advanced; undergraduate, graduate or post-graduate; professional training)

The educational level is important with respect to the choice of the language style and the didactic guidance of the learner. It may also provide important information about overall learning habits.

Example: Post-graduate training.

1.2.2 Educational prerequisites (previous knowledge, professional qualification)

Educational prerequisites help to establish the knowledge and skills upon which the learning unit shall be based.

Example: Natural scientists, engineers, physicians. No specific knowledge.

1.2.3 Age of students

Example: 30 – 60 years old, mean: 40 – 50 years old.

1.2.4 Language(s)

Example: Mostly German speaking, about 20% French speaking, who understand German.

1.2.5 Ratio of Women and Men

Example: About 40% women and 60% men.

1.2.6 Learning motivation

The students learning motivation is important for the design of the learning unit and the conception of didactic methods. Highly motivated students achieve better results when they have more control over the learning steps taken, whereas students with low motivation are better served when carefully guided through the learning steps.

It might be helpful to analyze whether the motivation comes from personal or professional interest, or if the motivation is externally driven, e.g. to obtain credit points, a diploma or a title; a requirement by an employer.

Example: Most of the students (about 70%) are highly motivated based on their professional interest and personal interest. In their daily jobs they have to evaluate ergonomics of work places with computers and other machines. Also, they all use computers in their daily work. A minority attends the course for the certification as an ergonomic expert only.

1.2.7 Learning habits

The students learning habits should determine the didactic style. If they are not used to learning on their own, the learning unit should provide detailed guidance.

Example: All students have an academic degree and are used to learning on their own.

1.2.8 Previous e-learning and computer experience

The design of the e-learning unit must take into account the previous e-learning and computer experience. Novice computer users often have difficulties with orientation in hypermedia, whereas hypermedia has been successfully used with advanced e-learning users.

Example: All students are users of office applications, e-mail and web-browsing. Only very few might have previous experience with e-learning.

1.2.9 Minimal computer requirements, which student have access to

All students should be able to access the e-learning unit.

Example: All students have access to the internet and modern browsers (IE 6 or Mozilla 1.5).

1.3 Environment

The analysis of the environment points out the resources available for the development of the learning unit.

1.3.1 Availability of an e-learning platform

Is an e-learning platform available?

Example: OLAT at the University of Zurich, WebCT at NET.

1.3.2 Availability of development software

The availability of development software determines the kind of media that can be produced.

Example: dLCMS (Content Element text and assembly to Learning Units), Photoshop (images), Illustrator (graphics, diagrams), Forte for Java (simulations).

1.3.3 Financial resources

Example: None, student diploma thesis.

1.3.4 Personnel resources

Example: One person for one month.

2 Learning Unit Concept

The first step of the concept phase is the deconstruction of the major learning objective into more detailed sub-objectives. Then, the learning content and the didactic methods are specified, which should be used to teach the learning objectives.

2.1 Task Analysis / Analysis of Knowledge Domain

Depending on the type of instruction, whether skills for a specific task are required (what the learner should be able to do), or whether knowledge on specific subject matter should be taught (what the learner should know), different approaches are appropriate. With respect to skills, a task analysis deconstructs the tasks into their respective sub-tasks. With respect to the education of general knowledge, the knowledge domain is analyzed to reflect the topic to be taught and its logical structure. This should result in an overview of the sub-topics, their possible structure, and their relevance.

Example: This example documents the task analysis of a simple usability evaluation for a specified user and task. The steps to do this are:

1. Perform a user analysis.
2. Perform a task analysis.
3. Define usability attributes to be tested depending on user and task.
4. Set usability goals, their measurements and acceptable performance levels.
5. Choose appropriate usability evaluation method.
6. Perform usability evaluation.

2.2 Learning Objectives

Based on the major learning objective and the task analysis or the analysis of the knowledge domain, detailed learning objectives are formulated.

Example: With respect to the example of a usability evaluation learning unit, the detailed learning objectives could be formulated as follows:
The student should:

1. Be able to conduct a simple user and task analysis.
2. Be aware of the ISO 9241-10 usability attributes.
3. Be able to understand the meaning of the ISO 9241-10 usability attributes.
4. Know that there are other usability attributes: effectiveness, efficiency, acceptance, emotionality.
5. Be aware that there is no standard way to operationalize the usability attributes.
6. Know important operationalizations: time to complete a task, number of clicks, number of errors on a task, subjective rating.
7. Have a general overview of usability evaluation methods.
8. Be able to perform a heuristic evaluation.
9. Know what a usability test is and be aware of its importance.

2.3 Specification of Learning Content

In this step, we specify the content, which serves to teach the learning objectives defined above. It should provide a brief description of the topics and their sequence.

Example
- Anesthesia apparatus as an example of the importance of usability
- Usability definition of ISO 9241-11: effectiveness, efficiency, satisfaction
- General overview of the usability evaluation procedure
- User and task analysis
- Usability attributes of ISO 9241-10
- Operationalizations of usability attributes
- Usability evaluation method overview
- Example 1: Heuristic evaluation
- Example 2: Usability testing
- Exercise of a heuristic evaluation

2.4 Didactical Strategies and Methods

2.4.1 General didactic strategy of the e-learning unit

General classes of didactic strategies are:

- *Instruction-led expository (sequential) design.* The learner is guided step-by-step through the learning unit. This strategy may used for learners with low motivation or lacking hypermedia experience.
- *Learner-centered explorative (logically structured) design.* The learning content is structured in a logical manner and the student can navigate through the learning unit with hyperlinks. This strategy is suited for motivated learners who are experienced with e-learning and hypermedia.

Specifying a didactic model is another way to characterize the didactic strategy. Didactic models are, for example:

- Training
- Tutorial
- Simulation
- Game
- Problem-oriented learning
- Mastery learning
- Others

Example: The usability evaluation learning unit primarily uses a sequential design, proposing to the learner a pre-defined sequence through the arrangement of the learning content. However, the design allows the learner to navigate through the unit freely.

2.4.2 Didactic concept

What general didactic concept should be used? How should the subject matter be presented? How should the learner be involved? Should there be a test?

Example: The usability evaluation learning unit uses the following didactic methods:

- Motivation: Anesthesia apparatus as an example of the importance of usability.
- Definition: Usability definition of ISO 9241-11: effectiveness, efficiency, satisfaction (present the stimulus: definitions)
- Overview: The usability evaluation procedure
- Explanation: User and task analysis
- Explanation: Usability attributes of ISO 9241-10
- Explanation: Operationalizations of usability attributes
- Explanation: Usability evaluation method overview, analytical vs. empirical methods
- In depth example 1: Heuristic evaluation (present the stimulus: definition, explanation)
- In depth example 2: Usability testing
- Exercise: Perform a heuristic evaluation online.

2.4.3 Media to be used

List the media to be used with the learning unit.

Example: Text, images.

2.5 Graphical structure of the e-learning unit

The learning unit may be represented in a chart to visualize the structure. A Flow chart or another diagram may be used sketch the relationship of contents, navigational helps, etc.

3 Content Chunking

In the next step, the previously specified content is broken down into small chunks, the Content Elements. It is important to perform this step carefully to assure that the Content Elements will be reusable in multiple educational contexts.

An important factor affecting reusability of Content Elements is granularity, i.e. the size of the Content Elements. Granularity depends on the following points:

1. *Coverage:* The larger a Content Element is and the more topics and subtopics it covers, the less likely it is to be reused in a different educational context. In order to be a basic building block, a Content Element should be

related to a single objective only: a main concept, fact, principle, process or procedure.

2. *Didactic content types:* In multiple educational contexts, varying combinations of didactic content types may be used for the same topic. Examples of didactic content types are: definitions, instructions, examples, exercises, self-assessment, etc.

   Furthermore, varying contents for a particular didactic content type may be required for different target groups. A teacher might want to adapt the content to the professional context of the learners. For example, teaching the same statistical concept, medical students would be presented an example using patient populations, whereas economists may view an example using business success. Separating the didactic content types into single Content Elements enhances the flexibility of composing learning units with varying didactic methods.

3. *Level of elaboration:* The different possible learner groups may need different levels of elaboration. While experts will need profound knowledge on some topic, other learners are likely to need only basic understanding. Therefore, the individual Content Elements should be created for different levels of elaboration. With this approach, a Content Element with the basics can be used for general users as well as for experts, while the Content Element with in-depth information can be used for experts only.

These guidelines propose the following steps for chunking content into small reusable Content Elements:

1. Break the content down into single objectives.
2. Specify the didactic content types to be used.
3. Define other potential learner groups.
4. Analyze the content for reuse with other potential learner groups.

Using a spreadsheet for the Content Chunking procedure allows for flexible planning and development of the Content Elements. The work can be continuously refined by adding a line for a new Content Element whenever necessary. At the same time, it maintains a concise view of the Learning Unit structure. This flexibility should emphasize the importance of jumping back to earlier planning steps when needed, even during the development phase of new learning media.

It is further suggested to add a column for the media to be used with the Content Element, and another column for already existing material intended to be used. This material may be already existing Content Elements or sources of learning material, which provide a basis for the development of new Content Elements. The additional columns are meant to be an aid for the specification of Content Elements and may be used as needed.

3.1 Break the content down into single objectives

The aim of this step is to analyze the structure of the content and to break it down into topics and subtopics, each of which is centered around a single objective. Subtopics may be indented to visualize the structure of the content. These items should be labeled by their content category:

- *Concept:* A concept is used to teach a group of objects, symbols, ideas, or events. Concepts are designated by a single word or term, share a common feature, and vary on irrelevant features.
- *Fact:* A fact is used to teach unique, specific, one-of-a-kind pieces of information. Facts are presented as statements, data, or pictures of specific objects.
- *Procedure:* A procedure is used to teach a performance.
    - A procedure is a sequential set of steps to be followed by one individual to accomplish a task or to make decisions.
    - A procedure lists directions for procedural tasks.
    - Actions within a procedure must be done the same way each time (within a given situation).
- *Process:* A process is used when you need to teach how a system works. This is helpful in supporting underlying job tasks, providing motivation, and ensuring overall quality of job performance. A process can be characterized as:
    - A flow of events that describes how something works
    - Not a task to be done by one person
    - Many persons or organizations are involved
    - Mechanical, business, or scientific
- *Principle:* Principles are explanations or predictions of why things happen in the world. They are cause-and-effect, correctional, or constraint relationships. Principles are further used when a task requires judgment or when guidelines must be applied to a situation.

Example	
Topic	*Content category*
1 Anesthesia apparatus as an example of the importance of usability	Concept
2 ISO 9241-11 Usability definition	Concept
2.1 Effectiveness	Concept
2.2 Efficiency	Concept
2.3 Satisfaction	Concept
3 Usability evaluation	Procedure
3.1 User and task analysis	Concept
3.1.1 User and task analysis methods	Procedure
3.2 Usability attributes	Concept
3.2.1 ISO 9241-10	Concept
3.2.1.1 Suitability for the task	Principle
3.2.1.2 Self-descriptiveness	Principle
3.2.1.3 Controllability	Principle
3.2.1.4 Conformity with user expectations	Principle
3.2.1.5 Error tolerance	Principle
3.2.1.6 Suitability for individualization	Principle
3.2.1.7 Suitability for learning	Principle
3.2.2 Other usability attributes	Concept
3.2.2.1 Emotionality	Concept
3.3 Operationalizations of usability attributes	Concept
3.3.1 Time to complete a task	Concept
3.3.2 Number of errors made	Concept
3.3.3 Number of clicks	Concept
3.4 Usability evaluation methods	Procedure
3.4.1 Heuristic evaluation	Procedure
3.4.1.1 List of heuristics	Principle
3.4.2 Usability testing	Procedure
3.4.2.1 Task for usability testing	Principle
3.4.2.2 Test method and procedure	Procedure
3.4.2.2.1 Think aloud protocol	Procedure
3.4.2.3 Test material and equipment	Concept
3.4.2.3.1 Usability Lab	Fact
3.4.2.4 Recruiting Test Participants	Procedure
3.4.2.5 Conducting the test session	Procedure
3.4.2.6 Analyzing and communicating results	Procedure
3.4.2.6.1 Highlight video	Concept
3.5 Literature	Fact

3.2 Specify the didactic content types to be used

In this step, didactic content types are assigned to every topic or subtopic. The didactic content types suggested are:

Expository	Definition	Narrative
Instruction	Example	Excursion
Glossary	Literature	Summary
Learning objectives	Advance organizer	Exercise
Questionnaire	Exam	Self-assessment
Experiment	Problem statement	Simulation

If a topic or subtopic should contain more than one didactic content type, one may consider creating a separate Content Element for every didactic content type. E.g. a topic may contain a definition, an expository for an explanation, and an example. In this case, three Content Elements might be created, one for the definition, one for the expository and one for the example.

Example		
Topic and subtopics	*Content category*	*Didactic content type*
1 Anesthesia apparatus as an example of the importance of usability	Concept	Example
2 ISO 9241-11 Usability definition	Concept	Definition
2.1 Effectiveness	Concept	Definition
2.2 Efficiency	Concept	Definition
2.3 Satisfaction	Concept	Definition
3 Usability evaluation	Procedure	Expository
3.1 User and task analysis	Concept	Expository
3.1.1 User and task analysis methods	Procedure	Expository
3.2 Usability attributes	Concept	Definition
3.2.1 ISO 9241-10	Concept	Expository
3.2.1.1 Suitability for the task	Principle	Definition
		Example
3.2.1.2 Self-descriptiveness	Principle	Definition
		Example
3.2.1.3 Controllability	Principle	Definition
		Example
3.2.1.4 Conformity with user expectations	Principle	Definition
		Example
3.2.1.5 Error tolerance	Principle	Definition
		Example

3.2.1.6 Suitability for individualization	Principle	Definition
		Example
3.2.1.7 Suitability for learning	Principle	Definition
		Example
3.2.2 Other usability attributes	Concept	Expository
3.2.2.1 Emotionality	Concept	Definition
		Example
3.3 Operationalizations of usability attributes	Concept	Expository
3.3.1 Time to complete a task	Concept	Definition
3.3.2 Number of errors made	Concept	Definition
3.3.3 Number of clicks	Concept	Definition
3.4 Usability evaluation methods	Procedure	Overview
3.4.1 Heuristic evaluation	Procedure	Definition
		Expository
3.4.1.1 List of heuristics	Principle	Definition
		Exercise
3.4.2 Usability testing	Procedure	Definition
		Instruction
3.4.2.1 Task for usability testing	Principle	Expository
		Example
3.4.2.2 Test method and procedure	Procedure	Overview
3.4.2.2.1 Think aloud protocol	Procedure	Expository
3.4.2.3 Test material and equipment	Concept	Expository
3.4.2.3.1 Usability Lab	Fact	Example
3.4.2.4 Recruiting Test Participants	Procedure	Expository
3.4.2.5 Conducting the test session	Procedure	Expository
3.4.2.6 Analyzing and communicating results	Procedure	Expository
3.4.2.6.1 Highlight video	Concept	Expository
3.5 Literature	Fact	Literature

3.3 Define potential learner target groups

This step involves defining potential learner target groups (in a short brainstorming session) to identify three to five types of individuals who might benefit from some of the learning content being created.

Example

- Postgraduate students of ergonomics (Ergo.) - target learner group of this learning unit
- Formation of usability experts (Usab.)
- Usability course for web designers (Web D.)
- Usability course for programmers (Prog.)
- Usability workshop for managers (Manag.)

3.4 Analyze the content for reuse with other potential learner groups

As a last step, the content structure is analyzed with respect to reuse for other potential learner groups. For each item and learner group, check if:

1. the Content Element could be potentially reused for this learner group,
2. the coverage planned would suit the potential learners.

If some parts of the Content Element planned fit the potential learner's needs but some other parts seem to be out of scope, one may consider dividing the content into two Content Elements.

Example						
Topic and subtopics	*Didactic content type*	*Target learners*	*Other potential learners*			
		Ergo.	*Usab.*	*Web D.*	*Prog.*	*Manag.*
1 Anesthesia apparatus as an example of the importance of usability	Example	X	X			X
2 ISO 9241-11 Usability definition	Definition	X	X	X	X	X
2.1 Effectiveness	Definition	X	X	X	X	X
2.2 Efficiency	Definition	X	X	X	X	X
2.3 Satisfaction	Definition	X	X	X	X	X
3 Usability evaluation	Expository	X	X	X	X	X
3.1 User and task analysis	Expository	X	X	X	X	X
3.1.1 User and task analysis methods	Expository	X	X	X	X	
3.2 Usability attributes	Definition	X	X			
3.2.1 ISO 9241-10	Expository	X	X	X	X	X
3.2.1.1 Suitability for the task	Definition	X	X	X	X	
	Example					
3.2.1.2 Self-descriptiveness	Definition	X	X	X	X	
	Example					
3.2.1.3 Controllability	Definition	X	X	X	X	
	Example					

3.2.1.4 Conformity with user expectations	Definition	X	X	X	X	
	Example					
3.2.1.5 Error tolerance	Definition	X	X	X	X	
	Example					
3.2.1.6 Suitability for individualization	Definition	X	X	X	X	
	Example					
3.2.1.7 Suitability for learning	Definition	X	X	X	X	
	Example					
3.2.2 Other usability attributes	Expository	X	X	X	X	
3.2.2.1 Emotionality	Definition	X	X	X	X	X
	Example					
3.3 Operationalizations of usability attributes	Expository	X	X			
3.3.1 Time to complete a task	Definition	X	X			
3.3.2 Number of errors made	Definition	X	X			
3.3.3 Number of clicks	Definition	X	X			
3.4 Usability evaluation methods	Overview	X	X	X	X	X
3.4.1 Heuristic evaluation	Definition	X	X	X	X	
	Expository					
3.4.1.1 List of heuristics	Definition	X	X	X	X	
	Exercise					
3.4.2 Usability testing	Definition	X	X			
	Instruction					
3.4.2.1 Task for usability testing	Expository	X	X			
	Example					
3.4.2.2 Test method and procedure	Overview	X	X			
3.4.2.2.1 Think aloud protocol	Expository	X	X			
3.4.2.3 Test material and equipment	Expository	X	X			
3.4.2.3.1 Usability Lab	Example	X	X			
3.4.2.4 Recruiting Test Participants	Expository	X	X			
3.4.2.5 Conducting the test session	Expository	X	X			
3.4.2.6 Analyzing and communicating results	Expository	X	X			
3.4.2.6.1 Highlight video	Expository	X	X			
3.5 Literature	Literature					

Appendix B: Detailed Results of the Pilot Project Study

Author 1

Learning Unit Analysis

Subject matter

The subject matter of the learning unit was “The Crossed Cylinder Technique of a Monocular Subjective Refraction” and the major learning objective was “mastery of the crossed cylinder technique”. The learning unit was to be a self-study lesson with no related classroom training. The expected learning time was to be one hour and a half at most.

The participant rated the subject matter to be 90% suitable for modularization using a scale from 0% to 100% (0% meaning unsuitable, 100% meaning very suitable). She justified this rating, saying that the refraction process could be easily divided into clearly defined steps. Asked for an example of a Content Element, she suggested that the step-by-step description of the procedure “to determine the cylinder axis” could be a Content Element.

The participant rated the suitability of the subject matter for e-learning in general to be 40%, because she thought it would be difficult to implement a practical training for the manipulation of the instruments. She stated that the suitability for e-learning could be up to 80% with many financial and personal resources (scale from 0% to 100%, 0% meaning unsuitable, 100% meaning very suitable).

She had used books, lecture notes and expert interviews to acquire learning material which had been classified into the categories “declarative knowledge” and “procedural knowledge”. Declarative knowledge included facts of physical optics and functions of instruments; procedural knowledge included the handling of instruments, the interpretation of measurements and the interaction with a patient. After an initial structuring of the collected learning material, subject matter experts had reviewed the material to remove irrelevant or redundant parts.

Learner profile

The course was to be part of a postgraduate professional training. The educational prerequisites were basic knowledge on the physiology of the eye considered as basic by ophthalmologists. The typical age of the students was to be between the ages of 25 and 30. The students were anticipated to be German speaking; the ratio of women and men was expected to be well-balanced. The students were thought to be motivated extrinsically and to have a very tight work schedule. It was expected that the students would have previous computer and e-learning experiences and would have access to a personal computer with internet access.

Learning Unit Concept

Didactical strategies and methods

The Learning Unit Concept has already been implemented as an example of a chapter using the participants' proprietary system.

Because the refractions process is a linear one, the didactic concept was primarily designed to provide a clear linear learning path, but using the navigation the students were free to choose their own learning paths. The whole refraction course contained several chapters – "The Crossed Cylinder Technique" unit, which was chosen to be transferred to the dLCMS for this study, was such a chapter. This chapter was divided into 13 scrollable pages, each of which was to be no larger than two computer screens. The participant used the notion "learning unit" for such a single page.

Each page contained an "expert comment" summarizing the issues to remember. The expert comment was implemented as a speaking animated figure; the speech was synthesized from text. The page also contained superscript "endnote" numbers. When moving the cursor over such a number, the corresponding glossary entry popped up on the screen. The pages integrated text, images, graphics, video, and animations. Video clips were used to demonstrate how to perform the refraction procedures and how to interact with a patient. The last page contained a multiple-choice quiz that only appeared if the student had worked though all pages. The original system indicated on which pages the student already had worked. Personal notes could be attached to every page.

Content Chunking

An existing chapter, called "The Crossed Cylinder Technique", was the starting point of the content chunking process. The HTML content could be accessed during the workshop using a web browser. The chapter was already divided into 13 pages. Although the participant received the chunking guidelines a few days in advance, she did not read them until the workshop. Supported by the coach, the participant proceeded step-by-step though the guidelines.

Breaking contents down into single objectives

The participant deleted the example data in the chunking spreadsheet and copied the titles from the existing table of contents into the first column. In the second column, she assigned content categories from the set given in the guidelines to the titles. The following problems occurred in this phase:

- None of the given content categories could be easily assigned to the two titles “The Optical Structure of a Cross Cylinder” and “The Schematic Structure of a Cross Cylinder”. It seemed as if a category like “structure” was missing. In the end, “fact” was chosen as content category.
- It was difficult to assign a content category to the title “Particularities Carrying-Out the Crossed Cylinder Technique”. The type “procedure” seemed to be the most appropriate, but the participant stated that “procedure” did not fit well for a description of particularities.
- Generally, the participant stated that the content categories were difficult to understand. For her, “concept” and “principle” were difficult to distinguish.

Specifying the didactic content types

In the next step, the participant assigned didactic content types from the list given in the guidelines to the titles. In the beginning she had difficulties understanding the meaning of the given terms – “advanced organizer” and “self-assessment” were not clear. She asked for a category “image”. The coach explained the didactic content types.

As a result, the didactic content type “expository” was added to all but one title. This one title got the type “definition” instead. Two titles were not given any further didactic content types. “Glossary” was added to almost all remaining titles with one exception. “Test” was assigned to two different titles; “animation” and “example” were used for one title each. No more than three didactic content types were assigned to a single title.

Analysis for reuse with other learner groups

As it was a very specialized subject matter, the participant could not identify other potential learner groups than the original target group, ophthalmologists and opticians. Therefore, she did not analyze the content structure with respect to reuse for other potential learner groups.

Results of the semi-structured interview

IQ1: She used her own method as she already had developed structured contents.

IQ2: Referring to the analysis phase, she rated the subject matter as not suited for e-learning.

IQ3: She rated the subject matter as well-suited for modularization, as in the analysis phase.

IQ4: She would divide the learning unit into single components (for example, definitions), creating smaller sections and she would try to structure it more clearly with less exploratory text. She would want to do this in close cooperation with the lecturer – the subject matter expert.

IQ5: In her opinion, learners would get a better overview. They could navigate more easily to areas they considered to be more interesting and thought to be important.

IQ6: She stated that the motivation to modularize content in a systematic manner was generally high. At the beginning, there was something akin to a “soup of contents”. Structuring and categorizing would help to get closer to the implementation. People with no idea of didactics would confuse ”content categories” and “didactic content types”.

IQ7: She reported that modularization would pay-off, because it has to be done anyway if you have to develop an e-learning application on the basis of lecture notes.

Content Development and Learning Unit Assembly

Content Development

At the beginning of the development phase, the participant opened the edit view of the dLCMS with her browser (Internet Explorer). Although she already had a look at the system some months before, the amount of information presented on the screen was overwhelming. The coach quickly explained the system.

After the explanation, the participant navigated to the existing learning object repository and added a new folder for her contents. Inside this folder she created new Content Elements. The existing chapter in her system, “The Crossed Cylinder Technique”, was accessible using a second browser window. She copied the HTML contents from this window and pasted them into the newly created Content Elements using the online-editor. The contents of each page were copied at once. In the end, she had copied contents creating five new Content Elements. During this process the following problems occurred:

- The glossary entries were not copied with copy and paste. In the original system, the glossary entries were displayed as pop-up fields when the user moved the cursor over a superscript “endnote” number.
- Contents with images, which were pasted into the online editor, were displayed correctly until the Content Elements were saved. Afterwards the images were lost. To correct this, the images had to be imported separately and the paths referring the images had to be adjusted. The same problem happened if the content contained multimedia elements, i.e. a Flash animation and an AVI video.
- If multimedia elements were inside a table, saving the Content Elements produced an error. The system did not support multimedia elements inside tables. The tables were used to position text next to the element.

- It was not clear what to do with the expert comments. The participant agreed that it would be sufficient to paste the text into the Content Elements. She explained that it would suffice without synthesized speech.

We observed that the tables that the participant had copied were used to position text at the right side of images or multimedia elements. Looking at the resulting Content Elements, the text columns on the right side were very narrow and very tall, which the participant disliked. This was caused by the dLCMS layout template, which had a fixed width for the contents area, which was not as large as the content area for which the table was originally designed.

The participant then created a new questionnaire element for the self-assessment related to the first Content Element. She added a multiple-choice question with three possible answers to the element. She asked if the questions could have a separate title. Noticing that the answers were always displayed in the same order, she remarked that she preferred a random ordering, as in her system.

Learning Unit Assembly

In the next step, the participant assembled the Content Elements, which were created in the Content Development phase, into a Learning Unit. Using the Lesson and Lesson Element objects, she created a top node and below this top node, a sequence of sub-nodes. Each node was immediately linked to the appropriate Content Element. This was done using the Content Element chooser window. During this process the following points were noticed:

- Using the chooser window, the participant navigated to her new Content Elements. All Content Elements, which should be assembled into the Learning Unit were there, but only one Content Element could be selected one by one. She asked if all Content Elements could be selected at once – that would save many mouse clicks.

- The system automatically used the title of the Content Element which is linked to the top node as the title for the whole Learning Unit. The participant stated that the overall title is different from the title of the first Content Element.

Results of the semi-structured interview

IQ8: The participant stated that the users of the dLCMS would need a good introduction. The user interface was not intuitive, and there were too many controls. The labels of the controls were often incomprehensible.

IQ9: Nothing was very difficult for her – only the terminology used and the orientation would need to be made clearer.

IQ10: She said that she did not need any markup elements. She could do what she wanted. Therefore, the interview questions *IQ11* to *IQ13* were not answered.

IQ14: The biggest problem for her was the editing of the texts as small modular Content Elements.

Analysis of the Final Content Elements

At the end of the Chunking and Development Workshop, six Content Elements had been created. All Content Elements were clearly self-contained, but analyzing the didactic content types and the single objective was difficult.

The Content Elements CE1.1 and CE1.2 did not clearly point out key statements. They were rather a mixture of several fragments of information which were somehow related to each other. It was not clear if these elements as a whole should be rated as one "extensive" expository element relating to single objectives or if the various fragments should be rated as separate pieces of learning contents. From our point of view, the lack of clarity adversely affected the didactic quality of the Content Elements. Therefore, we decided to rate each fragment individually. It should be noted that these contents were originally copied from lecture notes, which illustrates how difficult it is to create modular contents based on this type of existing materials.

Further, CE1.2 contained two classification schemas and provided six definitions of the categories presented. However, the Content Element provided a concise overview of these definitions.

Only two Content Elements could be clearly rated to represent a single didactic content type representing a single objective (CE1.3, CE1.6).

Table 8.1 tabularizes the results of our analysis of the Content Elements created by Author 1.

Table 8.1 Analysis of the Content Elements created by author 1.

	Self-contained	Didactic content types	Single didactic content type	Content Categories	Single objective
CE1.1	Yes	Expository, 2 Definitions	No	3 Concepts, 1 Principle	No
CE1.2	Yes	Expository, 6 Definitions	No	2 Principles, 6 Concepts	No
CE1.3	Yes	Expository	Yes	1 Principle	Yes
CE1.4	Yes	Expository, 2 Definitions	No	3 Concepts	No
CE1.5	Yes	Expository, 1 Example	No	1 Procedure	Yes
CE1.6	Yes	1 Self-assessment	Yes	1 Concept	Yes

Author 2

Learning Unit Analysis

In order to prepare for the *Analysis and Concept Workshop*, the participant had already read the guidelines and had worked on the forms. She had copied text from the Webclass project proposal onto the forms. Therefore, the data of the *Learning Unit Analysis* and *Learning Unit Concept Form*s related to the whole Webclass project and not only to the learning unit that was to be created for this study.

Subject matter

The subject matter of the learning unit was the “History of Energy”. As a major learning objective, the students were to understand that technical innovations take place in complex economical, political, and cultural contexts. The course was planned to be combined with classroom training. Classroom training consisted of at least one introductory class and further possible tutorial classes. The introductory class was intended to help prevent online anonymity and to clarify questions, such as the procedure and the requirements to attain credit points. The amount of time the students were expected to work for the course was akin to a traditional lecture with two hours per week during one semester. The students of the Swiss Federal Institute of Technology needed some credit points in the field of human and social sciences. This course offered such credit points.

The participant rated the subject matter to be 50% suitable for modularization using a scale from 0% to 100% (0% meaning unsuitable, 100% meaning very suitable). She stated that the comprehension of the topics would need a didactic thread. In her opinion, it would be difficult to create independent content elements that could be reused in different learning contexts because special elements may be needed to create a didactic thread. As an example of a Content Element, she referred to the sections of the existing course, which addressed a historical topic or question combining text and an image or an audio asset. To be more precise, she specified the example having a title, some text, and an image (see Figure 6.2).

The participant rated the suitability of the subject matter for e-learning in general to be 80%. For her, a trade-off in using e-learning for the subject matter was that the traditional discussions on the subject matter in the classroom were lost. The participant stated that discussions could not simply be replaced with internet chats. Internet chats would have to be encouraged and would need new ways of moderating. She also justified the rating of suitability with the number of students attending the online courses. Approximately 200 students used the online course as compared to 20 students who had previously attended the classroom lectures. The students seemed to appreciate the advantages of learning anytime and anywhere for mandatory credit points outside of the core discipline of their studies.

The learning contents of the Webclass project were based on PowerPoint presentations of the original lecture. Much work was invested in writing the texts, which had been presented orally in the original lectures.

Learner profile

The course was designed as an introduction to the history of technology for graduate students of the Swiss Federal Institute of Technology. These students were the main target group, but students of the University of Zürich could also attend the course. No educational prerequisites on the subject were needed. The students' mean age was 22; most of them were men. Language skills in German were required. It was anticipated that all students would speak German, although their first language might be Italian, French or English.

The student's main motivation was anticipated to be primarily based on the need for mandatory credit points in the field of human and social sciences, and secondarily on the curiosity of working with an online course; only in very few cases was the motivation anticipated to be based on specific an interest in the subject matter. The students' previous e-learning and computer experiences were supposed to be heterogeneous. All students had access to the computers in university public computer rooms.

Learning Unit Concept

Didactical strategies and methods

There were two aims important for the didactic concept of Webclass. On the one hand, the contents were designed to present the newest research results and methods of the history of technology. On the other hand, the authors wanted to gather experiences in editing non-sequential hypertext. In was clear that the traditional structuring of the subject matter needed to be adapted in favor of modularization, which supported easy readable hypertext and which provided a clear outline of the contents. Therefore, short blocks of text and multimedia elements were combined with links to other texts, graphics, movies, and audio material.

The presentation of transparent learning objectives was supposed to help the students to reflect the contents. The course provided interpretations of historical facts that were explicitly stated. Student tasks were designed to enable students to acquire the methods of history actively. For example, students were asked to conduct an internet enquiry on some topic. As the target learner group consisted mainly of students from engineering and natural sciences, a multidimensional approach was taken, aimed at an understanding of economical, ecological, cultural, and social questions. Whenever possible, historical sources were used. The integration of historical pictures, movies, and audio recordings was emphasized in order to achieve an attractive presentation as well as to enhance the students' awareness of the historical objects.

The resulting structure of the contents was designed to enable a flexible use of the historical contents so that they could be combined with different kinds of didactical methods (self-study, lecture, seminar, research colloquium). Each module was divided into 10 to 12 lessons, which could be used as a whole or as single lessons. The lessons consisted of three main types: the “theme pages” discussing the topic (approximately six pages), the so-called “context pages” providing background information, and a page containing an interpretation or a student task. Each page had a similar structure: an introduction, several sections, and a “material box”. Webclass used pop-up windows and so-called “material boxes” to reduce the download time of contents containing large multimedia elements. The “material boxes” contained links to external web pages, pages with images, PDF documents, additional learning material, and a page containing the bibliography.

The design of graphical elements was considered to be very important. It was to support intuitive navigation; e.g. historical citations should always be presented using the same graphical fields.

Content Chunking

An existing lesson, the “History of Energy”, was the starting point of the content chunking process. The HTML contents of the lesson could be accessed during the workshop using a web browser. The page was already divided into clearly distinguishable sections, each with a title and some text. Most of the sections also contained an image. Although the participant received the chunking guidelines a few days in advance, she did not read them until the workshop. Supported by the coach, the participant proceeded step-by-step though the guidelines.

Breaking contents down into single objectives

The participant deleted the example data in the chunking spreadsheet and copied the titles from the existing sections into the first column. In the second column, she tried to assign the content categories to the titles.

The participant was not able to assign a content category to the first two section titles. None of the knowledge categories seemed to be suitable. The first section was an introduction. It contained a problem statement, which asked for an interpretation schema to be used as an approach to the history of energy. As the contents presented a methodological question, the content category “process” was considered by the participant and the coach. However, the contents did not contain a step-by-step procedure. Although the section provided a definition of the term “energy”, the content category “concept” did not seem to fit either because it did not cover to the methodological aspect.

The same problem occurred in trying to assign a content category to the second section. Having lost much time in the meanwhile, it was decided to categorize the sections according to the participant’s own classification scheme instead.

The participant explained the categorization defined in the Webclass project. Some items had been based on contents, others on the graphical appearance. These items were referred to as “elements”. On the one hand, this categorization had evolved during the process of content editing. On the other hand, some graphical elements (e.g. an interesting image) had inspired the authors to create new contents. The modularization could therefore be characterized as an interplay of contents and graphical elements. The participant specified the elements used:

- Title, text, image
- Image using the width of the whole content area
- Comparison of images (two images positioned next to each other)
- Enumeration
- Citation box, poster box (font size larger than citation), definition box
- Material box
- Interpretation
- Student task (essay etc.)

Specifying the didactic content type

Looking at the list of didactic content types provided by the guidelines, the participant did not understand all English terms (as the participant’s first language was German). Furthermore, she criticized that there were no descriptions of the didactic content types in the guidelines. The coach explained the types that she did not understand. Then, the participant marked all didactic content types in the list that were used in the Webclass project. These types were: *expository, definition, narrative, instruction, example, excursion, literature, summary, exam, problem statement,* and *simulation*. Assigning the didactic content types to the specified sections did not seem to have an impact on the chunking results for the participant, the coach and the observer. Therefore, it was decided to proceed to the next step of the proposed chunking procedure.

Analysis for reuse with other learner groups

The participant identified four other potential learner groups for whom the contents could be reused: MTU students, students of the University of Zürich, students of other universities of applied sciences, and eventually, managers (the lectures of the MTU program were attended by the students of the *Department of Information Technology and Electrical Engineering* and were comparable to the lectures in the field of human and social sciences as described above).

All contents were considered to be reusable for all potential learner groups. Thus, the analysis for reuse with other learner groups did not yield new content elements.

Analyzing the potential reuse for other learner groups, the participant had doubts if her team would want to share the content to be used for other target learner groups. She did not want to give away the contents, because much effort was needed to create them. In the least, it should be guaranteed that each content element would contain the name of the authors. Another problem with respect to sharing contents was the copyrights on some contents (images) that were owned by other parties.

Results of the semi-structured interview

IQ1: She said that she used her own chunking method for the first step, as proposed by the chunking guidelines, based on the graphical elements described above. The reasons for this were the existing sections, which were suitable for Content Elements, and the problems in assigning content categories. She stated that she could perform the other steps as described in the guidelines.

IQ2: Referring to the analysis phase, she rated the subject matter suitability for e-learning to be 80%.

IQ3: She rated the subject matter to be 50% suited for modularization, as in the analysis phase.

IQ4: Modularization did not change the way she would think about a learning unit as she was able to copy the existing course.

IQ5: Compared to her habitual ways of creating learning units, the lesson did not change didactically, but the presentation of the material was very different as each Content Element was displayed in its own window.

IQ6: For her, modularization was laborious. She compared it to a database that somebody has to fill with contents before others can benefit. A great number of elements would be stored in the database, but the user could not immediately see the entirety.

IQ7: She stated that modularization did not pay-off if contents were modularized afterwards, but that the effort would pay-off for a new project. As there was no project with other professors so as to share contents, the effort would not pay-off for her team. She was not sure if professors, as they were very busy, would take part in a cooperation project. Her team was also not planning any cooperation. The efforts to develop the existing course should be first amortized. In her opinion, the different perspectives of professors on the subject matter might make it difficult to establish cooperation.

In the end, the participant's general impression of the Content Chunking procedure was that the content chunking process would probably yield a good didactic structure of the contents, but the extra effort would only pay-off if the contents will be reused by other authors.

Content Development and Learning Unit Assembly

Content Development

To start working with the dLCMS, the participant opened the edit view with her browser. Looking at all the information on the screen, she said that she did not know much about the system anymore (she had tested the system about a year beforehand). The coach then answered questions and provided some instructions. The main questions were:

- Where should the contents be stored or organized? How should the dLCMS repository objects be used? Could the Webclass structure, as a “theme – context – student task”, be represented by the organization of the repository?
- How should Learning Units be created?

The participant decided to create a new repository and to have a sub-folder for every lesson. She then created the repository and a sub-folder for the lesson from which a page was to be transferred to the dLCMS. The page of the existing Webclass course was accessible in a second browser window. She copied the HTML contents from this window and pasted them into the newly created Content Elements using the online-editor. The sections were copied at once. In the end, she had created seven new Content Elements. During this process, the following problems were encountered:

- Content with images, which was pasted into the online editor, was displayed correctly until the Content Element was saved. Afterwards the images were lost. The same happened to the captions that were copied together with the images.
- The participant wanted to store the images centrally in a separate folder. She stated that using this technique, all images would have the same path that would not change. Having the same base path for all images would be an advantage if Content Elements referring to the images were moved, or if images would be used in different Content Elements. It would further ease the update of images, which were stored and edited in a folder on the participant’s personal computer. She did not like putting all images in a subject folder inside the repository. Putting all images in such a folder would not be appropriate to the term “subject”. Therefore, she created a “images” Silva folder outside the repository.
- The participant used tables to position images next to text fields (layout tables). She had already heard that this was considered to be bad practice using content management systems, because this practice ignored the principle of separating contents and presentation (layout). To her, it was important to use different layouts of text and images. The coach showed

her how to make images “float right”. Using this technique, the subsequent text was displayed on the left side of the image. The participant stated that this was an elegant solution for this case. However, the participant continued to use layout tables for the subsequent Content Elements, because copying and pasting the original layout tables was much easier to perform. She would revise the Content Elements later, if needed.

- The participant wanted to preserve the layout of the “material box”. It used a table with two columns in the original layout. The left column contained icons indicating the link type (references to resources on the web, bibliography); the right column contained textual hyperlinks. She copied the original table into a separate Content Element and uploaded the icons, and added them to the table.

Learning Unit Assembly

In the next step, the participant assembled the Content Elements, which were created in the Content Development phase, to a Learning Unit. The following problems occurred in this process:

- At the beginning, it was not clear where to create the new Learning Unit. The participant first tried to add the lesson to the repository. This was not allowed by the system. Therefore, she concluded that Learning Units should be created outside of the repository.
- Using the Content Element chooser window to select the Content Elements, the participant asked if she could select all Content Elements at once. To her, it was very annoying to add the Content Elements one by one, as this required many mouse clicks.
- Looking at the resulting Learning Unit, she asked if she could combine the Content Elements into a single page. During the subsequent discussion, she was not sure which was better: having several Content Elements on one page, or displaying each Content Element in a separate window. If all Content Elements were displayed on a single page, as in the original course, the learner would have to scroll the page. If each element were displayed in its own window, the user would have to click through the Learning Unit. She stated that, in general, small pieces of information were advantageous for the web, but then the Content Elements would need to be rewritten. She definitely did not like the “material box” displayed on a separate page: through this arrangement the “material box” seemed to her to be on the same level of importance as the other Content Elements, which was not intended by the authors.
- She stated that the overall lesson title should be different from the title o the top Content Element.

Results of the semi-structured interview

IQ8: (1) The participant stated that the dLCMS was easy to use if Silva is known. (2) To her, it was unclear how the metadata could be used; e.g. was it possible to search for "all Content Elements for beginners"? (3) She was not yet able to estimate the added value of the dLCMS. (4) She was not able to transfer the existing page one to one. (5) To her, presenting every Content Element in its own page would produce too many pages.

IQ9: The most difficult aspect for her was the comprehension of the structuring elements to assemble Content Elements into Learning Units. Furthermore, missing or inadequate functionality (e.g. the copying of image captions) was difficult to handle.

IQ10: She had problems with images and the "material box".

IQ11: She could not answer this question because the task was to transfer the existing material. No development, and therefore no creativity, was required.

IQ12: She said that there were generally sufficient elements, provided that inline images could be used.

IQ13: For her, special elements for the "material box", picture credits, and bibliography were missing.

IQ14: She stated that the following problems should be solved: (1) The Learning Unit should have its own separate title, (3) it should be possible to combine several Content Elements onto one page, (4) the layout and styling of the "material box" should be customizable, and (5) pop-up windows for videos should be displayed without browser controls, e.g. the "back" and "forward" buttons.

Analysis of the Final Content Elements

At the end of the Chunking and Development Workshop, seven Content Elements had been created. Two elements were not self-contained (CE2.3, CE2.7): one linguistically referred to the preceding Content Element using the word "also" and one was a collection of links to further information resources. Three Content Elements represented an expository and one or two examples (CE2.2, CE2.4, CE2.5). However, the expository and the examples were tightly coupled to each other so that these Content Elements appeared to be a single content entity. The remaining elements clearly represented a single didactic content type (CE2.1, CE2.3, CE2.6, CE2.7). All Content Elements were clearly related to a single objective. It should be pointed out that one Content Element was assigned a new didactic content type category, named "introduction", because it did not expose a concept or an idea (CE2.1), but rather contained an opening to the topics presented.

Table 8.2 tabularizes the results of our analysis of the Content Elements created by Author 2.

Table 8.2 Analysis of the Content Elements created by author 2.

	Self-contained	Didactic content type	Single didactic content type	Content Categories	Single objective
CE2.1	Yes	Introduction	Yes	1 Concept	Yes
CE2.2	Yes	Expository, 1 Example	No	1 Concept	Yes
CE2.3	No[1]	Expository	Yes	1 Concept	Yes
CE2.4	Yes	Expository, 1Example	No	1 Concept	Yes
CE2.5	Yes	Expository, 2 Examples	No	1 Concept	Yes
CE2.6	Yes	Expository	Yes	1 Concept	Yes
CE2.7	No[2]	Literature	Yes	1 Concept	Yes

[1] Linguistic reference to other Content Elements.
[2] Links to the bibliography and additional external resources.

Author 3

Learning Unit Analysis

Subject matter

“Silva for Web Authors” was the subject matter of the learning unit. The major learning objective was supposed to be the ability to create and modify web pages using the content management system “Silva”. The learning unit was intended to be a self-study tutorial. The online tutorial was also to offer the possibility to get to know the system before attending a classroom training, should the learners had wanted to do so.

The participant was not yet able to rate the subject matter’s suitability for modularization. She said that some of the procedures in Silva contained many steps, but these should be presented as a whole. Using movies to visualize to procedures, the subject matter would be suitable for modularization. Without movies, the descriptions might be lengthy. Asked for an example of a Content Element, she suggested that “the procedure to create a Silva folder” could be a Content Element.

She rated the suitability of the subject matter for e-learning in general to be 100% (scale from 0% to 100%, 0% meaning unsuitable, 100% meaning very suitable).

Learner profile

The learners were assumed to be motivated on their own. It was also important that only authorized learners would be able to access the tutorial.

Learning Unit Concept

Didactical strategies and methods

As this was a tutorial, the learners were to be directed step-by-step though the important procedures of the content management system:

- Editing contents
- Creating, modifying, and publishing page and folders
- Using hyperlinks, images, and PDF files
- Etc.

It was assumed that the learner would follow the tutorial displayed in one browser window. In a second window, the learner was to apply the steps just ahead, having access to a training system. The pages were not to contain too much information. The learner would then proceed from page to page.

The participant wanted to have “next” and “previous” buttons as well as a navigation overview. The media to be used were text, images, and movies.

As an advantage of e-learning, she stated that the learner could pause whenever he wanted. However, she suggested that the learners should be offered a opportunity to quickly get answers to their questions.

Content Chunking

Having a tight work schedule, the participant did not have enough time to develop the tutorial “Silva for Web Authors”, as discussed in *Learning Unit Analysis* and *Learning Unit Concept* phase. Instead, she had created another learning unit, called “Changes of Silva 1.0 Compared to Version 0.9”. This learning unit was to point out the functional differences between the two software versions. The target learner group for this learning unit consisted of Silva users, who already had learned how to use the 0.9 version of Silva.

Further, she could not find the chunking guidelines. Accordingly, she just started to develop her learning unit, and modularized the content using her own method. No evaluation results regarding the proposed content chunking method could be collected in this case.

Results of the semi-structured interview

IQ1: She used her own method because she could not find the chunking guidelines.

IQ2: She did not develop what she originally had planned (see above). The learning unit “Changes of Silva 1.0 Compared with Version 0.9” was only to point out the functional differences of the two software versions. She said that e-learning would suit this subject ideally.

IQ3: She rated the subject matter as well-suited for modularization.

IQ4: She reported that she arranged the contents differently, like in an encyclopedia.

IQ5: The learning units would not change as she had already used modularization for classroom courses in the past.

IQ6: She stated that modularization was pleasing. It was just “part of the game”. Former participants of her classroom trainings had been missing a “clear thread”. In the meantime, she took great care to provide such a clear thread. Modularization would greatly help the learners to perceive this thread, but it would also a help her to create this thread. She indicated that in this sense, modularization yields a “useful categorization of contents”.

IQ7: Modularization would pay-off, because it provided a “useful categorization of contents”’.

Content Development and Learning Unit Assembly

Content Development

The participant developed seven Content Elements from scratch. Each of these Content Elements integrated animated screenshots pointing out what is new in the user interface (see Figure 6.3). She used Macromedia RoboDemo 5 to create the animations. One Content Element, describing the functions of the new online editor Kupu, was created by copying the contents from another Silva site into the dLCMS. This Content Element was based on text and images. Another Content Element for the new editor contained a short text with a hypertext link pointing to an external movie. The movie was a demo of the online editor with comments by a speaker.

The participant noted the following problems which occurred during the development of the participants learning unit:

- The participant looked up the dLCMS tutorial, which was available in the system. However, the tutorial was out-of-date. The terminology and the icons of the system had changed since the tutorial had been created.
- The participant asked how to proceed and where to store the Content Elements. The coach offered her two possibilities: to use a section of the existing repository, which already contained some Silva tutorial Content Elements, or to create her own repository. She decided to create her own repository. If the Content Elements were to be integrated into the existing repository, this could be done later.
- The next question was whether a subject folder should be created for each Content Element with its images and movie files. She already had created a repository structure containing four sections (authors, editors, chief editor, site manager). The authors’ section included a sub-section for the new

online editor. It further contained several Content Elements and Flash movies. Assuming that Content Elements had to be copied to a lesson, she was unsure if her structure, with subject folders containing several movies, was suitable. The coach answered that the subject folders were intended to subdivide the subject matter into separate topics, and that her approach was appropriate. He explained that the dLMCS does not copy the Content Elements into a lesson but that they are integrated into the lesson through links instead.

- She wanted to integrate animated screenshots, but she was not sure if this was possible and how to do it. The coach explained to her how to do this.

Learning Unit Assembly

The following problems were related to the *Learning Unit Assembly:*

- The participant asked why the lesson's top node needed to be assigned to a Content Element. She wanted to have three themes (authors, editors, site managers) as subchapters and the top node was simply to contain a table of contents, from which the learner could freely select a subject. The coach proposed to create a Content Element containing a table of contents for the lesson top node.
- She could not create an automatically generated table of contents (AutoTOC), a feature of the Silva system to which she was referring (the AutoTOC feature had been disabled for the dLCMS, because it was considered to be content which was not self-contained). The coach advised her to create a Content Elements and showed her a work-around to insert an AutoTOC all the same.
- After assembling the Content Elements to a Learning Unit, the participant noticed that the links to the Content Elements were broken. Each Content Element had to be reassigned to the Learning Unit. It was not clear what had happened in detail, but it is assumed that this problem was caused by a reorganization of the repository.
- Taking a look at the Learning Unit, she observed that images were not displayed, which had been previously rendered correctly. In discussing this issue with the coach, it became clear that the Content Elements contained faulty image paths. Again, it was assumed that this problem was caused by a reorganization of the contents in the repository.
- The participant suggested that Leaning Units should be able to contain other Learning Units as well as single Content Elements. At the moment, only Content Elements could be added to Learning Units.
- She wanted to publish the Learning Unit directly on the dLCMS system, but the Learning Unit could not be published without giving learners access to

the repository (the system was designed to export lessons to another LMS or a static web server in order to be published). Learners were not to be given access to the repository.

Results of the semi-structured interview

IQ8: The participant stated that not all procedures were intuitive. She frequently needed to look up the dLCMS tutorial. She said that she learned to use the dLCMS quickly because of the Silva expertise she already had. She suggested that the dLCMS tutorial provide some movies to visualize the basic procedures. Still, she judged the system to be very handy and much better than WebCT in respect to how contents were compiled.

IQ9: The participant had some problems operating the dLCMS, but all problems were solved by the e-mail support. She suggested that a FAQ for authors would be very helpful.

IQ10: She said that she did not use markup elements having implemented mainly movies (the animated screenshots). Therefore, the questions *IQ11* to *IQ13* were not answered.

IQ14: The system should provide better support to adjust the reference paths if images were transferred to the system.

Analysis of the Final Content Elements

At the end, 13 Content Elements had been created. With the exception of CE3.7, which referred to an external resource and was created by another author, all elements were self-contained.

Four elements were table of contents (TOC) and were not taken into consideration for the further analysis (CE3.1, CE3.2, CE3.7, CE3.12). They provided an overview of the Learning Unit and contained links to the other Content Elements. By its nature, a TOC references other elements and is therefore neither self-contained nor relating to a single objective. Therefore, a Content Element consisting solely of a TOC was not analyzed any further. In Table 8.3, “TOC” is specified as didactic content type, and “N/A” (“not applicable”) is used in the other fields.

All remaining Content Elements, with the exception of CE 3.7 (see below), were self-contained.

Most of the Content Elements were designed as animated screenshots with balloons commenting on the changes of the new Silva user interface version. Two of these Content Elements specified the changes of a particular user interface view (CE3.3, CE3.9). Their didactic content type was rated as *expository* and the single objective was rated as *fact*. The rest of the animated screenshots described the new operating procedures (CE3.4, CE3.5, CE3.10, CE3.11, CE3.13). Three of them were considered to contain two procedures and were therefore classified to be related to approximately more than a single objective (CE3.4, CE3.11, CE3.13). Interestingly, CE3.11 was clearly divided into two parts, both of which had a full screen title.

Content Element CE3.7 contained only a link to an external animation with speech and was therefore not rated to be self-contained. Instead of the Content Element, we rated the external resource as an *example* of the operating *procedure*, thus representing a single didactic content type relating to a single objective.

Content Element CE3.8 contained expository text with images to describe the operation of the Kupu editor, which the participant had copied as a single page from some other source. It described the procedures of 11 sub-tasks, therefore this Content Element was not rated, as relating to a single objective.

Table 8.3 tabularizes the results of the analysis of the Content Elements created by Author 3.

Table 8.3 Analysis of the Content Elements created by author 3.

	Self-contained	Didactic content type	Single didactic content type	Content Categories	Single objective
CE3.1	N/A[1]	TOC[1]	N/A[1]	N/A[1]	N/A[1]
CE3.2	N/A[1]	TOC[1]	N/A[1]	N/A[1]	N/A[1]
CE3.3	Yes	Expository	Yes	1 Fact	Yes
CE3.4	Yes	Instructions	Yes	2 Procedures	No
CE3.5	Yes	Instructions	Yes	1 Procedure	Yes
CE3.6	N/A[1]	TOC[1]	N/A[1]	N/A[1]	N/A[1]
CE3.7	No[2]	1 Example	Yes	1 Procedure	Yes
CE3.8	Yes	Instructions	Yes	11 Procedures	No
CE3.9	Yes	Expository	Yes	1 Fact	Yes
CE3.10	Yes	Instructions	Yes	1 Procedure	Yes
CE3.11	Yes	Instructions	Yes	2 Procedures	No
CE3.12	N/A[1]	TOC[1]	N/A[1]	N/A[1]	N/A[1]
CE3.13	Yes	Expository, Instructions	No	2 Procedures	No

[1] Some Content Elements were used as table of contents (TOC) with links to other Content Elements.

[2] The Content Element contained a link to an external animation with speech. The results of the other columns apply to this external resource.

Author 4

Learning Unit Analysis

Subject matter

The subject matter of the learning unit was an “Introduction to Usability Evaluation”, and as the major learning objective, the students were “to be able to conduct a simple usability evaluation of a system for a specified user and task”. Generally, usability is taught in the context of “user centered design”. The experiences of previous usability lessons of this post-graduate study suggested that the students were not very interested in design methodologies. This was explained by the fact that most students were working in the field of occupational medicine and occupational hygiene. Therefore, the actual learning unit was to focus on usability evaluation only.

The learning unit was designed to be a part of a blended learning course consisting of an online learning unit and classroom teaching. The online learning unit was to be a self-study lesson. In the classroom part, the students were to conduct a usability test working in teams together with a usability expert. The expected learning time of the online learning unit was to be approximately 60 minutes. The entire usability evaluation course was part of a two-day module, two-thirds of which had to be attended in order to obtain credit points.

The participant rated the subject matter to be suitable for modularization. He justified this rating stating that the subject matter could easily be chunked into usability concepts, usability principles, and usability procedures, which were self-contained and therefore reusable in various educational contexts.

He also rated the subject matter to be suitable for e-learning, because computer interfaces could easily be simulated in an e-learning unit.

The learning unit was to be designed from scratch in order to create a lesson which is well adapted for the students. The learning material at hand consisted of lecture notes, PowerPoint presentations, screenshots of user interfaces, and other images.

Learner profile

The course was to be part of a postgraduate professional study on ergonomics. The students were supposed to be natural scientists, engineers, and physicians. No specific knowledge was required.

The age of the students was supposed to be in the range from 30 to 60 years. It was anticipated that all students would speak German. Approximately 20% would have French as their first language. The ratio of women and men was expected to be approximately 40% women and 60% men.

Most of the students were anticipated to be highly motivated. This was justified by the fact that they had to evaluate work places with computers and other machines in their daily jobs. Further, they were all computer users. A minority attended the course for the certification as an ergonomic expert only. An academic degree was a prerequisite to attend the post-graduate study, therefore all students were anticipated to be used to learning on their own.

All students were expected to have computer experiences as users of office applications, e-mail, and web browsing. It was assumed that only few would have had previous e-learning experiences. It was expected that all students had access to the internet and modern browsers (Internet Explorer 6 or Mozilla 1.5).

Learning Unit Concept

Task Analysis

Together with an expert, the participant analyzed the task of a usability evaluation. The following steps were identified: (1) user analysis, (2) task analysis, (3) specification of the usability attributes to be tested based on the expected users and tasks, (4) specification of the usability goals including measurement methods and acceptable performance levels, (5) selection of an appropriate usability evaluation method, (6) conducting the usability evaluation.

Learning Objectives

Based on the task analysis, the subsequent learning objectives were specified. The learners were to: (1) be able to conduct a simple user and task analysis, (2) be aware of the ISO 9241-10 usability attributes, (3) be able to understand the meaning of the ISO 9241-10 usability attributes, (4) know that there are also other usability attributes, (5) be aware that there is no standard way to operationalize the usability attributes, (6) know important operationalizations, such as time to complete a task, number of clicks, number of user errors, and subjective rating of users, (6) have a general overview of usability evaluation methods, (7) be able to perform a heuristic evaluation, (8) know what a usability test is and be aware of its importance.

Specification of the Learning Content

The participant then compiled a list of contents that he wanted to use in order to teach the learning objectives:

- Anesthesia apparatus as an example of the importance of usability
- Usability definition of ISO 9241-11: effectiveness, efficiency, satisfaction
- General overview of the usability evaluation procedure
- User and task analysis
- Usability attributes of ISO 9241-10
- Operationalizations of usability attributes
- Usability evaluation method overview

- Example 1: Heuristic evaluation
- Example 2: Usability testing
- Exercise of a heuristic evaluation.

Didactical strategies and methods

At the beginning the participant intended to create a learning unit using the “problem based learning” method. Taking a closer look at the contents, it became clear that problem-oriented learning would not suit a 60-minute learning unit which covered such a broad range of topics. Taking into account that the students would perform a practical usability test in the classroom part, he decided to use a concise expository design. The contents were to be arranged in a chapter-like manner, allowing the learner to follow a pre-defined sequence of contents. However, the students should also be able to freely navigate through the contents.

The media to be used were text and images. The participant planned to integrate an online exercise, a heuristic evaluation of a user interface, at the end of the learning unit. In order to support effective learning, it became clear that the students would not simply have to rate the user interface, but would also have to explain their ratings. This could not easily be implemented using multiple-choice type questionnaires and it seemed that the expected duration of the learning unit did not provide enough time for more elaborate learning activities. Nevertheless, the participant wanted to provide some interactive learning activities and decided to integrate several multiple-choice type self-assessments instead.

Content Chunking

The participant performed the *Content Chunking* according to the procedure described in the chunking guidelines.

Breaking contents down into single objectives

The participant added the titles of the topics and subtopics to the first column of the chunking spreadsheet. In the second column, he tried to assign the content categories *concept, fact, procedure, process,* and *principle* to the topics. During this phase he noticed the following problems:

- The participant wanted the content hierarchy in the chunking spreadsheet to be clearly visible. First he tried to add a new column with numbers indicating the hierarchy level, but this did not noticeably point out the structure. Then he added a numbering scheme with a separate number for every sub-level of topics (e.g. “2. Usability Definition”, “2.1 Effectiveness”). This helped to visualize the hierarchy, but after the rearrangement of Content Elements the numbering had to be redone manually.
- He could not figure out which content category should be assigned to an overview of usability evaluation methods. On the one hand, the content

category *procedure* seemed to match because evaluation methods are sequential steps to be followed to accomplish a task. On the other hand, the overview itself did not exactly contain a description of a sequence of steps, but rather a group of names for different methods, all of which can be used for a usability evaluation and can be summarized by the title “usability evaluation methods”. This suggested using the content category *concept*. In the end, he chose the content category *procedure*.

- He encountered another problem assigning a content category to each of the “dialog principles” of ISO 9241-10 (e.g. “self-descriptiveness”, “controllability”, etc.). The Content Elements should contain a definition. Therefore, he considered the Content Element to be a *concept*, but in his opinion, they could also be classified as a *principle* because they contained information to judge the usability of a product. The title of the ISO standard, “Dialog Principles”, emphasized using *principle* as well. In the end, he used the content category *principle* for these elements.

Specifying the didactic content types

In the next step, the participant went through all of the topics and assigned a didactic content type from the list in the guidelines to the topics and subtopics. He added new lines just underneath 12 of the 37 items in order to add a second didactic content type to that item. Nine added lines were labeled “example”. “Expository”, “exercise”, and “instruction” were used for the other three lines.

Two Content Elements, “usability evaluation methods” and “usability-testing method and procedure”, were labeled “overview” instead of using a didactic content type given by the guidelines.

Analysis for reuse with other learner groups

The participant identified four other potential learner groups next to the actual target learner group. These learner groups were “usability experts”, “web designers”, “programmers”, “managers”. He created a new column in the chunking spreadsheet for each of these learner groups.

In this step, he tried to analyze if the Content Elements, defined by each line of the spreadsheet, could be reused for the potential learner groups identified.

The participant had difficulties deciding whether or not Content Elements could be reused for other learner groups. For example, he could not figure out what managers, one of the potential learner groups, needed to know. This would depend on the detailed learning objectives for his learner group. The learning objectives could only be obtained through a learning unit analysis, which he did not think was his job.

In the end, this analysis did not yield any new Content Element.

Results of the semi-structured interview

IQ1: He used the proposed method to modularize the contents.

IQ2: He rated the subject matter to be 60% suitable for e-learning. He explained that although the subject matter covers computers interfaces, perception and behavior of humans play an important role. Still, human behavior, with respect to human computer interaction, cannot be easily modeled and simulated, which made it difficult to develop interactive online learning activities. Further, he stated that a usability evaluation would take into account multiple factors, each of which might yield contradicting results. Therefore, experts estimating these factors may draw different conclusions. This makes it hard to test the learner's ability of usability judgment using multiple-choice type questionnaires.

IQ3: He rated the suitability of the subject matter to be 90%, stating that the topic could easily be divided into concepts and principles.

IQ4: He said that modularization did not change the way he was thinking of the learning unit.

IQ5: In his opinion, chunking resulted in a modular and very structured learning unit.

IQ6: (1) He stated that the chunking spreadsheet was an aid to refine the planning step-by-step. (2) He said that it was difficult to decide whether Content Elements could be reused for other learner groups without knowing what the learning objectives for the group would be. His job was the development of a learning unit for ergonomists. Therefore, it was not his job to perform a learning unit analysis for other learner groups in order to obtain detailed learning objectives. (3) In his opinion, creating self-contained elements needed an extra effort, because the structure had to be thought out well.

IQ7: Modularization would pay-off as contents needed to be structured anyway, for e-learning as well as for lecture notes. Modularization would help to plan the development of a learning unit.

Content Development and Learning Unit Assembly

Content Development

The participant developed about half of the resulting 58 Content Elements from scratch. About one-third of them had to be created by copy and pasting small pieces of text from a Word document containing lecture notes. Then, the texts pasted into the Content Elements had been intensively reedited and rearranged. Seven Content Elements had been created copying some paragraphs and images from the web without much modification.

The participant noted the following problems during the learning unit development:

- The participant wanted definitions to be displayed in separate boxes. No special markup element was available for definitions, but there was a markup element for citations, which displayed the contents inside yellow boxes. Therefore, he decided to misuse the citations element to markup definitions.
- Based on the chunking structure, the participant had worked in the Content Chunking phase, he created several content element containing only a definition (e.g. the ISO 9241-11 definition of "effectiveness", see Figure 6.4). Looking at the result, he wanted the page to present more information than just a single definition. This information could be an explanation or an example.
- In one Content Element, the participant added a short textual example between two paragraphs of exploratory text. The example was to be clearly distinguishable from the other text. The participant used italic script to markup the example, but he stated that there should be a better way to markup examples.
- Bibliographic information should be added to the Content Elements. This information should be clearly distinguishable from the rest of the text. There was no markup for this type of information, therefore the participant decided to use the annotation element for bibliographic information. He further adjusted the styling of the annotation element using the system's style sheets (CSS) in order to display the information in a grey box and to indent all lines but the first line of a paragraph (see Figure 6.4).
- The dLCMS Questionnaire Content Element for multiple-choice questions did not suit the participants' needs. He explained that the buttons were labeled in English instead of German. He further did not want to use the "hint" button, which could not be omitted easily. Therefore, he developed raw HTML and JavaScript code, which he integrated into Content Elements. He considered this approach to be a hack.
- The participant wanted to add headers to each column of different tables. The system offered a possibility to add headers to the whole table, but no headers for single columns were provided. Normal paragraph headers could be assigned to text inside table cells, but the participant judged that the spacing atop of the paragraph header was too large to be used in table cells. In the end, he used italic script, but he suggested that there should be special markup for column and probably also row headers.
- In one case, the participant wanted to add a table caption. No special markup was available for this, therefore he used italic script. He suggested that there should be special markup for table captions.

- The participant noticed that the repository sidebar navigation tree to the left becomes very long as new content and subject folders are added. In order to navigate to items at the end of the navigation tree, the participant had to scroll the page. He considered this to be awkward, because he frequently switched between Content Elements.
- The participant was wondering if he should also store the originals used to create Assets in the repository. For example, he created a figure using Microsoft PowerPoint. Then, he converted the figure into a bitmapped image and uploaded it to the system. He reasoned that the PowerPoint original needed to be uploaded as well if other authors would want to edit the image.

Learning Unit Assembly

The participant started to assemble the Learning Unit in parallel to the development of the Content Elements. In order to see the evolving lesson, he frequently switched back and fourth between the repository, where Content Elements were edited, and the Learning Unit view, where he immediately added the new Content Elements to the Learning Unit.

Results of the semi-structured interview

IQ8: (1) The participant stated that the dLCMS made it easy to create learning material with the help of pre-defined formats and layouts. (2) The system enabled him to easily change the layout and the navigation of a whole Learning Unit later. (3) The metadata and search system were not mature yet. (4) The participant had to jump between pages very often, editing contents, assembling Content Elements, correcting and adapting the contents, publishing everything, exporting it to an LMS, going back to correct and adapt things, etc..

IQ9: In his opinion, the most difficult aspect was to find the optimal size of Content Elements.

IQ10: He reported that he could do what he wanted to do, but he “misused” some of the markup. He used the citation element for definitions, the annotation element to set off bibliographic information, and italic script for examples. He integrated multiple-choice self-assessments using raw HTML and JavaScript code, which he considered to be a hack. Further, it was not clear how to set-off titles in the top row of tables and how to markup table captions.

IQ11: He did not think that his creativity was constrained.

IQ12: He stated that the set of markup elements was not mature yet, referring to the answer above.

IQ13: Same answer as above.

IQ14: He stated that the following issues should be solved: (1) It should be easier to organize the contents in the system, (2) the multiple-choice Questionnaire Content Element needs to be improved, (3) there should be a system for bibliographic information which would automatically generate bibliographies where needed, (4) it should be possible to combine several Content Elements onto one page, (5) there should be a way to include copyrighted material which would be hidden from other authors.

Analysis of the Final Content Elements

Fifty-eight Content Elements had been created by Author 4. The detailed results of the Content Element analysis are presented in Table 8.4.

Two Content Elements were created as an overview (CE4.1, CE4.40) and thus referred to Contents Clements contained in the Learning Unit. These were the only elements which were not rated to be self-contained. For these elements, none of the didactic content types or content categories seemed to be appropriate. Therefore, the category “overview” was assigned as didactic content type and content category.

Eighteen Content Elements did not represent a single didactic content type. Most of these elements contained a definition and either an expository providing an explanation or an example illustrating the definitions. One Content Element contained a very short expository and an example.

Fifty-three Content Elements related to a single objective. The Content Elements that were not related to a single objective were:

- The overview elements (see above).
- An element containing a classification of concepts, which were shortly described in the Content Element, thus containing several concepts (CE4.12).
- A rather long Content Element containing a list of ten principles (4.49). The developer was aware of not creating a Content Element related to a single objective, but he did not want to have ten additional Content Elements containing only a title with one or two sentences.
- A Content Element containing literature references (CE4.58).

Table 8.4 Analysis of the Content Elements created by author 4.

	Self-contained	Didactic content type	Single didactic content type	Content Categories	Single objective
CE4.1	No	Overview	Yes	Overview	No
CE4.2	Yes	1 Definition, Expository	No	1 Concept	Yes
CE4.3	Yes	1 Definition, Expository	No	1 Concept	Yes

	Self-contained	Didactic content type	Single didactic content type	Content Categories	Single objective
CE4.4	Yes	Expository	Yes	1 Concept	Yes
CE4.5	Yes	1 Definition, Expository	No	1 Concept	Yes
CE4.6	Yes	1 Definition, Expository	No	1 Concept	Yes
CE4.7	Yes	1 Definition, Expository	No	1 Concept	Yes
CE4.8	Yes	1 Definition, Expository	No	1 Concept	Yes
CE4.9	Yes	Self-assessment	Yes	1 Concept	Yes
CE4.10	Yes	Expository	Yes	1 Procedure	Yes
CE4.11	Yes	Expository	Yes	1 Concept	Yes
CE4.12	Yes	Expository	Yes	2 Concepts	No
CE4.13	Yes	Expository	Yes	1 Concept	Yes
CE4.14	Yes	Self-assessment	Yes	1 Concept	Yes
CE4.15	Yes	Expository	Yes	1 Concept	Yes
CE4.16	Yes	Expository	Yes	1 Concept	Yes
CE4.17	Yes	Expository, 1 Example	No	1 Concept	Yes
CE4.18	Yes	Expository	Yes	1 Concept	Yes
CE4.19	Yes	Expository	Yes	1 Concept	Yes
CE4.20	Yes	Expository	Yes	1 Procedure	Yes
CE4.21	Yes	Self-assessment	Yes	1 Concept	Yes
CE4.22	Yes	Expository, 1 Example	No	1 Concept	Yes
CE4.23	Yes	Expository	No	1 Fact	Yes
CE4.24	Yes	1 Definition, 1 Example	No	1 Principle	Yes
CE4.25	Yes	Example	Yes	1 Principle	Yes
CE4.26	Yes	1 Definition, 1 Example	No	1 Principle	Yes
CE4.27	Yes	1 Example	Yes	1 Principle	Yes
CE4.28	Yes	1 Definition, 1 Example	No	1 Principle	Yes
CE4.29	Yes	1 Example	Yes	1 Principle	Yes
CE4.30	Yes	1 Definition, 1 Example	No	1 Principle	Yes
CE4.31	Yes	2 Examples	No	1 Principle	Yes
CE4.32	Yes	1 Definition, 1 Example	No	1 Principle	Yes
CE4.33	Yes	1 Example	Yes	1 Principle	Yes

	Self-contained	Didactic content type	Single didactic content type	Content Categories	Single objective
CE4.34	Yes	1 Definition, 1 Example	No	1 Principle	Yes
CE4.35	Yes	Example	Yes	1 Principle	Yes
CE4.36	Yes	1 Definition, 1 Example	No	1 Principle	Yes
CE4.37	Yes	1 Example	Yes	1 Principle	Yes
CE4.38	Yes	Self-assessment	Yes	1 Concept	Yes
CE4.39	Yes	Expository, 1 Example	No	1 Concept	Yes
CE4.40	Yes	Overview	Yes	1 Concept	Yes
CE4.41	Yes	Expository	Yes	1 Concept	Yes
CE4.42	Yes	Expository	Yes	1 Concept	Yes
CE4.43	Yes	Expository	Yes	1 Concept	Yes
CE4.44	Yes	Expository	Yes	1 Concept	Yes
CE4.45	Yes	Self-Assessment	Yes	1 Concept	Yes
CE4.46	No	Overview	Yes	1 Overview	No
CE4.47	Yes	Expository	Yes	1 Procedure	Yes
CE4.48	Yes	Expository	Yes	1 Principle	Yes
CE4.49	Yes	Expository	Yes	10 Principles	No
CE4.50	Yes	Expository	Yes	1 Concept	Yes
CE4.51	Yes	Self-assessment	Yes	1 Concept	Yes
CE4.52	Yes	Expository	Yes	1 Procedure	Yes
CE4.53	Yes	Expository	Yes	1 Concept	Yes
CE4.54	Yes	Exploratory	Yes	1 Procedure	Yes
CE4.55	Yes	Exploratory	Yes	1 Procedure	Yes
CE4.56	Yes	Exploratory	Yes	1 Procedure	Yes
CE4.57	Yes	Self-Assessment	Yes	1 Procedure	Yes
CE4.58	Yes	Literature	Yes	Literature	No

Appendix C: Content Element Markup Schema

The structured markup schema will be described based on the elements and the element attributes. The specification of the elements relies on the Document Type Definition (DTD) syntax (W3C 2004a). This syntax describes the name, number, and order of other elements to be contained by an element. The attributes of the elements are listed in tables right underneath the element specification. The table contains a column for the *attribute name*, the *attribute type*, and a description of the attribute. Three attributes types are used here: *CDATA* (a string of characters), *Enumeration* (a list of names which may not contain spaces or punctuation), and *NUMBER* (a decimal number).

The specification of the elements is grouped as follows: *root elements*, *headings*, *paragraph elements*, *lists*, *tables*, *images*, *multimedia elements*, *basic inline elements*, *links*, *index items*. The descriptions of the elements can be found in the Section 4.2.2 “Content Element Markup Schema”)

Root Elements

In the structured markup schema for *Content Elements*, the root node is specified as `ContentElement`.

```
ContentElement ::= (Title, DocumentContent)

Title ::= (#PCDATA)

DocumentContent ::= ((Heading | SubHeading |
        ParagraphHeading | SubParagraphHeading |
        LeadParagraph | Paragraph | Annotation |
        Preformatted | Citation | List | NestedList |
        DefinitionList | Table | Image | ExternalSource)*)
```

Headings

```
Heading, SubHeading, ParagraphHeading, SubParagraphHeading
        ::= ((#PCDATA | Emphasis | Superscript | Subscript |
        IndexItem)*)
```

Paragraph Elements

```
LeadParagraph, Paragraph, Annotation ::= ((#PCDATA | Strong |
        Emphasis | Underline | Superscript | Subscript |
        Link | IndexItem)*)
```

```
Preformatted ::= (#PCDATA)

Citation ::= (Author, Source, Paragraph+)

Author ::= (#PCDATA)

Source ::= (#PCDATA)
```

All paragraph elements have no attributes.

Lists

```
List ::= (ListItem*)
```

Attribute name	Attribute type	Description
ListType	*Enumeration:* disc circle square 1 i I a A	Defines the type of the list being either an unordered list with bullets, circles, or squares, or a numbered list using Arabic numbers (1, 2, 3), roman numbers (i, ii, iii / I, II, III), or lower- or uppercase letters (a, b, c / A, B, C).

```
ListItem ::= ((#PCDATA | Strong | Emphasis | Underline |
       Superscript | Subscript | Link | IndexItem)*)

NestedList ::= (NestedListItem*)
```

The NestedList element has the same ListType attribute as the List element.

```
NestedListItem := ((Heading | SubHeading | ParagraphHeading |
       SubParagraphHeading | LeadParagraph | Paragraph |
       Annotation | Preformatted | List | NestedList |
       DefinitionList | Table | Image)*)

DefinitionList ::= ((DefinitionTerm, DefinitionDescription)+)
```

Attribute name	Attribute type	Description
Type	*Enumerated:* normal compact	Two basic styles are defined for definition lists: *normal* – normal spacing between the list items. *compact* – reduces spacing between items.

```
DefinitionTerm ::= ((#PCDATA | Strong | Emphasis | Underline
       | Superscript | Subscript | Link | IndexItem)*)

DefinitionDescription ::= ((#PCDATA | Strong | Emphasis |
       Underline | Superscript | Subscript | Link |
       IndexItem)*)
```

Tables

```
Table ::= ((RowHeading | Row)*)
```

Attribute name	Attribute type	Description
`Type`	*Enumerated:* plain listing grid datagrid	Four basic table styles are defined: *plain* – bold heading, no borders. *listing* – bold underlined heading, borderless rows. *grid* – bold bordered heading, bordered rows *datagrid* – bold borderless heading, bordered rows.
`Columns`	NUMBER	Number of columns.
`ColumnInfo`	CDATA	The column information attribute defines horizontal alignment (right, center, left) and the relative width of all columns.

```
RowHeading ::= (#PCDATA)

Row ::= (Field*)

Field ::= ((Heading | SubHeading | ParagraphHeading |
        SubParagraphHeading | LeadParagraph | Paragraph |
        Annotation | Preformatted | List | NestedList |
        DefinitionList | Image)*)
```

Images

```
Image ::= EMPTY
```

Attribute name	Attribute type	Description
`Path`	CDATA	The path to the image.
`Alignment`	*Enumerated:* left center right float-left float-right	This attribute describes the alignment of the image with respect to its context.
`Link`	CDATA	A URI specifying the location of the target resource if the image should be clickable as a hypertext link.

Multimedia Elements

The `ExternalSource` element provides a mechanism to include types of data which were not foreseen when the structured markup scheme was specified. These elements are tightly coupled to the functions needed for data processing and presentation. The `ID` attribute refers to a server-sided program scripts, which will process the data contained in the `Parameter` child elements. The set of parameters needed are defined by the different program scripts.

```
ExternalSource ::= (Parameter*)
```

Attribute name	Attribute type	Description
ID	CDATA	The path to the external source handler.

```
Parameter ::= (#PCDATA)
```

Attribute name	Attribute type	Description
Key	CDATA	The name of the parameter.

Basic Inline Elements

```
Emphasis ::= ((#PCDATA | Strong | Emphasis | Underline |
        Superscript | Subscript | Link | IndexItem)*)
Strong ::= ((#PCDATA | Strong | Emphasis | Underline |
        Superscript | Subscript | Link | IndexItem)*)
Underline ::= ((#PCDATA | Strong | Emphasis | Underline |
        Superscript | Subscript | Link | IndexItem)*)
Superscript ::= ((#PCDATA | Strong | Emphasis | Underline |
        Superscript | Subscript | Link | IndexItem)*)
Subscript ::= ((#PCDATA | Strong | Emphasis | Underline |
        Superscript | Subscript | Link | IndexItem)*)
```

Links

Links should not be nested, i.e. no further `Link` element should be used by child elements at any depth inside a link. Unfortunately this requirement is difficult to be specified in a DTD.

```
Link ::= (#PCDATA| Strong | Emphasis | Underline |
        Superscript | Subscript)
```

Attribute name	Attribute type	Description
URL	CDATA	A URI specifying the location of the target resource of this link.
Target	CDATA	This attribute is optional and specifies the browser window, in which the resource will be opened.

Index Items

```
IndexItem ::= EMPTY
```

Attribute name	Attribute type	Description
`Name`	CDATA	The normalized index term.

Appendix D: dLCMS Metadata Set

This section contains the detailed metadata set for the dLCMS project. It is designed to be compatible with the mandatory elements specified in by the SCORM metadata information model (ADL 2001a) and the ARIADNE Educational Metadata Recommendation (ARIADNE 2002). Both the SCORM and the ARIADNE metadata models are based on the IEEE LTSC Learning Object Metadata (LOM) standard (IEEE LTSC 2002).

The metadata elements are presented in the same order as provided by the LOM standard and is divided accordingly into the sections: *general*, *life cycle*, *meta-metadata*, *technical*, *educational*, *rights*, *classification*. Additional elements, which are not contained in the LOM specification, are added at the end of the appropriate section, e.g. the dLCMS *content category* is at the end of the "educational" section.

In the following table, each row describes a metadata element by:

1. *Metadata Element:* This is the dLCMS name for the metadata element.
2. *Description:* A short description of the element.
3. *Data Input:* Contains the data type of the element if the data has to be entered by the user. "System" signifies that the metadata can be automatically generated.
4. *Example / Comments:* Data examples are shown in italics. Further comments are provided in regular script.
5. *LOM:* The mapping of the metadata element to LOM (IEEE LTSC 2002).
6. *ARIADNE:* The mapping of the metadata element to the ARIADNE Educational Metadata Recommendation (ARIADNE 2002).
7. *Dublin Core:* The mapping of the metadata element to the Dublin Core metadata set (DCMI 2004) based on the mapping specifications of LOM and ARIADNE.

The current dLCMS implementation does not support multiple or multilingual metadata entries.

Metadata Element	Description	Data Input	Example / Comments	LOM	ARIADNE	Dublin Core
General						
Identifier [1,2]	A globally unique label that identifies this learning object.[3]	System		1.3.2:General.CatalogEntry.Entry, with 1.3.1:General.CatalogEntry.Catalog = 'Ariadne'	1.0 identifier	Identifier
Catalog [1]	The name or designator of the identification or cataloging scheme for this entry. A namespace scheme.[3]	System	*URI, ARIADNE, ISBN,* etc.	1.3.1:General.CatalogEntry.Catalog		
Title [1,2]	Name given to this learning object.[3]	Free text	*Introduction to Usability Evaluation*	1.2:General.Title	1.1 title	Title
Short title	A short title to be used for navigation.	Free text	*Usability Evaluation*			
Language [2]	The primary human language or languages used within this learning object to communicate to the intended user.[3]	Fixed list	*en, de-CH,* etc.	1.4:General.Language	1.4 language	Language
Description [1]	A textual description of the content of this learning object.[3]	Free text	*Methods for usability evaluation. In-depth treatment of heuristic evaluation and usability testing.*	1.5 Description		Description

Metadata Element	Description	Data Input	Example / Comments	LOM	ARIADNE	Dublin Core
Main concept [1,2]	The main concept that is covered by the educational resource.[4]	Free text	ARIADNE maintains an extendable list in their repository. This list is not available outside of the ARIADNE system.	1.6 Keyword and 9.2.2.2:Classification.TaxonPath. Taxon[4].Entry, with 9.1:Classification.Purpose = 'Discipline' and 9.2.1:Classification.TaxonPath.Source = 'Ariadne' and 9.2.2.2:Classification.TaxonPath. Taxon[1] = 2.1 discipline type and 9.2.2.2:Classification.TaxonPath. Taxon[2] = 2.2 discipline and 9.2.2.2:Classification.TaxonPath. Taxon[3] = 2.3 subdiscipline	2.4 main concept	Subject
Keywords	Keywords associated with this resource.	System	The Content Elements index items will by used as keywords.			
Life cycle						
Version [1]	The edition of this learning object.[3]	System	Determined by the versioning and publishing system	2.1:LifeCycle.Version		
Publication status [1]	The completion status or condition of this learning object.[3]	System	*Draft*, *final*, *revised*, *unavailable* (LOM)	2.2:LifeCycle.Status		
Creator [2]	The creators of the learning resource.	System (can be overwritten using free text)	By default the authenticated user who created the resource	2.3.2:LifeCycle.Contribute.Entity, with 2.3.1:LifeCycle.Contribute.Role = 'Author'	1.2 authors	Creator
Institution [2]	The publisher or university or corporation that the author was affiliated to when the educational resource was released or published.[4]	Free text	*Institute of Hygiene and Applied Physiology, ETH Zürich*	2.3.2:LifeCycle.Contribute.Entity with 2.3.1:LifeCycle.Contribute.Role = 'Publisher'	1.5 institution	Publisher

Metadata Element	Description	Data Input	Example / Comments	LOM	ARIADNE	Dublin Core
Publication date	The release or publication date.	System	See version	2.3.3:LifeCycle.Contribute.Date, with 2.3.1:LifeCycle.Contribute.Role = ‘Publisher’	1.3 date	Date
Meta-metadata						
Metadata identifier	A globally unique label that identifies this metadata record.[3]	System	The system stores metadata together with the contents, therefore metadata cannot be edited separately.	3.1:MetaMetaData.Identifier		
Metadata author [2]	The metadata author.	System (same as *creator*)	See above	3.3.2:MetaMetaData.Contribute.Entity, with 3.3.1:MetaMetaData.Contribute.Role = 'Creator'	6.1 author	
Metadata creation data [2]	Date of the creation of the metadata.	System (same as *publication date*)	See above	3.3.3:MetaMetaData.Contribute.Date, with 3.3.1:MetaMetaData.Contribute.Role = 'Creator', and 3.3.2:MetaMetaData.Contribute.Entity = Metadata author	6.2 creation date	
Metadata last modified [2]	Date of the last metadata modification.	System (same as *publication date*)	See above	Chronologically last 3.3.3:MetaMetaData.Contribute.Date	6.3 last modified date	
Metadata scheme [1]	The name and version of the authoritative specification used to create this metadata instance.[3]	System	Provided by the system	3.4:MetaMetaData.MetadataScheme		
Metadata language [2]	Language of this metadata instance.[3]	System (same as *language*)	See above	3.5:MetaMetaData.Language	6.4 language	

Metadata Element	Description	Data Input	Example / Comments	LOM	ARIADNE	Dublin Core
Technical						
Format [1,2]	Technical datatype(s) of (all the components of) this learning object. [3]	System	MIME types of the resources, e.g. *text/html, image/jpeg,* etc.	4.1:Technical.Format	4.2 file media types	Format
Size [2]	The size of the learning objects in bytes.[3]	System		4.2:Technical.Size	4.3 package size	
Location [1,2]	A string that is used to access this learning object.[3]	System	E.g. the resource URL	4.3:Technical.Location	4.1 document handle	
Operating system [2]	Name of the required technology to use this learning object.[3]	System (fixed to *Multi-OS*)	dLCMS is designed only for web-based resources.	4.4.2:Technical.Requirements.Name, with 4.4.1:Technical.Requirements.Type = 'Operating System'	4.4 operating system type	
Educational						
Interactivity type [2]	Predominant mode of learning supported by this learning object.[3]	Fixed list	*Active, expositive, mixed, undefined* (LOM)	5.1:Educational.InteractivityType	3.2 document type	Type
Resource type [2]	Specific kind of learning object.[3]	Fixed list	*Exercise, simulation, questionnaire,* etc. (LOM)	5.2:Educational.LearningResourceType	3.3 document format	Type
End user type [2]	Principal user(s) for which this learning object was designed.[3]	System (fixed to *learner*)	*Learner*	5.5:Educational.IntendedEndUserRole	3.1 end user type	
Context	The principal environment within which the learning and use of this learning object is intended to take place.[3]	Fixed list	*School, higher education, training, other* (LOM)	5.6:Educational.Context	3.4.2 context	
Learning time [2]	Approximate or typical time it takes to work with or through this learning object for the typical intended target audience.[3]	Integer	Learning time in minutes.	5.9:Educational.TypicalLearningTime	3.8 pedagogical duration	

Metadata Element	Description	Data Input	Example / Comments	LOM	ARIADNE	Dublin Core
Skill level	The level of the intended learners' previous knowledge on the topic.	Fixed list	*Beginner, advanced, expert* (dLCMS)	9.2.2.2:Classification.TaxonPath. Taxon[1].Entry, with 9.1:Classification.Purpose='Skill Level', and 9.2.1:Classification.TaxonPath.Source = 'dLCMS'		
Content category	The cognitive level of the contents.	Fixed list	*Concept, fact, procedure, principle, process* (dLCMS)	9.2.2.2:Classification.TaxonPath. Taxon[1].Entry, with 9.1:Classification.Purpose='Educational Objective', and 9.2.1:Classification.TaxonPath.Source = 'dLCMS'		
Didactic content type	The didactic content type of the Content Elements specifying the didactic intension of the learning resource.	Fixed list	*Expository, definition, narrative, instruction, example, excursion, glossary, literature, summary, learning objectives, advance organizer, exercise, questionnaire, exam, self-assessment, experiment, problem statement, simulation* (dLCMS)	9.2.2.2:Classification.TaxonPath. Taxon[2].Entry, with 9.1:Classification.Purpose='Educational Objective', and 9.2.1:Classification.TaxonPath.Source = 'dLCMS', and 9.2.2.2:Classification.TaxonPath. Taxon[1] = content category		
Media	The media by which the content are presented.	Fixed list	*Text, formula, table, chart, diagram, map, image, animation, video, oral text, music, sounds, 3D-model* (dLCMS)			
Rights						
Cost [1,2]	Whether use of this learning object requires payment.[3]	Fixed list	*Yes, no* (LOM)	6.1:Rights.Cost	5.1 access rights	Rights

Metadata Element	Description	Data Input	Example / Comments	LOM	ARIADNE	Dublin Core
Restrictions [1,2]	Whether copyright or other restrictions apply to the use of this learning object.[3]	Fixed list	*Yes, no* (LOM)	6.2:Rights. CopyrightAndOtherRestrictions	5.2 restrictions	Rights
Rights [1,2]	Comments on the conditions of use of this learning object.[3]	Free text	*Copyright 2004 by ETH Zürich. Except where otherwise noted, this resource is licensed under a Creative Commons License. To view a copy of this license, visit [link to license].*	6.3:Rights.Description	5.3 usage remarks	
Classification						
Discipline type [2]	One of the two wide groups of knowledge disciplines that are generally differentiated.[4]	System (based on *discipline*)	*Human or Social Sciences, Natural, Exact or Engineering Sciences* (ARIADNE)	9.2.2.2:Classification.TaxonPath. Taxon[1].Entry, with 9.1:Classification.Purpose='Discipline', and 9.2.1:Classification.TaxonPath.Source = 'Ariadne'	2.1 discipline type	Subject
Discipline [2]	The knowledge field in the context of which the learning or teaching is to take place.[4]	Fixed list	*Biology/Life Sciences, Chemisty/Bio-Chemistry,* etc. (ARIADNE)	9.2.2.2:Classification.TaxonPath. Taxon[2].Entry, with 9.1:Classification.Purpose = 'Discipline' and 9.2.1:Classification.TaxonPath. Source = 'Ariadne' and 9.2.2.2:Classification.TaxonPath. Taxon[1] = Discipline type	2.2 discipline	Subject

Metadata Element	Description	Data Input	Example / Comments	LOM	ARIADNE	Dublin Core
Subdiscipline [2]	The more specific knowledge sub-field in the context of which the learning or teaching is to take place.[4]	Free text	ARIADNE maintains an extendable list of subdisciplines (ARIADNE).	9.2.2.2:Classification.TaxonPath. Taxon[3].Entry, with 9.1:Classification.Purpose = 'Discipline' and 9.2.1:Classification.TaxonPath. Source = 'Ariadne' and 9.2.2.2:Classification.TaxonPath. Taxon[1] = discipline type and 9.2.2.2:Classification.TaxonPath. Taxon[2] = discipline	2.3 subdiscipline	Subject

[1] Mandatory elements by SCORM 1.2 (ADL 2001a).

[2] Mandatory elements by the ARIADNE Educational Metadata Recommendation V 3.1 (ARIADNE 2002).

[3] The original LOM metadata element explanation is cited (IEEE LTSC 2002).

[4] The comment of the ARIADNE Educational Metadata Recommendation is cited (ARIADNE 2002).

References

ADL (2001a). Sharable Content Object Reference Model (SCORM) version 1.2. The SCORM Content Aggregation Model, Advanced Distributed Learning. http://www.adlnet.org/ (2002-02-25)

ADL (2001b). Sharable Content Object Reference Model (SCORM) version 1.2. The SCORM overview, Advanced Distributed Learning. http://www.adlnet.org/ (2002-02-25)

ADL (2002). Emerging and enabling technologies for the design of learning object repositories, Advanced Distributed Learning. http://xml.coverpages.org/ADLRepositoryTIR.pdf (2005-03-10)

ADL (2004). SCORM Content Aggregation Model. Version 1.3, Advanced Distributed Learning. http://www.adlnet.org/ (2004-03-04)

ARIADNE (2001). ARIADNE Foundation for the European knowledge pool. http://www.ariadne-eu.org/ (2004-11-01)

ARIADNE (2002). ARIADNE educational metadata recommendation - V3.2, EPFL, K.U.Leuven (Leuven, B), ARIADNE Foundation. http://www.ariadne-eu.org/en/publications/metadata/ams_v32.html (2004-12-22)

Ausubel, D. P. (1960). The use of advance organizers in the learning and retention of meaningful verbal material. Journal of Educational Psychology (51): 267-272

Ballstaedt, S.-P. (1997). Wissensvermittlung die Gestaltung von Lernmaterial. Weinheim, Beltz PsychologieVerlagsUnion

Barritt, C. and Lewis, D. (2000). Reusable learning object strategy. Definition, creation process, and guidelines for building, Cisco Systems Inc.

Baruque, L. B. and Melo, R. N. (2003). Learning theory and instructional design using learning object. Learning Objects 2003 Symposium: Lessons Learned, Questions Asked, Honolulu, Hawaii. http://www.cs.kuleuven.ac.be/~erikd/PRES/2003/LO2003/Baruque.pdf (2004-02-18)

Bloom, B. S., Mesia, B. B. and Krathwohl, D. R. (1956). Taxonomy of Educational Objectives: The classification of educational goals: Handbook I, cognitive domain. New York, Longmans, Green

Braxton, S., Kimberly, B. and Looms, T. (1995). Instructional design methodologies and techniques. http://www.futureu.com/publications/braxton/ (2004.03.02)

Chitwood, K., May, C., Bunnow, D. and Langan, T. (2000). Battle stories from the field: Wisconsin online resource center learning objects project. The Instructional Use of Learning Objects: Online Version. Wiley, D. A. (Ed.). http://reusability.org/read/chapters/chitwood.doc (2001-11-27)

Clases, C., Manser, T. and Wehner, T. (2001). Hyperlearning. Prozessbegleitende Evaluation eines Weiterbildungsangebots für die Schulleitungen der teilautonomen Volksschulen im Internet. Harburger Beiträge zur Psychologie und Soziologie der Arbeit Nr. 25

DCMI (2004). Dublin Core Metadata Element Set, Version 1.1: Reference, Dublin Core Metadata Initiative. http://dublincore.org/documents/2004/12/20/dces/ (2005-04-04)

Downes, S. (2001). Learning objects. Resources for distance education worldwide. International Review of Research in Open and Distance Learning **2**(1). http://www.irrodl.org/content/v2.1/downes.html (2002-02-14)

Duval, E. (2004). We're on the road to…. ED-MEDIA 2004, Lugano, Switzerland. http://www.cs.kuleuven.ac.be/~hmdb/publications/files/pdfversion/41316.pdf

Duval, E. and Hodgins, W. (2003). A LOM Research Agenda. www2003 - The Twelfth International World Wide Web Conference, Budapest, Hungary. http://www2003.org/cdrom/papers/alternate/P659/p659-duval.html.html

EducaNext (2004). EducaNext: A service for knowledge sharing. http://www.educanext.org/whitepaper.pdf

Gagné, R. M. (1985). The conditions of learning. New York, Holt, Rinehart & Winston, Inc.

Guttormsen Schär, S., Messerli, S., Schierz, C. and Schluep, S. (2004). Information categories for learning-object metadata. An empirical study on the practical validity of theoretically founded information categories. ED-MEDIA 2004, World Conference on Educational Multimedia, Hypermedia & Telecommunications, Lugano, Switzerland, Association for the Advancement of Computing in Education (AACE)

Hackos, J. (2002). Content Management for Dynamic Web Delivery, John Wiley & Sons

Hamel, C. J. and Ryan-Jones, D. (2002). Designing Instruction with Learning Objects. International Journal of Educational Technology (IJET) **3**(1)

Holtzblatt, K. and Jones, S. (1993). Contextual inquiry: A participatory technique for system design. Participatory design: Principles and practices. Schuler, D. and Namioka, A. (Eds.). Hillsdale, Lawrence Erlbaum Associates**:** 177-210

Horn, R. E. (1993). Structured writing at twenty-five. Performance and Instruction (32): 11-17. http://www.stanford.edu/~rhorn/a/topic/stwrtng_infomap/artclStWritingAt25.html (2005-01-27)

Horn, R. E. (1998). Structured writing as a paradigm. Instructional development: State of the art. Romiszowski, A. and Dills, C. (Eds.). Englewood Cliffs, Educational Technology Publications. http://www.stanford.edu/~rhorn/a/topic/stwrtng_infomap/artclParadigm.pdf (2003-03-04)

IEEE LTSC (2001). Draft standard for learning technology - Learning Technology Systems Architecture (LTSA). New York, Institute of Electrical and Electronics Engineers, Inc. http://ltsc.ieee.org/doc/wg1/IEEE_1484_01_D09_LTSA.pdf (2002-04-09)

IEEE LTSC (2002). Draft Standard for Learning Object Metadata. New York, Institute of Electrical and Electronics Engineers, Inc.

IMS (2000). IMS Digital Repositories White Paper, IMS Global Learning Consortium. http://www.imsproject.org/imsdr_whitepaper_v1p6.pdf (2002-08-13)

IMS (2001a). IMS Content Packaging Information Model, IMS Global Learning Consortium. http://www.imsglobal.org (2003-03-15)

IMS (2001b). IMS Learning Resource Meta-Data Information Model, IMS Global Learning Consortium. http://www.imsglobal.org (2002-01-30)

IMS (2002a). IMS Question & Test Interoperability: An Overview, IMS Global Learning Consortium. http://www.imsglobal.org (2003-03-15)

IMS (2002b). IMS Simple Sequencing Information and Behavior Model, IMS Global Learning Consortium. http://www.imsglobal.org (2003-03-15)

IMS (2003). IMS Learning Design Information Model, IMS Global Learning Consortium. http://www.imsglobal.org (2005-02-01)

Infrae (2005). The Silva open source content management system. http://www.infrae.com/products/silva (2005-01-10)

Issing, L. J. (2002). Instruktions-Design für Multimedia. Information und Lernen mit Multimedia und Internet. Issing, L. J. and Klimsa, P. (Eds.). Weinheim, Psychologie Verlags Union**:** 151-176

Kerres, M. (1998). Multimediale und telemediale Lernumgebungen: Konzeption und Entwicklung. München, Oldenbourg

Koper, R. (2000). From change to renewal: Educational technology foundations of electronic environments, Open University of the Netherlands. http://eml.ou.nl/introduction/docs/koper-inaugural-address.pdf (2002-04-09)

Koper, R. (2001). Modeling units of study from a pedagogical perspective. The pedagogical meta-model behind EML, Educational Technology Expertise Centre, Open University of the Netherlands. http://eml.ou.nl/introduction/docs/ped-metamodel.pdf (2002-02-20)

Koppi, T. and Lavitt, N. (2003). Institutional Use of Learning Objects Three Years on: Lessons Learned and Future Directions. Learning Objects 2003 Symposium: Lessons Learned, Questions Asked, Honolulu, Hawaii. http://www.cs.kuleuven.ac.be/~erikd/PRES/2003/LO2003/Koppi.pdf (2004-02-18)

Kupu (2005). Kupu Editor. http://kupu.oscom.org/

L'Allier, J. J. (1997). Frame of reference: NETg's map to its products, their structures and core beliefs. http://www.netg.com/research/frameref.htm

Lambe, P. (2002). The autism of knowledge management. http://greenchameleon.com/thoughtpieces/autism.pdf (2003-04-01)

Latteier, A., Pelletier, M., McDonough, C. and Sabaini, P. (2003). The Zope Book (2.6 Edition). http://zope.org/Documentation/Books/ZopeBook/2_6Edition (2005-01-10)

Longmire, W. (2000). A primer on learning objects. Learning Circuits (March, 2000). http://www.learningcircuits.org/2000/mar2000/Longmire.htm (2004-04-13)

LSAL (2003). SCORM best practices guide for content developers, Carnegie Mellon Learning Systems Architecture Lab. http://www.lsal.cmu.edu/lsal/expertise/projects/developersguide/

McGee, P. A. (2004). Learning Object Repositories. http://elearning.utsa.edu/guides/LO-repositories.htm (2004-11-01)

MERLOT (2004). Multimedia educational resource for learning and online teaching. http://www.merlot.org/ (2004-11-01)

Merrill, M. D. (1983). Component Display Theory. Instructional-design theories and models: An overview of their current status. Reigeluth, C. M. (Ed.). Hillsdale, Lawrence Erlbaum Associates**:** 279-333

Merrill, M. D. (1987). A lesson based upon Component Display Theory. Instructional Design Theories in Action. Reigeluth, C. M. (Ed.). Hillsdale, Lawrence Erlbaum Associates**:** 201-244

Najjar, J., Ternier, S. and Duval, E. (2004). User behavior in learning object repositories: an empirical analysis. ED-MEDIA 2004, Lugano, Switzerland. http://www.cs.kuleuven.ac.be/~najjar/papers/edmedia2004.pdf (2005-03-10)

Nichani, M. (2001). LCMS = LMS + CMS [RLOs] - How does this affect the learner? The instructional designer? elearningpost (May 2, 2001). http://www.elearningpost.com/elthemes/lcms.asp (2002-01-16)

Polsani, P. R. (2003). Use and Abuse of Reusable Learning Objects. Journal of Digital information **3**(4). http://jodi.ecs.soton.ac.uk/Articles/v03/i04/Polsani/ (2003-04-24)

Python Software Foundation (2005). Python Programming Language. http://www.python.org/ (2005-01-10)

Rawlings, A., Rosmalen, P. v., Koper, R., Rodriguez-Artacho, M. and Lefrere, P. (2002). Survey of Educational Modelling Languages (EMLs), CEN/ISSS WS/LT. http://www.cenorm.be/cenorm/businessdomains/businessdomains/isss/activity/emlsurveyv1.pdf (2005-01-18)

Reigeluth, C. M. and Stein, F. (1983). The elaboration theory of instruction. Instructional design theories and models. Reigeluth, C. M. (Ed.). Hillsdale, Erlbaum Associates

Salo, D. (2001). What is this thing called structure? XML for typesetters. Electronic Book Web. http://12.108.175.91/ebookweb/stories/storyReader$380

Schluep, S., Ravasio, P. and Guttormsen Schär, S. (2003). Implementing learning content management. Ninth IFIP TC13 International Conference on Human-Computer Interaction (Interact 2003), Zürich, Switzerland, IOS Press

Schulmeister, R. (2003). Lernplattformen für das virtuelle Lernen: Evaluation und Didaktik. München, Oldenbourg

Schüpbach, E., Guggenbühl, U., Krehl, C., Siegenthaler, H. and Kaufman-Hayoz, R. (2003). Didaktischer Leitfaden für E-Learning. Didactic guidelines for e-learning. Bern, h.e.p.-Verlag

South, J. B. and Monson, D. W. (2000). A university-wide system for creating, capturing, and delivering learning objects. The Instructional Use of Learning Objects: Online Version. Wiley, D. A. (Ed.). http://reusability.org/read/chapters/south.doc (2001.11.27)

Süß, C. (2000). Adaptive Knowledge Management: A Meta-Modeling Approach and its Binding to XML. 2. GI-Workshop Grundlagen von Datenbanken, Kiel, Germany. http://im.fmi.uni-passau.de/db/literatur?project=LMML

Süß, C. and Freitag, B. (2000). Entwicklung und Nutzung von Teachware. Das Passauer Knowledge Management System (PaKMaS). Computerunterstütztes

Lernen. Kammerl, R. (Ed.). München, Oldenbourg. http://daisy.fmi.uni-passau.de/papers/SF00/paper.pdf

Verbert, K. and Duval, E. (2004). Towards a global component architecture for learning objects: A comparative analysis of learning object content models. ED-MEDIA 2004, Lugano, Switzerland. http://www.cs.kuleuven.ac.be/~hmdb/publications/publicationDetails.php?id=41315

W3C (1999a). HTML 4.01 Specification. http://www.w3.org/TR/1999/REC-html401-19991224 (2005-01-05)

W3C (1999b). Mathematical Markup Language (MathML™) 1.01 Specification. http://www.w3.org/1999/07/REC-MathML-19990707 (2005-02-01)

W3C (1999c). Namespaces in XML. http://www.w3.org/TR/REC-xml-names (2005-01-05)

W3C (1999d). XSL Transformations (XSLT). Version 1.0. http://www.w3.org/TR/xslt (2005-01-05)

W3C (2002). XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition). A Reformulation of HTML 4 in XML 1.0. http://www.w3.org/TR/2002/REC-xhtml1-20020801 (2005-01-05)

W3C (2003). Scalable Vector Graphics (SVG) 1.1 Specification. http://www.w3.org/TR/2003/REC-SVG11-20030114/ (2005-02-01)

W3C (2004a). Extensible Markup Language (XML) 1.0 (Third Edition). W3C Recommendation 04 February 2004. http://www.w3.org/TR/2004/REC-xml-20040204/ (2005-01-05)

W3C (2004b). XML Schema Part 0: Primer Second Edition. http://www.w3.org/TR/xmlschema-0/ (2005-01-05)

Wagner, E. D. (2002). Steps to creating a content strategy for your organisation. The e-Learning Developers' Journal (October 29). http://www.elearningguild.com/pdf/2/102902MGT-H.pdf (2005-01-26)

Walsh, N. and Muellner, L. (1999). DocBook: The definitive guide. Sepastopol, O'Reilly. http://www.docbook.org/tdg/en/ (2004-12-29)

Wiest, S. and Zell, A. (2001). Improving Web Based Training using an XML content base. ED-MEDIA 2001, Tampere, Finland. http://www-ra.informatik.uni-tuebingen.de/publikationen/2001/wiest_edmedia2001.pdf

Wiley, D. A., Ed. (2000). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. The Instructional Use of Learning Objects: Online Version. http://reusability.org/read/chapters/wiley.doc

Glossary

A

ADL ***A**dvanced **D**istributed **L**earning:* Initiative to establish standards for a new distributed learning environment for the government, industry and academia. An important contribution of ADL is the *SCORM* specification.

AICC ***A**viation **I**ndustry **CBT** **C**ommittee:* International association for the development of guidelines and specifications for CBT.

ARIADNE ***A**ssociation of **R**emote **I**nstructional **A**uthoring and **D**istribution **N**etwork for **E**urope:* An association of mainly higher education institutions in Europe for sharing learning resources.

Asset Electronic representations of media: text, images, videos, web pages.

C

CBT ***C**omputer **B**ased **T**raining*

Component Display Theory A strategy for designing instruction which emphasizes different components of instruction for different instructional goals. The Component Display Theory classifies learning along two dimensions: contents (facts, concepts, principles, procedures) and performance levels (remembering, using, finding).

Content Aggregation A Content Aggregation is a combination of *learning objects* in a cohesive unit of instruction. This notion is used by the *SCORM* standard.

Content Chunking The process to divide learning contents into modular components.

Content Element The *dLCMS* defines a Content Element as a small, modular unit of learning content, which

- Serves as basic building blocks of learning contents,
- Can be aggregated into larger didactically sound Learning Units,
- Is *self-contained*,
- Represents a single *didactic content type* and relates to a single objective (fact, concept, principle, procedure, process),
- Is reusable in multiple instructional contexts.

Content Packaging A data format for the packaging of learning materials, such as courses, into interoperable, distributable packages. Two important standards are available: the *IMS* Content Packaging specification and the *SCORM* Content Aggregation Model.

Contextual Inquiry A method to investigate and understand the usage of a product in the context of its use. It is an approach using interviews and observations to gain rich information and is based on three core principles: context, partnership, and focus.

CSS ***C**ascading **S**tyle **S**heets:* A *W3C* standard for adding style (e.g. fonts, colors, spacing) to web documents.

D

Didactic Content Type	Didactic content types (definitions, examples, exercises, simulations, self-assessments, etc.) define a level of *granularity*. They may be seen as didactic steps according to one of *Gagné's nine instructional events*. *Learning objects* based on didactic content types can be flexibly combined to suit different learner groups needs. Components with a high potential for reuse together can be used together with elements which apply to a scientific discipline more specifically.
dLCMS	***d**ynamic **L**earning **C**ontent **M**anagement **S**ystem:* A tool which implements a *modularization* strategy combined with *structured markup* to enhance the reusability of learning contents.
DocBook	An *XML* markup language for the documentation of soft- and hardware.
DTD	***D**ocument **T**ype **D**efinition:* A declaration that formally specifies the structure of an *XML* document.

E

EducaNext	A portal supporting services for the creation and sharing of knowledge for higher education.
EML	***E**ducational **M**odeling **L**anguage:* An *XML*-based meta-model to describe educational scenarios, which has been developed by the Open University of the Netherlands (OUNL).

F

Formative Evaluation	A method of judging the worth of a program or product while the program activities or product development are forming or happening. Formative evaluation focuses on the process.

G

Gagné's Nine Instructional Events	Gagné created a nine-step process, which correlates to and addresses the conditions of learning based on a cognitive model of a human's learning process. The nine steps are: 1. Gaining attention (reception) 2. Informing learners of the objective (expectancy) 3. Stimulating recall of prior learning (retrieval) 4. Presenting the stimulus (selective perception) 5. Providing learning guidance (semantic encoding) 6. Eliciting performance (responding) 7. Providing feedback (reinforcement) 8. Assessing performance (retrieval) 9. Enhancing retention and transfer (generalization).
Granularity	Granularity is the relative size, scale, level of detail which characterizes *learning objects*.

H

HTML	***H**yper**T**ext **M**arkup **L**anguage:* The general publishing language of the World Wide Web, a *W3C* standard.

I

IEEE LTSC	***I**nstitute of **E**lectrical and **E**lectronics **E**ngineers **L**earning **T**echnology **S**tandards **C**ommittee:* Develops technical standards, recommended practices, and guides for learning technologies.
IMS	*IMS Global Learning Consortium:* Non-profit organization for the development of open technical specifications for interoperable learning technology.
ISD	***I**nstructional **S**ystems **D**esign*: A formal process for designing computer-based or traditional instructor-led training. It includes five phases: analysis, design, development, implementation, evaluation.
ISDMELO	***I**nstructional **S**ystems **D**evelopment **M**ethodology based on **e**-**L**earning **O**bjects:* A methodology to develop e-learning instruction using the *learning object* paradigm.

J

JavaScript	An object-based scripting programming language used mostly in web pages. The standardized version of JavaScript is also known as ECMAScript.

K

Kupu	An open source, browser-based online editor using JavaScript technology.

L

Learning Design	An *IMS* specification providing a flexible language to describe a wide range of pedagogies in online learning. It evolved from the development of *EML*.
Learning Object	A relatively small, self-contained unit of learning content which can be aggregated with other learning objects into larger learning units and which is reusable in multiple instructional contexts.
Learning Object Component Model	Learning content component models define different levels of components, the properties of these components, and how the components can be aggregated.
Learning Objective	The desired learning outcome of teaching and learning in relation to people's knowledge and understanding, skills, values and attitudes, practices and behavior.
Learning Unit	The *dLCMS* defines a Learning Unit as an aggregation of *Content Elements*, which is presented to learners. Typically, a Learning Unit serves as an online lesson.
LMML	***L**earning **M**aterial **M**arkup **L**anguage:* An *XML*-based markup language for learning contents, which was developed at the University of Passau.
LMS	***L**earning **M**anagement **S**ystem:* A software system designed to facilitate management and student involvement in e-learning. An LMS organizes and provides access to online education services for students, teachers, and administrators. These services usually include access control, provision of learning content, communication tools, and administration of user groups.

LOM ***L**earning **O**bject **M**etadata:* This *IEEE LTSC* standard specifies the syntax and semantics of metadata for *learning objects* to facilitate search, evaluation, acquisition and use of learning objects.

LSAL *Carnegie Mellon **L**earning **S**ystem **A**rchitecture **L**ab*

M

MathML ***Math**ematical **M**arkup **L**anguage:* An *XML*-based markup language for mathematical formulas to be included in web pages; a *W3C* standard.

MERLOT ***M**ultimedia **E**ducational **R**esource for **L**earning and **O**nline **T**eaching:* A *learning object* repository for faculty and students of higher education.

Modularization The concept of a modular *learning object component model* as well as the *content chunking* process.

O

OLAT ***O**nline **L**earning **a**nd **T**raining:* An open source *LMS* developed by the University of Zürich.

Q

QTI *IMS **Q**uestion and **T**est Interoperability **S**pecification:* A standardized, *XML*-based data format for question and tests.

R

RIO ***R**eusable **I**nformation **O**bject:* A granular, reusable chunk of information being the basic building block of Cisco's *RLO* strategy.

RLO ***R**eusable **L**earning **O**bject:* As defined by Cisco's RLO strategy, an RLO combines 7 ± 2 *RIOs*, an overview, a summary and an assessment.

S

SCORM ***S**harable **C**ontent **O**bject **R**eference **M**odel:* The standard model for sharable learning content objects provided by *ADL*. SCORM combines the standards and specifications of other organizations to produce a consistent functional model.

SCO ***S**harable **C**ontent **O**bject:* The reusable basic building blocks of learning content defined by *SCORM*. A SCO should be independent of learning context and must not contain any links to other SCOs. It is the smallest logical unit of instruction to be delivered and tracked with an LMS.

Self-contained In order to be freely combined, *learning objects* must not depend on another specific learning object. In the scope of our work, *learning objects* are considered to be self-contained if they have no other explicit reference to other contents, such as hypertext links or explicit linguistic reference in the text, i.e. words or sentences referring to other contents.

Silva An open source content management system based on the *Zope* web application framework.

Structured Markup Structured markup serves as a data format and allows one to separate contents from the visual presentation. Structured markup can be applied to contents using markup languages, such as *HTML* and *XML*. The different pieces of contents are marked up using tags, which assign a name to the content fragments.

Structured Markup Schema A formal specification of a set of elements and structuring rules in order to create a special-purpose markup language. Two important ways to create structured markup schemas for *XML* are Document Type Definitions (*DTD*) and *XML Schemas*.

Structured Writing The *Structured Writing* method was developed as a comprehensive performance-based approach for instructional developers and business writers to prepare clear and concise training manuals, proposals, reports, and memos.

SVG ***S****calable* ***V****ector* ***G****raphics:* A *W3C* standard for describing two-dimensional graphics in *XML*.

U

URL ***U****niform* ***R****esource* ***L****ocator:* A standardized address for resources on the Internet.

V

Validity An XML document is valid if it complies with all the rules specified by a *structured markup schema*.

W

W3C *World Wide Web Consortium:* A standardization organization for web technologies.

X

XML *E**x**tensible* ***M****arkup* ***L****anguage:* A general-purpose markup language for creating special-purpose markup languages; a *W3C* standard.

XHTML *Extensible HyperText Markup Language:* An *HTML* version which is fully *XML compliant;* a *W3C* standard.

XML Schema A *structured markup schema* for *XML*; a *W3C* standard.

XSL *E**x**tensible* ***S****tylesheet* ***L****anguage:* A *W3C* standard for expressing stylesheets and transformations of XML documents.

Z

Zope An open source, object oriented web application server written in the programming language Python.

ZIP A widely-used data format for compressed file archives.

Index

About the Author

Samuel Schluep was born in Wilkinsburg PA, USA in1964. After attending primary and high school in the USA, and Switzerland, he studied electrical engineering at the ETH in Zurich, specializing in communications technology. He graduated in 1990 with a master's degree. His diploma thesis investigated robotic control with neural networks. During the subsequent six years he worked in the industry as a development engineer for exhaust measuring instruments. Since 1997 he has been a research associate at the Institute of Hygiene and Applied Physiology at the ETH Zurich in the field of man machine interaction. His research activities have focused on the modeling of users' problem solving strategies and mental models, on the design of interactive learning applications, and on guidelines for multimedia design. In 2001 he started his PhD investigating technologies which enhance the reusability of e-learning contents.