Olli Hokkanen

REUSABLE DIGITAL LEARNING MATERIAL PRODUCTION

Master's thesis

Teacher Education study line

17.12.2003

University of Jyväskylä

Faculty of Information Technology

Author: **Olli Hokkanen**

Contact: olhokkan@cc.jyu.fi

Title: Reusable Digital Learning Material Production

Work: Master's Thesis

Pages: 83

Study Line: Teacher Education study line

Orderer: University of Jyväskylä, Faculty of Information Technology

Keywords: Learning Object, Reusability, Metadata

Tiivistelmä: Oppisisällöt ja niitä kuvaavan metatiedon standardointi ovat viime aikoina olleet keskeisesti esillä opetusteknologiaa koskevissa keskusteluissa, sillä näiden käsitteiden sanotaan tarjoavan suoraviivaisen ratkaisun uudelleenkäytettävän oppimateriaalin tuottamiseen ja jakamiseen eri toimijoiden kesken. Tässä pro gradu -työssä esitellään näiden käsitteiden taustalla olevia teorioita ja standardeja, sekä pohditaan kyseisten käsitteiden soveltuvuutta oppimateriaalin tuottamiseen niin teorian kuin käytännön esimerkkien kautta.

Abstract: Learning objects and the standardisation of learning object metadata have lately been central to many discussions in the field of educational technology, since they seem to offer a straightforward solution for creating and distributing reusable learning material. This thesis introduces the theories and established standards behind these concepts and considers their applicability for learning material production both in theory and practise.

Foreword

I would like to thank various people at the Agora Center, department of the University of Jyväskylä, for providing the possibilities for accomplishing this thesis. Especially I want to thank my thesis supervisor, Professor Tommi Kärkkäinen, and Senior Researcher Marja Kankaanranta for their valuable comments during the production of this thesis.

Jyväskylä, December 2003

Olli Hokkanen

'Two inventions of man must surely be viewed as the most difficult: the art of government and the art of education.'

Immanuel Kant (1724–1804)

Terms

ID	Instructional Design – the systematic process of translating principles of learning into plans for instruction.
LCMS	Learning Content Management System – computerised system for learning content management.
LO	Learning Object – a modular teaching material component.
LOM	Learning Object Metadata – metadata for describing a learning object.
RIO	Reusable Information Object – a learning object without strictly defined pedagogies.
RLO	Reusable Learning Object – same as a learning object.
RDF	Resource Description Framework for Metadata Standards – a foundation for metadata processing and systems interoperability.
DC	Dublin Core – an organisation that promotes interoperable metadata standards.
XML	Extensible Markup Language – a universal markup language for data interchange between applications.
XPath	XML Path Language – a querying languge for addressing specific parts of an XML data object.
XSLT	XSL Transformations – a language for specifying transformations of XML documents.

DTD	Document Type Definition – a set of rules how XML data should be structured.
VLE	Virtual Learning Environment – a software application that supports online learning.
W3C	World Wide Web Consortium – a consortium that develops interoperable technologies for World Wide Web.
IEEE	Institute of Electrical and Electronics Engineers – an association of electrical and electronics engineers
LTSC	IEEE Learning Technology Standards Committee - develops technical standards, recommended practices, and guides for learning technology.

Contents

1 INTRODUCTION 7

2 ON LEARNING AND INSTRUCTION 9

2.1 LEARNING THEORIES 9
2.2 INSTRUCTIONAL DESIGN 14

3 LEARNING OBJECTS 17

3.1 DEFINITIONS 17
3.2 GOALS BEHIND LO RESEARCH 19
3.3 GRANULARITY OF LEARNING OBJECTS 21
3.4 PEDAGOGY OF LEARNING OBJECTS 22

4 XML – EXTENSIBLE MARKUP LANGUAGE 25

4.1 SYNTAX 26
4.2 DTD – DOCUMENT TYPE DEFINITION 27
4.3 XML SCHEMA 28
4.4 XML NAMESPACES 30
4.5 XHTML 30
4.6 XPATH AND XSLT 31

5 LEARNING OBJECT METADATA 35

5.1 DUBLIN CORE METADATA STANDARD 35
5.2 RESOURCE DESCRIPTION FRAMEWORK - RDF 36
5.3 ONTOLOGIES 38
5.4 LEARNING OBJECT METADATA STANDARDISATION 40
5.4.1 LOM 40
5.4.2 CELEBRATE 42
5.4.3 IMS 43

6 THOUGHTS ON LEARNING OBJECTS 47

6.1 LEARNING OBJECTS IN PRACTISE 47
6.2 PRACTICAL GRANULARITY AND PEDAGOGY OF LEARNING OBJECTS 50
6.3 LO, RLO OR RIO? 51

7 LEARNING MATERIAL REPOSITORIES 53

7.1 EXISTING LEARNING MATERIAL REPOSITORIES 54
7.1.1 Merlot 54
7.1.2 MIT OpenCourseWare 56
7.2 REPOSITORY TECHNOLOGIES 58
7.2.1 Digital repositories interoperability – IMS specification 58
7.2.2 Norwegian education portal 59
7.3 DIGITAL LIBRARIES SUPPORTING LEARNING CONTENT PRODUCTION 60

7.3.1 CiteSeer ... 60
7.3.2 Public Library of Science - PLoS ... 61
7.4 SOFTWARE AGENTS ... 62
8 CASE STUDY FOR CREATING ONLINE COURSE CONTENT ... 65
8.1 TECHNOLOGIES FOR CREATING LEARNING OBJECTS ... 65
8.1.1 Creating a content package ... 66
8.1.2 Content package metadata ... 67
8.1.3 Incorporating learning objects to a virtual learning environment ... 68
8.2 THE TOPIC-CASE APPROACH FOR CREATING COURSE CONTENT ... 69
8.3 TOPIC-CASE DEVELOPMENT PROCESS PHASES ... 69
8.3.1 Phase 1: Background study ... 70
8.3.2 Phase 2: Content design ... 72
8.3.3 Phase 3: Pedagogical design ... 73
8.3.4 Phase 4: Technical design ... 74
8.3.5 Phase 5: Realisation and assessment ... 74
8.4 ANALYSIS ... 75
9 CONCLUSIONS ... 76
REFERENCES ... 79

1 Introduction

This thesis focuses on introducing some approaches to learning object (LO) design and research based on creating learning object repositories. A short introduction to XML, usage of digital libraries supporting content production, existing solutions on creating metadata on digital learning material, and analysis of the topic-case driven approach in web course content production are also introduced.

‘Learning objects’, ‘learning object metadata’ and ‘learning object repositories’ are terms that have lately been central to many discussions and projects of both public and private educational organisations. These terms have been associated with a range of benefits, some of which seem strange in educational context, such as systems interoperability and resource reusability. On the basis of the benefits that these terms might suggest, governments and industry are spending substantial amounts of money in educational technology. This has given rise to a significant learning object and standardisation movement in the field of educational technology. (Friesen, 2003)

This thesis advances from theoretical issues to more technical aspects of learning objects research, such as standardisation etc. It is not the purpose of this paper to introduce pedagogical aspects of teaching and learning extensively, but since teaching and designing learning material is, or at least it should be, based on understanding of human learning, some aspects of pedagogical issues and instructional design are discussed.

Main research questions considered in the thesis are the following:

- What is the current situation in learning objects research, are there any ‘best practises’ to create reusable learning objects?
- What is the current situation in learning object metadata research, do we need so many different metadata standards?
- How are different learning theories taken into consideration in learning object research?

- How to apply learning object design theories in practise? Is it possible at this point to implement an online course to existing virtual learning environments by using and creating reusable learning objects?

Chapter 2 concentrates on introducing different learning theories and their impact on the systematic design of instruction, instructional design. In chapter 3 the concept 'learning object' is introduced. Chapter 4 focuses on introducing XML and its applications that are commonly used with learning object technologies.

Learning object metadata and some metadata standardisation efforts are introduced in chapter 5. Chapter 6 consists of a critical overview of learning objects and their applicability for learning content production. Some existing learning material repositories and information resources that are usable for learning content production are reviewed in chapter 7.

In chapter 8 an application is tested for creating reusable learning content that is described by standardised metadata. Furthermore, one structured methodology for web course content production is considered. Chapter 9 consists of conclusions of the issues concerned in this thesis.

2 On learning and instruction

Many theories define learning, but no single leaning definition is universally accepted by theorists, researchers, and practitioners. Learning can be defined as behavioural change or change in the capacity for behaviour. It involves a changed capacity to behave in a given fashion, i.e., becoming capable of doing something differently. Learning is also the acquisition of knowledge and skills, the formation of mental structures, and the processing of information and beliefs. (Schunk, 2000)

2.1 Learning theories

Behaviorism, cognitivism and constructivism – that is one way to characterise and divide learning theories, both historically and conceptually. Modern understanding on human learning no longer states that learning is simply the transfer of knowledge from a teacher to students, or simplifies learning as "repeating a new behavioral pattern until it becomes automatic", as is the case in the operant conditioning theory within behavioral psychology. To understand the historical aspects of different theories of learning in computer aided instruction, as well as their importance today, a short review of behaviorism, cognitivism and constructivism is introduced in this chapter. This summary is mostly based on Dale H. Schunk's book on learning theories (Schunk, 2000).

Behaviorism rose in the beginning of the 20th century. The theory of behaviorism concentrates on the study of behaviours that can be observed and measured. Russian psychologist Ivan Pavlov's (1849-1936) contribution to the theory of learning was his well known work on classical conditioning, remembered by the famous experiment on dogs salivating at the sight of the attendant bringing them food. John B. Watson (1878-1958) is generally considered the founder of modern behaviorism. He believed that Pavlov's model could be extended to explain diverse forms of learning and personality characteristics. Watson believed that people are born capable of displaying just three emotions: love, fear, and rage. All other human behaviour is established through stimulus-response associations through conditioning.

Operant conditioning theory, formulated by B.F Skinner (1904-1990), is based on the assumption that features of the environment (stimuli, situations, events) serve as cues for responding. Reinforcement strengthens responses (behaviour) and increases their future possibility of occurring, whereas punishing consequences decrease behaviour when the stimuli are present. Operant principles have been applied to many aspects of teaching and learning, e.g., simulations and programmed instruction.

Skinner believed that instruction is most effective when:

(a) Teachers present the material in small steps.

(b) Learners actively respond than passively listen.

(c) Teachers give feedback immediately following learners' responses.

(d) Learners move through the material at their own pace.

Basically, according to behavioristic principles human behaviours occur in response to external factors and stimulating these factors changes the probability of the desired response to occur in the future. Behavioristic methods are most useful in repetitive tasks that require rapid actions to be taken when necessary. They are often teacher-centered and content driven, forcing learners into a passive role, merely listening to lectures and following instructions.

Teaching methods that concentrate on repeating and memorising individual facts do not necessarily help learners to achieve a deeper understanding of the whole issue, nor do they help learners to transfer their skills to other contexts. For example, memorising formulas of physics without understanding the symbols in them probably does not help the learner to apply the formulas to real problems. However, behavioristic methods can be succesfully applied to some parts of teaching process, e.g., for introducing course content, as long as the drawbacks of behaviorism in education are realised.

Behavioral theories dominated the psychology of learning for the first half of the 20th century. However, as early as in the 1920's the limitations of the behavioral learning theory became obvious. Jean Piaget (1896-1980) was one of the most important authors in developing cognitivism, in which learning is seen as acquisition of knowledge and cognitive structures due to information processing. As Schunk (2000) states, cognitive theorists claimed that the operant conditioning theory, ignoring mental processes, offers an incomplete account of human learning. As opposed to behaviorism, the knowledge acquisition in cognitivism is measured by what learners know, not necessarily what they do.

Social cognitive theory, developed by Albert Bandura (1925-) states that people learn from their social environments. Human functioning is seen as a series of mutual interactions among personal factors, behaviours, and environmental events. Social cognitivism interprets learning as an information processing activity in which knowledge is cognitively represented as symbolic representations serving as guides for action. Learning occurs enactively through actual performances and indirectly by observing models and listening to instructions. (Schunk, 2000)

Cognitivistic learning theories often state that learners acquire knowledge into their existing mental structures by actively processing new information. This means that new information and learners' prior knowledge should overlap for a meaningful learning experience to occur. Cognitive methods are most useful when the aim is to gain understanding on the whole subject instead of merely remembering some separate facts or behaviour models. Basically teaching should always be based on learners' previous knowledge and the teacher should activate learners' existing mental representations on the subject by some means.

Constructivism, based on cognitive psychology and social psychology, states that individual learner must actively build knowledge and skills; people are active learners and must construct knowledge for themselves. The basic distinction to previously mentioned theories is that while behaviourists view knowledge as something that happens in response to external factors (stimulus-response associations), and cognitivists view knowledge as abstract symbolic representations of reality inside the learner's head, constructivists view knowledge as constructed internally by each individual. This means that knowledge cannot be transferred from one person to another intact, since each individual shapes their own knowledge to fit their personal frame of reference. In constructivism problem solving and authentic situations are considered important factors of a meaningful learning experience. Socio-constructivist theory, or social constructivism, is basically an extension of the above mentioned constructivist idea, stating also that meaningful construction of knowledge occurs when the learner interacts with other learners. (Schunk, 2000)

According to constructivist principles of learning, learners must take an active role in the learning process and construct new knowledge themselves to achieve meaningful 'deep learning' experiences. Teacher's role is mainly to support the knowledge creation process. Constructivistic methods are most useful in complex and unique learning goals and activities, but they do often require advanced learning skills from the learners. Applying constructive methods to teaching also aims to help learners develop their learning skills, i.e., their capability to independently maintain and develop their professional capabilities without formal training. One should also remember that constructivism, for example, is not one single theory but rather an umbrella of several constructivist theories.

To clarify the basic distinctions between these three theories and their practical applications, some example methods that are used for teaching different skills are presented in Table 1.

Theory	**Application**	**Description**
Behaviorism	Drill and practise, simulation	Learning through repetitive exercises is a commonly used way of teaching skills that require physical actions to be taken in a very short period of time in response to external factors, e.g., accidents (fire service personnel) or hostile attacks (armed forces).

Cognitivism	Real-life based problem solving	New information is based on learners' previous knowledge. Learners also actively participate in the learning process by applying new information into meaningful problems, instead of just passively listening to instructions or lectures.
Constructivism	Case analysis and resolution (groupwork)	Basically the same as in cognitivism, but learners take larger responsibility of their own learning and externalise their knowledge by some means. Actively processing information and creating new knowledge, possibly together with other learners, is considered the most efficient way to learn new skills in (cognitively) complex environments and tasks.

Table 1. Learning theories and applications.

However, these three main theories of learning overlap each other. For example, behaviorism and constructivism, despite their obvious differences, should not be seen as two opposite theories that have nothing in common. Both theories try to explain the learning experience and share some of the elements that are usable when organizing teaching and learning activities. In fact, Ertmer and Newby (1993) suggest that teaching strategy should be chosen according to learner's previous experience on the subject and the level of cognitive processing required by the task (see Figure 1). From this point of view the teacher can choose a suitable teaching strategy from all available practical applications of the different learning theories.

After decades of learning research and the influence of cognitive psychology to this research, educational psychologists and training designers developed model for the systematic design of instruction, instructional design (Häkkinen, 2002). This model is shortly reviewed in the next section.

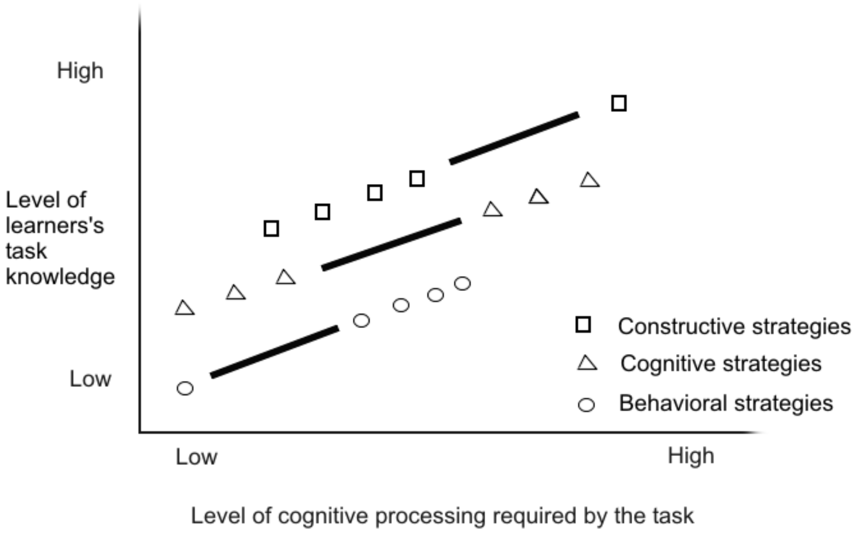

Figure 1. Comparison of different instruction strategies (Ertmer & Newby, 1993).

2.2 Instructional Design

According to Gagne (1985), combining various kinds of learning outcomes and the ways of analysing learning requirements in a rational and systematic manner, it is possible to describe a total set of ideas that constitute a *theory of instruction*. Applying these ideas into plans for instructional materials is called *instructional design* (ID). Instructional system development (ISD) combines instructional design theories with computer software and networks.

Gagne (1985) states that instructional design generally requires two major questions to be answered as basis for (instructional) design procedures:

1. What kinds of capabilities are to be learned?

2. What kinds of stimulation external to the learner will best support the internal processing necessary for learning?

After the learning outcome is identified, the instructional designer specifies the external conditions of learning and sequences them in a series of events given the general name of instruction.

Gagne's idea of instructional design can be divided into the following steps (Gagne, 1985):

1. Analyse the domain knowledge into a hierarchy of atoms, which are either a small piece of knowledge or a simple combination of previously specified atoms.

2. Sequence the instructional atoms so that a combination atom is not taught until its components have been taught.

3. Design an instructional approach for each atom in the sequence. This procedure is often divided into three phases: task analysis, selection of instructional method, and evaluation. Task analysis identifies the factors (e.g., students' prior knowledge) that need to be taken into account when the instructional methods are selected. Finally, the instructional atoms are tested and revised using, e.g., empirical methods to evaluate effectiveness, acceptability, and feasibility of use of the instructional package.

Gagne's theory of instructional design is closely related to Skinner's idea on how to organise instruction effectively. Teaching content is organised into instructional sequences, and learners proceed through prescribed sequences. If the learning content is clearly specified (e.g., a well-defined task in a factory), ID methods are effective. Basically, instructional design is a set of methods, materials and assessments aimed at promoting competence in defined outcomes. Traditional ID methods and models have been considered to be externally directed and content driven. However, nowadays the learning content, as well as the learning goals and activities, can be quite complex and unique. In such cases the problems of traditional ID methods become obvious. (Häkkinen 2002)

Tennyson (1994) stated that the (traditional) ID models are often too idealistic and the ID practise should be seen as a dialogue between available solutions and the definition and refinement of instructional goals. In this sense instructional design is neither as a topdown, objectives driven activity, nor as a bottom up, opportunistic activity, but a combination of the two.

Current notions on learning have challenged instructional design considerably. Different conceptions of teaching and learning have, however, lead to different views on instruction. According to Häkkinen (2002), constructivism emphasises authentic problem solving, while according to elaboration theory instruction should be organized in increasing order of complexity for optimal learning. Häkkinen points out that the main problem of instructional design has been its isolation from other fields of teaching, learning and technology. Instructional designers have not been very open to new alternatives; they have not reassessed the basic foundations or assumptions of the ID models. Based on, e.g., the constructivist philosophy of learning, instruction should support the construction of knowledge rather than just communicate information, as is often the case in traditional ID models.

Since the main concept of this thesis, i.e., the learning object model, has been developed by instructional designers, possible drawbacks due to the tradition of ID should be considered when analysing the models of learning object design and development. However, as Wiley (2000, 2003) states, instructional design theory, or instructional strategies, guidelines, and criteria for their application, must play a large role in the application of learning objects technology if it is to succeed. Wiley refers to the fact that though many organizations and committees promote discussion around the technology standards necessary to support learning object-based instruction, or about their financial opportunities, there has been little conversation around the instructional design implications of learning objects. Current learning object approaches also frequently contradict recent research on learning, although they harmonise well with 1980’s learning research.

3 Learning objects

The term 'learning object' was introduced into the field of instructional technology around 1994. The main idea was that educational resources could be broken into modular components, which could later on be combined by instructors, learners, and eventually computers into larger structures that would support learning. Because the learning objects would be in the digital form, they could be simultaneously used and reused in different learning contexts. (Gibbons et al., 2000)

3.1 Definitions

The building blocks of the new learning technology are called learning objects. They are the elements of a new type of computer-based instruction grounded in the object-oriented paradigm of computer science (Wiley, 2000). The idea is that teachers and instructional designers can build learning objects that are deliverable over the Internet and can be reused a number of times in different contexts.

As Szyperski (1998) states, the terms 'component' and 'object' are often used interchangeably within the field of computer science. However, they are not the same concept. According to Szyperski, the characteristic properties of (software) components are:

- A component is a unit of independent deployment that is well separated from its environment and other components.
- A component is a unit of third-party composition.
- A component has no persistent state.

Likewise, the characteristic properties of (software) objects are (Szyperski, 1998):

- An object is a unit of instantiation; it has a unique identity.
- An object has state, which can be persistent.

- An object encapsulates its state and behaviour.

It could be said that a software component is a reusable asset, often a unit of third-party composition and constructed from objects. According to Szyperski (1998), software objects are instances of classes, which themselves are related by (traditional) inheritance. Exceptions to Szyperski's object definition are object-oriented languages that do not use classes, for example Self (Ungar & Smith, 1987) and Kevo (Taivalsaari, 1996). From the software engineering perspective, a more suitable description for a 'learning object' would be a 'learning component'.

Reigeluth and Nelson (1997) suggested that "when teachers first gain access to instructional materials, they often break the materials down into their constituent parts. They then reassemble these parts in ways that support their individual instructional goals". In Wiley's (2000) opinion this initial step of decomposition could be bypassed if the instructors received instructional resources, learning objects, as individual components. This would potentially increase the speed and efficiency of online learning material development.

To achieve widespread use of learning object technology, many organisations have developed and promoted instructional technology standards. Without such standards, universities, corporations, and other organisations would have no way of assuring the interoperability of their instructional technologies, specifically learning objects. Standardisation of learning object metadata is discussed in chapter 5.

IEEE Learning Technology Standards Committee's (LTSC) definition of a learning object states (IEEE, 2002): "A learning object is defined as any entity, digital or non-digital, that may be used for learning, education or training." This definition has been criticised (Wiley, 2000) for being too broad, since by this definition any entity that has ever existed (person, historical event etc.) is a learning object. Another definition by Wiley (2000) of a learning object is "any digital resource that can be reused to support learning". This definition narrows LTSC's definition to contain only digitally deliverable entities. Sjunnesson (2003) defines learning object as "the smallest part supporting learning in a digital system of learning resources." This definition specifies the learning object as an atom of a larger system, making the features of learning resource management and design more important. On the other hand, the Educational Object Economy Foundation (EOE, 2003) defines learning object as an 'educational Java object', more precisely a Java applet available on the web. It seems that there exists as many definitions for a learning object as there are authors and organisations trying to employ this concept.

3.2 Goals behind LO research

Wiley (2002) mentions two main goals behind learning objects research:

- To improve the economics of online instruction.
- To enable pedagogical innovation.

Many people are interested in reusability and learning objects because they seem to offer a solution to the teacher bandwidth problem (reusability.org, 2003). 'Teacher bandwidth' is a term describing the number of students a teacher can service. In this sense teachers can be seen as bottlenecks that limit the number of students who can gain access to educational opportunity. Current LO research projects have somewhat different ideas how learning object repositories are to be used to support learning. One classification introduces three main categories of LO research (reusability.org, 2003):

1. Automation: intelligent computer program replaces teacher, creating course content automatically from a learning object repository.
2. By-hand: teachers find and use learning objects to create course content.
3. By-community: informal learning that takes place in social groups without a formal teacher.

The first approach tries to overcome the teacher bandwidth problem by replacing the teacher with an intelligent computer program, an intelligent agent that automatically selects and sequences learning objects for students. One purpose of the IEEE's LOM project (IEEE, 2002) is "to enable computer agents to automatically and dynamically compose personalised lessons for an individual learner”. Other organisations focusing in the area of automatical composition of learning material are, e.g., ADL/SCORM (http://www.adlnet.org/) and IMS (http://imsproject.org/).

The second approach leaves the responsibility for combining and creating learning objects to teachers and instructors. Presently this seems to be the most realistic way to approach the creation and reuse of learning objects, since there seems to be no consensus, e.g., on explicitly specified learning object metadata, which is a necessity for creating intelligent tutoring systems.

On the other hand, fully automated instructional systems have been widely criticised. Based on, e.g., the constructive paradigm of learning the construction of new knowledge is the most efficient way of acquiring new knowledge and, moreover, learners must construct new knowledge for themselves, since knowledge cannot be transferred from a person (or computer) to the learner intact. Modern learning theories emphasise the context in learning, such as ‘social context’ or ‘situatedness’. According to Wiley (2003) the instructional design behind learning objects is increasingly moving towards decontextualisation. This is true because of an inversely proportional relationship between the size of a learning object and its potential reuse. This means that the less specific the internal context of the learning object, the more instructional contexts it will fit in, thus having a greater potential for reuse. However, a simple sequencing of highly decontextualised educational resources, i.e. learning objects, does not produce a meaningful context for learning. This leads to discussion of suitable granularity of learning objects.

The third category is the informal, ‘by-community’-style learning that takes place in diverse social groups, collaborating to reuse existing resources in multiple ways. This approach is out of the scope of this thesis. Furthermore, one possibility for learning material creation is a situation where teacher and learners co-operatively create learning content from existing resources, e.g., learning objects. This is basically a combination of the ‘by-hand’ and ‘by-community’ categories.

3.3 Granularity of learning objects

When creating learning objects, at least two important design issues are inevitable: granularity and combination. Learning object granularity refers to the size of the learning object, while combination refers to the manner in which the learning objects are assembled to facilitate instruction (Wiley, 2000). As mentioned in the previous chapter, when creating teaching material teachers tend to break the source materials down into their constituent parts, finally reassembling these parts in ways that support their individual instructional goals. This description captures one of the basic notions behind the learning objects idea: ‘pre-deconstruct’ instructional media to increase the efficiency of instructional design (Gibbons et al., 2000).

The size of a learning object varies depending on several factors, e.g., the purpose and context of the learning object. Many organisations have released recommendations on the granularity of learning objects; the smallest learning object being, e.g., just a digital image and the largest a whole set of courses leading to a certificate. LO definitions give different meanings to the size of the learning object; IEEE Learning Technology Standards Committee´s (IEEE, 2002) definition can basically mean anything from a digital image to a whole course (talking about digital resources only), while the American Society for Training & Development (ASTD, http://www.astd.com) defines a learning object as “a collection of reusable information objects, overview, summary, and assessments that supports a specific learning objective”, making the learning object entity a relatively large piece of information.

Several learning object metadata specifications give instructions on how to express the granularity in the metadata section of the learning object definition. Both the IMS and LOM specifications have an ‘aggregation level’ element in their metadata element set that defines the functional granularity of the learning object. In the LOM specification the aggregation level 1 object is a piece of raw media, e.g., a digital image or some text. Level 2 objects are collections of level 1 learning objects, e.g., a lesson. Level 3 objects are sets of level 2 objects, e.g., courses, and finally level 4 objects are defined, e.g., as a set of courses leading to a certificate (IEEE, 2002, IMS, 2001c).

It is probably safe to say that there is no standard size for a learning object. This is, in fact, quite obvious, since there is not even a standard definition of what a learning object actually is. Furthermore, the 'learning object' paradigm is based on the object-oriented paradigm of computer science. In object-oriented design the granularity of an object can be anything from a small domain-independent function (i.e., a generic component) to a large domain-specific component (e.g., a database). It seems that the granularity of a learning object depends on the context and the purpose it is used for. However, when creating reusable components for teaching purposes, there has to be some way to define a suitable granularity framework recommendation to be used within the organisation where learning objects are created and used.

The amount of information in a learning object is a two-fold matter. Basically, the object should have a meaningful purpose to be used independently, but the context of the object should not be too large if the learning object is to be used with other learning objects in different contexts. As Wiley (2003) mentions, content developers tend to move towards decontextualisation when creating learning objects to achieve high potential for reusability, but highly decontextualised learning objects do not necessarily provide a meaningful context for learning. Modern learning theories state that learners need to fit new information to their existing frame of reference to achieve the 'deep learning' experience. Rich context helps learners make these necessary links between new information and their existing knowledge. For the learner to be able to benefit from the material the context has to be organised in a meaningful way.

3.4 Pedagogy of learning objects

Most learning object standards define the type of the learning resource based on, e.g., the teaching approach or pedagogical paradigm mainly used within the learning object. For example, in the Celebrate metadata specification (Celebrate, 2003) the learning object's 'Educational Learning Resource Type' element can be defined, e.g., as an assessment, drill and practise, exploration, and information resource. One or more values can be selected for each learning object. Similar elements can be found in LOM (IEEE, 2002), IMS (IMS, 2001b) and SCORM (ADL, 2001) metadata specifications.

However, the amount and quantity of pedagogy included in a learning object is not a self-explanatory thing. It is widely accepted that ‘drilling’ type instructional materials can be used to train routines in well-defined scenarios (simulations etc.), but they do not work well when teaching issues that require complex information processing. According to present understanding on human learning, when teaching complex issues and problems a more advanced, e.g., a problem-based approach would work better. The traditional ID design methods are even said to lead to surface learning rather than deep learning (Häkkinen, 2002). If traditional ID methods (e.g., drill and practise) are integrated tightly to a learning resource, it may become unusable in different contexts where some other teaching philosophy would be more suitable. It has even been said that “learning design and reusability are incompatible” (Downes, 2003). This means that if learning components contain information about their relations to other resources on the basis how they are to be sequenced later on, those components become unusable in different contexts.

However, for example the IMS Learning Design (IMS LD) specification tries to solve this problem by separating the learning design and content (IMS, 2003c). The IMS LD specification provides a language for expressing different pedagogies to be used within learning objects. Basically the IMS LD is an integration of the Educational Modelling Language (EML) work, developed by the Open University of the Netherlands, and several existing IMS specifications. The purpose of the specification is defined as: “the development of a framework that supports pedagogical diversity and innovation, while promoting the exchange and interoperability of e-learning materials" (IMS, 2003c).

According to Bannan-Ritland et al. (2001), to incorporate constructivist principles, a learning objects system must also be able to support learner-generated artefacts by incorporating learner contributions. This means that the learning objects should also be dynamic while they are used in VLE, so the learners can co-operatively or individually create new knowledge and attach it to the existing teaching material.

Friesen (2003) has stated that pedagogy as a whole is not something that can simply be understood as neutral in its relation to technology or technical specifications. Developers and designers will have to recognise and choose relevant (and probably differing) pedagogical solutions when creating learning objects, or risk pedagogical irrelevance. In my opinion also pedagogically neutral instructional resources are suitable to be used and reused, as long as the teacher or instructor compiling the teaching activity from the resources has the abilities to choose and implement a relevant pedagogical approach, according to the level of learners' previous task knowledge and the level of cognitive processing required by the task (see Figure 1). Teaching activities have been organised for decades, or even centuries from 'pedagogically neutral' sources of information, long before discussions of learning objects and standards. However, instruction material as described in ADL's (Advanced Distributed Learning) Sharable Content Object Reference Model (SCORM) Version 1.2 that "will adapt itself to the unique needs, abilities, background, interests and cognitive style of each learner" (ADL, 2001) does not seem to be a very realistic approach by means of existing technological solutions supporting content creation and e-learning platforms. This kind of automatically adaptive learning material was also the main target of Friesen's criticism.

4 XML – Extensible Markup Language

Most learning object approaches use XML (Extensible Markup Language) for describing learning objects and the corresponding metadata. Therefore, in this chapter some main concepts of XML are introduced. More practical approaches on use of XML, for example describing learning object metadata using XML syntax, are introduced in the following chapters.

Ever since the invention of the printing press, writers have made notes on manuscripts to instruct the printers on production issues, e.g., typesets. A collection of notes that conform to a defined syntax and grammar can be called a markup language. Computer applications, to be interoperable, need to use a common markup language (Birbeck et al., 2001).

XML was designed as a universal markup language for data interchange between applications. XML was developed by an XML Working Group in 1996. The latest updated XML specification was published by W3C in October 2000 (W3C, 2000). According to Birbeck et al. (2001), some of the the design goals for XML were:

- **Extensibility and separation of semantics and presentation.** XML provides a basic syntax but does not define the actual tags; the tag set can be freely extended. Furthermore, the XML syntax says nothing about the actual presentation of the XML data.

- **Simplicity and internalisation.** XML data can be used for computer-to-computer messages, as well as for human readable documents. Also, XML documents may use any of the world's different alphabets, scripts, and writing systems as defined in the Unicode 3.0 (http://www.unicode.org/) and ISO/IEC 10646 (http://www.iso.ch) standards.

- **Usable over the Internet.** XML, being a text format, can be easily transmitted over the Internet using the basic WWW-protocols such as HTTP or HTTPS. There is also a version of HTML represented in an XML-compatible form – XHTML 1.0 (http://www.w3.org/TR/xhtml1/).

4.1 Syntax

The XML syntax closely resembles the syntax of HTML documents, but there is an important difference between these two markup languages: HTML describes data and presentation, while XML only describes data and says nothing about the presentation. Data that conforms to the XML 1.0 REC Specification (W3C, 2000) is known as *well formed*. Data is well formed XML if, e.g., (Birbeck et al., 2001):

- All start-tags have matching end-tags, or are empty elements. For example: *<data>This is text</data>*, and *<data/>*

- Element tags do not overlap. This is forbidden: *<data>This is text<name>More text</data></name>*

- Characters used for markup are properly escaped. Five characters (<, >, &, ‘, “) have special meaning in XML mark-up. For example, the correct entity reference for < is *<*

- Elements form a hierarchical tree with a single root node. Only one root node can exist in a single XML document.

An example of a well-formed and valid XML document containing information about students is presented in Example 1.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE students SYSTEM "students.dtd">
<students>
   <student>
      <name>Olli Opiskelija</name>
      <studyline>Computer Science</studyline>
      <starting_year>2000</starting_year>
   </student>
   <student>
```

```
        <name>Maija Meikäläinen</name>
        <studyline>Sociology</studyline>
        <starting_year>1999</starting_year>
    </student>
</students>
```

Example 1. XML document containing information about students.

The first line is the *XML declaration*, which is optional in XML documents. However, it is best to always include an XML declaration in an XML document (Birbeck et al., 2001). The XML declaration identifies the data as XML and states the XML version and encoding that are used within the document. The second line is the *Document Type Declaration* stating that the document is to be validated using the given DTD (*Document Type Definition*). The third line is the root node of the document that contains all data elements.

4.2 DTD – Document Type Definition

DTD (Document Type Definition) is a set of rules on how XML data should be structured. A DTD can be incorporated within XML data, or it can exist as a separate document. An XML document that conforms to the XML 1.0 REC Specification (i.e. is *well formed*) and also complies with syntax, structural, and other rules as defined in a DTD, is called *valid* XML data.

Separating the XML data description to a DTD allows all applications using the data share a single description of the data, known as the *XML vocabulary*. A group of XML documents that share a common XML vocabulary is known as a *document type*, and each individual document that conforms to a document type is a *document instance* (Birbeck et al., 2001). A DTD for the XML file about students in Example 1 is presented in Example 2.

```
<!ELEMENT students (student*)>
<!ELEMENT student (name, studyline, starting_year)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT studyline (#PCDATA)>
<!ELEMENT starting_year (#PCDATA)>
```

Example 2. DTD for student data.

The DTD states that the root element of the XML document is ‘students’ and it can contain multiple ‘student’ elements. Each ‘student’ element has three elements: ‘name’, ‘studyline’, and ‘starting_year’. The content of these elements must be textual data.

4.3 XML Schema

In addition to DTD, another way to define the structure of XML documents is to use XML Schema. Unlike DTDs, the XML Schema uses XML 1.0 syntax and can be edited with generic XML tools. With XML Schema it is possible to define XML data descriptions more accurately than by using DTDs. For example, XML Schema allows the use of common data types (e.g., value=”decimal”) and precise specification of element structures (e.g., a certain element can appear 1 to 7 times).

An XML document that is validated against an XML Schema is slightly different than the previous XML document. For example, the beginning of such document could be:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<students xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xsi:noNamespaceSchemaLocation="students.xsd">
…
```

In this example the second line of the document states that this specific XML document is to be validated against an XML Schema file named ‘students.xsd’.

An XML Schema for the XML file about students in Example 1 is presented in Example 3.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="name" type="xs:string"/>
<xs:element name="studyline" type="xs:string"/>
<xs:element name="starting_year" type="xs:gYear"/>
<xs:element name="student">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="name"/>
   <xs:element ref="studyline"/>
   <xs:element ref="starting_year"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>
<xs:element name="students">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="student" maxOccurs="unbounded"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>
</xs:schema>
```

Example 3. XML Schema for student data.

In this XML Schema document the data type of the element ‘starting_year’ is defined as *gYear*, meaning that the element content has to be presented using the Gregorian calendar year. Otherwise this XML Schema is corresponding to the DTD given in the Example 2.

If the number of students attending a class should for some reason be restricted to 15-30, the corresponding way to express this within the previous XML Schema in Example 3 would be:

```
<xs:element ref="student" minOccurs="15" maxOccurs="30"/>
```

The benefit of using XML Schemas instead of DTDs is that they provide much more control over the content of a document instance. This means, for example, that controlling the element content can be left for the XML parser instead of writing the actual control code into an application. This should reduce the possibility of errors when creating complicated software applications that use XML for data storage and interchange.

4.4 XML Namespaces

When XML documents are created and used together with other XML documents with a shared vocabulary, there is a possibility of element name collisions between different documents. For example, just about any XML data could contain an element named <name> or <title>. As Birbeck et al. (2001) state: "XML Namespaces provides a compound name syntax that extends the definition of XML 1.0 names to ensure that unique names can be generated for shared vocabularies". Basically, XML Namespaces is nothing more than a named collection of names. A practical example on the use of XML namespaces with RDF is introduced in section 5.2.

4.5 XHTML

XHTML (http://www.w3.org/TR/xhtml1/) is an abbreviation for Extensible HyperText Markup Language. The XHTML 1.0 implementation obeys all of the grammar rules of XML, while conforming to the vocabulary of HTML (Birbeck et al., 2001). W3C XHTML 1.0 recommendation (W3C, 2002) mentions some of the benefits for using XHTML:

- XHTML documents are XML conforming. As such, they are readily viewed, edited, and validated with standard XML tools.

- XHTML documents can be written to operate as well or better than they did before in existing HTML-conforming user agents as well as in new, XHTML 1.0 conforming user agents.

- XHTML documents can utilise applications (e.g., scripts and applets) that rely upon either the HTML Document Object Model or the XML Document Object Model, DOM (http://www.w3.org/TR/REC-DOM-Level-1/).

Definitions of strictly conforming XHTML 1.0 documents can be found in the W3C XHTML 1.0 recommendation (W3C, 2002). An example of an XHTML 1.0 document is presented in Example 5.

4.6 XPath and XSLT

The XML Path Language (XPath) is a querying language that is used to address specific parts of an XML data objects as nodes within the document tree. The XPath handles XML data as paths within an abstract hierarchical tree structure of nodes (Birbeck et al., 2001). XPath allows selected or filtered information from the source XML data or document to be exchanged or displayed, making it a very important factor in data interchange between computers or applications.

In practise, the most important use of XPath at present is within XSLT (Extensible Stylesheet Language Transformations), whether in transformation of XML to HTML or transformation of one form of XML to another (Birbeck et al., 2001). XSLT is a language for specifying transformations of XML documents.

For example, the path to an individual student's name in Example 1 is */students/student/name*, and there can be several 'student' elements within the XML document. Therefore, the application needs to loop through all 'student' elements to be able to extract every student's name from the document. XPath expressions for extracting this information are presented in Example 4.

To extract some information from the XML document in Example 1 to another format, for example to an HTML file, an XSLT stylesheet is created. Within the stylesheet in Example 4, XPath expressions are used to extract wanted information from the source document.

```
<?xml version="1.0" encoding="iso-8859-1"?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">

<xsl:output method="xml" indent="yes" encoding="iso-8859-1" doctype-
public="-//W3C//DTD XHTML 1.0 Transitional//EN" doctype-
system="http://www.w3.org/TR/xhtml1/DTD/transitional.dtd"/>

<xsl:template match="/">

  <html>

  <body>

    <h2>Student names</h2>

      <xsl:for-each select="students/student">

        <h3><xsl:value-of select="name"/></h3>

      </xsl:for-each>

  </body>

  </html>

</xsl:template>

</xsl:stylesheet>
```

Example 4. XSLT stylesheet.

The stylesheet goes through each ‘student’ element in the XML file and writes the content of the ‘name’ element into the result document. The expression `<xsl:template match="/">` sets the context node of the XPath expression as the document root (the parent of the ‘students’ element in the source XML document). The expression `<xsl:for-each select="students/student">` loops through all ‘student’ elements that are children of the ‘students’ element. Finally, the expression `<xsl:value-of select="name"/>` extracts the content of the ‘name’ element from each ‘student’ element.

All other text, including the HTML markup elements, is written into the result document as they appear within the stylesheet.

XML can be transformed for presentation using an XSL Transformations tool, such as Saxon (http://saxon.sourceforge.net), JAXP (http://java.sun.com/xml) or Xerces (http://xml.apache.org). An XHTML document that is created by XSL transformation from the XML document in Example 1 using the XSLT stylesheet in Example 4 is presented in Example 5.

```
<?xml version="1.0" encoding="iso-8859-1"?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/transitional.dtd">

<html>

<body>

<h2>Student names</h2>

<h3>Olli Opiskelija</h3>

<h3>Maija Meik&auml;l&auml;inen</h3>

</body>

</html>
```

Example 5. XHTML document.

The first two lines of the document in Example 5 identify the document as XML data and, more specifically, as XHTML 1.0 data.

Since the source XML document contains non-ascii characters (*"ä"*), the *iso-8859-1* encoding is used in both the source document and the stylesheet. That way the XML parser transforming the data into XHTML knows how to handle these characters, for example, the character *"ä"* is encoded as *"ä"* within the XHTML document. This means that most browsers are able to validly display the result XHTML document.

XML syntax is not meant to be written by hand, but using a suitable application. Traditionally, user-end applications for authoring XML documents have been significantly expensive. However, for example JEdit (http://www.jedit.org/) is a freely available, open source application for text editing that provides possibilities for authoring XML documents with support for validating documents against a DTD. A more advanced and also freely available XML editor from Altova Corp., Authentic 2004 (http://www.xmlspy.com/products_doc.html), has a so-called WYSIWYG-style interface, which hides the technical aspects of XML from the user. One specific application for creating XML-syntax for learning object metadata is tested in chapter 8.

5 Learning Object Metadata

Metadata is generally known as data about data, or information about information. Usually metadata is written in some markup language and stored internally (metadata is part of the object) or externally (metadata is attached to the object). The approach to store metadata externally is likely to be most practical for many non-textual resources, and is increasingly used for text as well, primarily to support easier maintenance and sharing of metadata.

In the web design the W3C definition of metadata is: “metadata is machine understandable information about web resources or other things” (Berners-Lee, 1997). In this sense the ‘machine understandable information’ is the main idea behind metadata standardisation and usage. Although all digitally stored information is ‘machine readable’, it certainly is not ‘machine understandable’, since computers, unlike humans, don’t make distinctions of the context of the data they process without any specific instructions. Learning object metadata is often expressed as systematic description of learning objects to enable searching and administration. Standardisation of learning object metadata is discussed in section 5.4.

In this thesis the focus is on machine understandable metadata. This kind of metadata has to be well defined, i.e. it has well defined semantics and structure, for computers and intelligent agent applications to be able to read the metadata. One powerful way to define this sort of metadata is the Resource Description Framework (RDF), which is discussed in section 5.2. One way to utilise the RDF in practise is to use the Dublin Core metadata standard, which is introduced in section 5.1.

5.1 Dublin Core Metadata Standard

The Dublin Core Metadata Initiative (DCMI) is an organisation promoting interoperable metadata standards and developing specialised metadata vocabularies for describing resources that enable more intelligent information discovery systems. DCMI has produced the Dublin Core metadata standard, which is a simple yet effective element set for describing a wide range of networked resources. (Dublin Core, 2003)

The Dublin Core metadata standard can be described as a small language for making statements about resources. It consists of two classes of terms: elements (nouns) and qualifiers (adjectives). These terms can be arranged into a simple pattern of statements. The Dublin Core basic element set consists of elements describing the content (e.g., Title, Description), intellectual property (e.g., Creator, Rights), and Instantiation (e.g., Date, Language). Each element is optional and may be repeated. The Dublin Core Element Set has been translated to many languages, including Finnish (Hillman, 2003). More information on the practical usage of the Dublin Core will be given in the next section.

5.2 Resource Description Framework - RDF

Resource Description Framework is a foundation for processing metadata and providing interoperability between applications that exchange machine-understandable information. The basic RDF data model consists of three object types: resources (all things described by RDF expressions), properties (aspects, characteristics, attributes or relations used to describe a resource), and statements (a specific resource together with a named property and the value of that property) (RDF Specification, 1999).

The basic structure of a RDF statement consists of three parts:

- the subject (resource, identified by URL)
- the predicate (attribute, e.g., the type of metadata)
- the object (value, e.g., the value of the metadata)

The RDF statement can be described as a diagram, as shown in Figure 2.

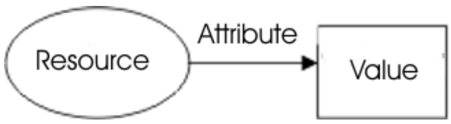

Figure 2. RDF statement diagram.

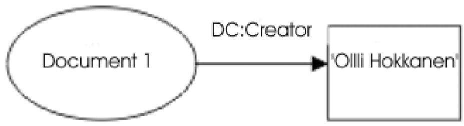

Figure 3. Dublin Core RDF statement.

For example, the data model for expressing that “Olli Hokkanen is the author of Document_1” using the Dublin Core element set is illustrated in Figure 3. The corresponding syntactic way of expressing this data model is presented in Example 6.

```
<?xml:namespace ns = "http://www.w3.org/RDF/RDF/" prefix ="RDF" ?>

<?xml:namespace ns = "http://purl.oclc.org/DC/" prefix = "DC" ?>

<RDF:RDF>

  <RDF:Description RDF:about = "http://uri-of-Document_1">

    <DC:Creator>Olli Hokkanen</DC:Creator>

  </RDF:Description>

</RDF:RDF>
```

Example 6. Dublin Core RDF statement.

The first two lines in Example 6 express that this statement uses XML namespaces to identify the use of the RDF and Dublin Core schemas. The RDF schema is declared as a mechanism for the declaration of the necessary vocabulary needed for expressing the data model. The Dublin Core schema is declared in order to utilise the vocabulary defined by the Dublin Core metadata standard. The element `<RDF:RDF>` marks the boundaries in this document where the content is explicitly intended to be mappable into an RDF data model instance.

Sjunnesson (2003) mentions four areas where RDF statements appear important in educational area:

- Intelligent software agents finding relevant information.

- Personal annotations of any learning resource.
- Collaborative and distributed authoring and course construction.
- Reuse of learning material.

Systematically and explicitly described metadata along with commonly used and formally defined vocabularies – ontologies – is bound to become an important issue with creation, use, and maintenance of learning object repositories.

5.3 Ontologies

Ontologies have become common in the World Wide Web in recent years. An ontology defines a common vocabulary for people who need to share information in a domain. It includes also machine understandable definitions of basic conceptions in the domain and relations among them (Noy & McGuinness, 2001). Shah et al. (2002) define an ontology as: “ontology is an explicit specification of a representational vocabulary for a domain; definitions of classes, relations, functions, constraints, and other objects”. The ontologies on the web range from large taxonomies categorising web sites (e.g., http://www.yahoo.com) to categorisations of products for sale and their features (e.g., http://www.amazon.com). Noy and McGuinness (2001) list some of the reasons for developing an ontology:

- To share common understanding of the structure of information among people or software agents. For example, software agents can extract information from several different sites that share and publish the same underlying ontology of the used terms.
- To enable reuse of domain knowledge. Several existing ontologies can be integrated to describe a special domain of interest.
- To make domain assumptions explicit. The domain assumptions can be easily changed afterwards if the domain knowledge changes. Hard-coding assumptions

about the domain with some programming language makes these assumptions harder to find, change, and understand later on than by using a specified ontology.

- To analyse domain knowledge based on machine-interpretable definitions of the domain concepts.

Learning object repositories, if they are to be interoperable and efficient, have to be based on a taxonomy that is widely accepted for defining educational resources. Controlled vocabularies are often called *taxonomies* (Birbeck et al., 2001).

To clarify the difference between a *vocabulary*, a *taxonomy*, and an *ontology*, these terms and their relations can be summarised as:

- A vocabulary is a commonly used set of terms.
- A taxonomy is a controlled vocabulary (Birbeck et al., 2001). Terms in a taxonomy are usually organised into a hierarchical structure.
- An ontology is a common vocabulary that also includes machine understandable definitions of basic conceptions in a specific domain and relations among them (Noy & McGuinness, 2001).

Basically, in the educational context ontology development means the categorisation of educational materials and their relations so that they can be effectively stored and used in different contexts. At the present moment global (or even national) use of commonly accepted keywords for describing learning objects and resources is still under development.

Several libraries of reusable ontologies are available on the Web, for example the DAML ontology library (http://www.daml.org/ontologies/) and the Ontolingua ontology library (http://www.ksl.stanford.edu/software/ontolingua/). A number of publicly available commercial ontologies can also be used, for example the Rosettanet (http://www.rosettanet.org), a consortium of several information technology, electronic components, and telecommunications companies. In education, ontology development also means that the educational taxonomic ontologies have to merge with the commercial and other kinds that may intervene with education (Sjunnesson, 2003).

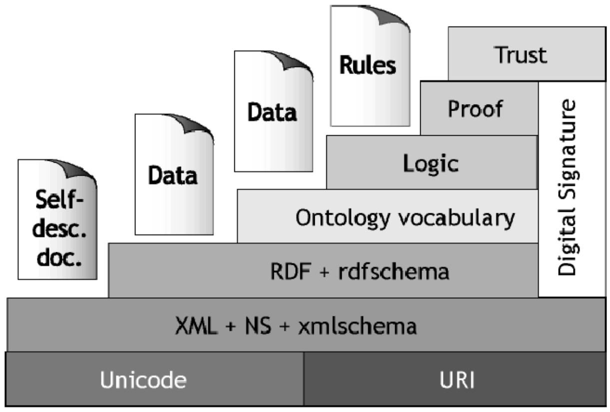

Figure 4. Original vision of the Semantic Web by Berners-Lee (Shah et al., 2002).

Figure 4 illustrates the original idea of the Semantic Web (http://www.w3.org/2001/sw/) and relations between the concepts introduced in chapters 4 and 5. XML technologies provide the necessary means to declare and use simple data structures, which are stored as machine-readable XML documents. Since XML is defined only at the *syntactic* level, RDF and ontologies are used to define necessary formal *semantics* over the XML documents. (Shah et al., 2002)

5.4 Learning object metadata standardisation

To achieve interoperability between different learning and business systems, the standardisation of metadata elements describing the resources is needed. Several standards for describing learning object metadata have been developed through collaboration between the private and public sectors. One of the first attempts to standardise learning object metadata descriptions is the LOM standard.

5.4.1 LOM

The IEEE Learning Technology Standards Committee (LTSC) has been developing the Learning Object Metadata (http://ltsc.ieee.org/wg12/) standard since 1997. LOM defines a set of metadata elements that can be used to describe learning resources. The LOM standard has also been translated to Finnish by TIEKE (Finnish Information Society Development Centre, http://www.tieke.fi/). The IEEE LOM Draft Standard is intended to support consistent definition of metadata elements across multiple implementations, but it does not include information on how to represent metadata in a machine-readable format, which is a necessity for exchanging metadata (IMS, 2001a).

The LOM standard focuses on the minimal set of attributes needed to allow learning objects to be managed, located, and evaluated. The basic fields and entity types of the standard can be extended, and the fields can have a status of obligatory (must be present) or optional (can be absent). Relevant attributes of learning objects to be described include type of object, author, owner, terms of distribution, and format. Where applicable, Learning Object Metadata may also include pedagogical attributes such as; teaching or interaction style, grade level, mastery level, and prerequisites. The standard does not determine how these features are implemented. Some aims of the LOM project are, e.g., (IEEE, 2002):

- To enable learners or instructors to search, evaluate, acquire, and utilise learning objects, and to enable the sharing and exchange of learning objects across any technology supported learning systems.

- To enable the development of learning objects in units that can be combined and decomposed in meaningful ways and to enable computer agents to automatically and dynamically compose personalized lessons for an individual learner.

- To enable a strong and growing economy for learning objects that supports and sustains all forms of distribution: non-profit, not-for-profit and for profit.

- To enable education, training, and learning organisations, both government, public, and private, to express educational content and performance standards in a standardised format that is independent of the content itself.

- To define a standard that is simple yet extensible to multiple domains and jurisdictions so as to be most easily and broadly adopted and applied and to support necessary security and authentication for the distribution and use of learning objects.

LOM has been the starting point for several other learning object standards for more defined purposes, for example the CELEBRATE and IMS definitions.

5.4.2 CELEBRATE

CELEBRATE (Context eLearning with Broadband Technologies) is a large-scale (€5m), 30 month (June 2002 - November 2004) project supported by the European Commission's IST (Information Society Technologies) Programme (Celebrate, 2003). The Celebrate project involves 500 schools across Europe and 22 partner organisations. These 22 partners in the project include European Ministries of Education along with universities, educational publishers (e.g., SanomaWSOY), and technology suppliers such as Sun Microsystems all over Europe. The project's aim is to provide schools with access to an online content repository that includes the ability to share learning objects and 'components' ('components' meaning multimedia assets on Celebrate's web pages; information objects elsewhere in this thesis) that can be used to create learning objects.

It is intended that Celebrate will act as a catalyst for the European eLearning content industry (the entire value-chain including content owners, publishers, broadcasters, national school networks and ICT platform vendors). The Celebrate project deals with several research issues, which will be tested during the project. For example, one aim of the project is to explore how the learning object methodology can be applied to educational activities and services as well as learning materials. Celebrate is also going to investigate how learning objects can be handled by a 'new generation' of Learning Content Management Systems (LCMS) from different vendors and to test the interoperability of these systems in a real-life demonstrator. (Celebrate, 2003)

The Celebrate project web pages state the aims of the project as: ”CELEBRATE will particularly analyse the extent to which new, more flexible forms of content development and distribution (based on reusable LOs) impact upon the learning process and support a new pedagogy for eLearning in schools based on constructivist learning models. In particular, the project will assess how the use of LOs encourages the development of key digital age skills such as collaborative working, creativity, multidisciplinarity, adaptiveness, intercultural communication and problem-solving." (Celebrate, 2003)

One dimension of the project is to create a metadata definition to be used within the Celebrate project. This Celebrate Metadata Application Profile defines mandatory, recommended, and optional elements of LOM Data Model and extends it by defining new elements and new vocabularies for Celebrate project. These new elements are 'Learning Principles' and 'CELEBRATE Digital Rights'. New vocabularies have been defined for 'Learning Resource Type', 'Intended End User Role' and 'Context' in 'Educational' category and some refinements have been made to 'Language' value space and 'Typical Age Range' value space.

These research issues are definitely crucial to the whole learning objects issue if it is to succeed. There are already some learning objects available via the Celebrate web pages (http://celebrate.eun.org), most of which are multimedia applications implemented with Macromedia Flash technology (http://www.macromedia.com/software/flash/). In the year 2004 the Celebrate project is going to open a portal for submitting learning objects into a repository and creating suitable metadata for them. Some results and experiences of the Celebrate project are to be published in the year 2004.

5.4.3 IMS

The IMS Global Learning Consortium (http://www.imsglobal.org) develops open technical specifications for interoperable learning technology. IMS specifications are available to the public at no charge, and no fee is required to implement the specifications. IMS is a worldwide non-profit organisation that includes more than 50 members from every sector of the global e-learning community. They include, e.g., hardware and software vendors, educational institutions, publishers, government agencies, systems integrators, and multimedia content providers.

IMS has contributed to the learning object metadata standardisation movement by developing descriptions and guides on how to represent metadata in XML and how to implement learning object repositories. IMS has also published taxonomy and vocabulary lists to be used with LOM metadata descriptions. IMS previously released a document describing a short list of LOM metadata elements (called the IMS Core) that was hoped to simplify meta-data implementation. However, the IMS Core specification did not meet its expectations, so IMS has dropped all references to the IMS Core specification from present standards. (IMS, 2001a)

The IMS Learning Resource Meta Data XML Binding specification describes how the IEEE LOM element set (with some IMS related extensions) is represented in XML. The root element `<lom>` contains following elements (IMS, 2001b):

- `<general>` General information about the learning object.
- `<lifecycle>` Features related to the history and current state of this learning object and those who have affected this learning object during its evolution.
- `<metametadata>` Information about the metadata instance itself.
- `<technical>` The technical requirements and characteristics of the learning object.
- `<educational>` Conditions of educational use of the resource.
- `<rights>` Conditions of use of the resource.
- `<relation>` Features of the resource in relationship to other learning objects.
- `<annotation>` Comments on the educational use of the learning object, for example suggestions and comments on instructional methods.
- `<classification>` Characteristic of the resource by entries in classifications.

An illustration of the usage of `<general>`-element is shown in Figure 5.

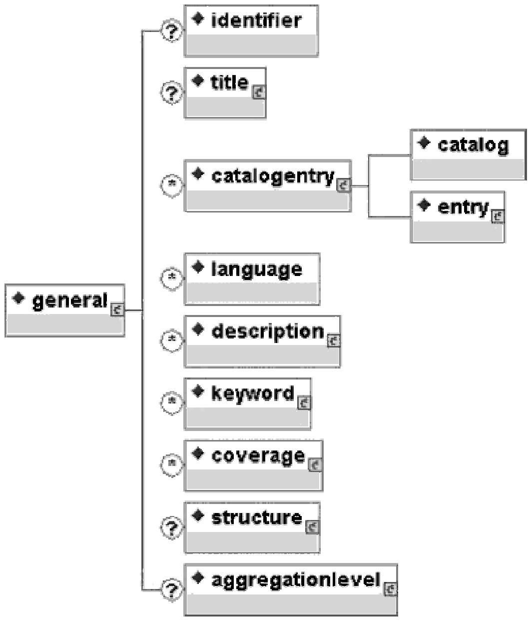

Figure 5. IMS Learning Resource Meta-Data XML Binding <general>-element description (IMS, 2001a).

In Figure 5 all elements tagged with a question mark (e.g., `<title>`, `<aggregation level>`) can appear once or not at all inside the `<general>` element. Elements tagged with an asterisk character can appear several times or not at all.

The XML-syntax describing a course using the IMS Learning Resource Meta Data XML Binding specification is presented in Example 7. In this example the `<general>`-element contains only sub-elements `<title>`, `<description>`, and `<keyword>`.

```
<general>

  <title>

    <langstring xml:lang="en">Introduction to Java</langstring>

  </title>

  <description>

    <langstring xml:lang="en">Introductory course to Java and object-
oriented programming </langstring>

  </description>
```

```
  <keyword>
    <langstring xml:lang="en">java</langstring>
  </keyword>
</general>
```

Example 7. IMS XML Binding example.

The `<langstring>`-element along with the `xml:lang` attribute declares the language used inside the element. Naturally, the IMS XML Binding is not meant to be written by hand, but by using some metadata application. The usage of an open source application (ReLoad) to create an IMS Content Package (IMS, 2003b) is discussed in section 7.2.

One valuable source of information about learning object standards and especially the IMS specifications is CETIS (The Centre for Educational Interoperability Standards, http://cetis.ac.uk/), which is an organisation advising Universities and Colleges in the UK about the strategic, technical, and pedagogic implications of educational technology standards. CETIS also manages UK implementation groups examining the use of IMS specifications, and promotes discussion on practical implementations of educational technology approaches and standards.

6 Thoughts on learning objects

In this chapter an example scenario that shows why learning objects have become an issue of interest within the field of learning technology is introduced. The granularity and pedagogy of learning objects are also discussed. Furthermore, I try to clarify different definitions for learning objects.

6.1 Learning objects in practise

To summarise the situation why the talk about learning objects has emerged, the following three imaginary cases represent my understanding of the present situation. These cases illustrate a scenario in which the learning object approach could be used to benefit the teachers and learners alike.

A teacher's objective is to create material for a course taught in classroom, VLE (virtual learning environment), or a combination of these. If suitable material cannot be found straight away (e.g., a book meant to be used to teach the issue), the teacher usually has to create the materials he or she is going to use throughout the course. At comprehensive schools and similar institutions just one specific book to teach some issue (along with ready made instruction material for the teacher) is often used, but this is seldom the case at university level. Of course, at comprehensive schools teachers often produce and exploit material in addition to the 'official' books and materials.

Case 1. Traditionally, when using VLEs or when creating normal classroom teaching material, the teacher or instructor gathers information from wide variety of resources (books, internet etc.), chops the information to usable sized components (lessons etc.), adds pedagogy to these components, and, finally, constructs the actual teaching materials to be delivered to the learners. The main problem behind this approach is that every teacher or instructor has to do the same information gathering, chopping and reconstructuring process individually, and the end result (e.g., lesson notes or materials in VLE) is usually not reusable by other teachers or instructors. Therefore, some means to make this process more effective would be extremely useful for organisations and individuals in the educational sector.

Case 2. As mentioned before in this paper, the vision behind learning objects is the learning object repository where a large number of well-designed and pedagogically sound learning objects can be found and reused in several contexts. Teacher or instructor would simply make a search in this repository for suitable learning objects and transfer them to the new environment (classroom or VLE). Even software agents could be used to construct digitally deliverable courses if the metadata within the objects in the repository contains sufficient information about the relations of the learning objects. Also individual learners could use this repository and software agents to search for learning material according to their needs. This approach has been widely criticised to be too optimistic as mentioned in section 3.2.

Case 3. It seems, based on the issues mentioned above, that the most realistic approach for university level education would be to create an information object (information object meaning a learning object without strictly defined pedagogies) repository rather than a learning object repository. This means that the objects in the repository contain information in a suitable format that can be used to teach something, along with the necessary metadata in a structured format so queries can be made to the objects database and possible relations between objects can be expressed. However, most pedagogical issues are to be taken care of by the teacher or instructor using these instructional objects. Pedagogy as a whole can be a too complicated issue to be expressed technically in a reasonable amount of time. Furthermore, a competent teacher or instructor should be capable of applying a suitable teaching approach in different contexts without detailed instructions on how to use a certain instructional media, e.g., a learning object.

For example, the teacher chooses suitable resources from a set of learning objects concerning the subject at hand. Then she or he sequences them in a meaningful way and adds a suitable pedagogical approach, according to the context of the subject, the level of learners' previous knowledge, and the level of required cognitive skills. The pedagogical approach could be a groupwork, small research based on the material, or some suitable teaching method used in a classroom etc. The teacher could also add additional materials to support the usage of existing learning objects, maybe even create new learning objects in the repository to be used later on.

The sequencing of the learning objects most likely requires some assembling in a sense that in reality the learning objects can not be attached together without any extra effort, for example binding the objects together into a meaningful entity with additional textual contribution. Learning material very commonly contains textual references to previous subjects to help learners realise connections between different topics (e.g., “As mentioned in the previous chapter, …”). If learning objects are meant to be used as individual entities, these textual references naturally should not exist in the object itself (e.g., “As mentioned in *<name_of_another_LO>*”). This affects considerably the way that information can be presented in a learning object if it is to be automatically sequenced to be used in several contexts.

At first, sequencing and modifying learning objects by hand seems to be an ineffective solution compared to a fully automated system where learning objects could be chosen and sequenced based on certain criteria (e.g., “artificial intelligence”). The teacher has to evaluate the quality and suitability of the learning objects and think about how to use them in the best way to support teaching. Basically, the teacher has to process the data, internalise the content, and assemble the actual teaching material. However, during this process the teacher probably gets a deeper understanding of the material and it’s quality than if he or she would just be receiving and using a ready made course structure without any personal commitment. Personally, I would not want to attend a lecture where the lecturer sees the teaching materials (slides etc.) for the first time at the lecture, or attend a course in a VLE where the material is automatically created and sequenced by a computer program and the teacher’s role is reduced to an administrator. This kind of automated instruction, which was the ideal form of efficient education of the masses according to the learning conceptions of the 70’s, does not fit well into modern understanding of human learning.

Using computerised systems for automatic generation of learning content from, e.g., learner contributions or databases and exploiting the results as the basis for co-operative work in a VLE can be a very motivating and rewarding way to facilitate instruction for the learners and teachers alike. However, trying to replace teacher from the process of teaching with technical substitutes available at the present moment is not likely to succeed.

6.2 Practical granularity and pedagogy of learning objects

As mentioned in section 3.3, learning object granularity (i.e. the size of a learning object) is a complicated issue and depends on several factors. In addition to the issues mentioned before, I would also point out that the concept of a learning object, as well as suitable size and amount of metadata, is probably different depending on the author who creates such entities. A teacher with limited resources and computer skills probably does not want to fill in dozens of elements defined in various metadata specifications to describe learning content, or create massive teaching entities with simulations and assessments when creating teaching material (lectures etc.). Likewise, a subject specialist group developing reusable digitally deliverable learning components in a larger scale has the resources to make use of all the definitions, which should help the use of these resources in sophisticated environments. Perhaps it is best to try to find the core concepts of the content and use them as the basis for individual learning objects, expanding the size and complexity of each learning object according to available resources (mainly time reserved for implementation).

Furthermore, in my opinion, it is not a necessity to include specific or restrictive pedagogical instructions (e.g., assessments, tests) in a learning object, since this considerably restricts the choice of the pedagogical paradigm to be used in instruction. However, as mentioned in section 3.4, technically simple applications of modern educational theories (e.g., activating questions and real life based problems) do probably not restrict the usage of a learning object in different contexts. Integrated simulations, assessments, or multiple-choice questions, on the other hand, could decrease reusability potential if they cannot easily be departed from the learning object if needed. Altogether, it is probably best to separate pedagogical approaches from the actual information content when designing and creating learning objects.

The IMS Learning Design specification (IMS, 2003c), which tries to separate learning design and content, eventually turning the pedagogical design into a reusable component, tries to realise the issue. However, the IMS LD specification, as the other specifications dealing with Learning Objects, is quite a complex specification, so powerful tools and applications are required before a non-technical person can implement the XML-syntax defining a specific IMS Learning Design element.

As mentioned in section 3.4, Bannan-Ritland et al. (2001) have stated that, to incorporate constructivist principles, a learning objects system should be able to support learner-generated artefacts by attaching learner contributions to learning objects. However, if the learning objects are used in a virtual learning environment, learner contributions could as well be attached to the VLE's database instead of the actual learning objects. Of course the contributions can later on be transformed to new learning objects, if needed. In practise this means that learning objects do not have to be dynamic in relation to learner-generated artefacts. In my opinion learning objects should rather be seen as any other source of information that can be imported into a VLE or other medium supporting learning.

6.3 LO, RLO or RIO?

In this thesis the discussion is about learning objects (LO). However, the reusable components of instruction have several definitions and several names. The same acronym (e.g., LO) can have several different meanings according to the author, or different terms can mean the same concept.

ASTD's (American Society for Training & Development) online e-learning magazine (ASTD, 2003) defines a reusable information object (RIO) as "a collection of content, practice, and assessment items assembled around a single learning objective. RIOs are built from templates based on whether the goal is to communicate a concept, fact, process, principle, or procedure." Likewise, a reusable learning object (RLO) is defined as "a collection of RIOs, overview, summary, and assessments that supports a specific learning objective." In ASTD's definition a reusable learning object is a large piece of information, implemented to support a specific learning objective and concluded from smaller objects, RIOs. And, to make things even more complicated, a reusable information object, according to ASTD, is a collection of content, practise, and assessment items. This definition of a RIO is considerably wider than for example Wiley's or Sjunnesson's definitions of a learning object (Wiley, 2000, Sjunnesson, 2003). Furthermore, the RIO definition, being a collection of a concept, fact, and a process, is a clear application of the traditional instructional design method developed by Gagne. As mentioned in section 2.2, the problems of traditional ID methods and models become obvious in complex and unique learning goals and activities (Häkkinen 2002). For example, assessments that are used to test and ensure that skills have been acquired, do not necessarily work well with complex information contexts.

For clarity's sake I have chosen to focus my discussion in this thesis to learning objects defined as a (digitally deliverable) reusable atom of instruction. ASTD's concept of a learning object is too large to be reused in several contexts. Likewise, narrowing the concept of a LO to cover all digital entities (e.g., images) as Wiley (2000) has stated does not fit well into the concepts of modularity, separation of content and context, and reusability borrowed from object-oriented software design. A digital image is just a digital image, no matter what definition it is given to.

7 Learning material repositories

The main idea behind learning object design and research is the ideal of a learning object repository, a renewing source of information that contains reusable components (learning objects) for creating teaching and learning materials, lectures, whole courses etc. Two different views on how a learning object repository could be created and maintained are commercial (for-profit) and free (non-profit) approach. In the commercial approach so called ‘subject experts’ get paid for creating material for the repository, and the access to using the repository is restricted only for paying customers, institutions etc. The driving power behind this approach is to create a growing economy based on learning objects, where structured and standardised information chunks, learning objects, would be the goods delivered for paying customers as in any other economy.

The other approach is the ‘open source’ way of creating new knowledge. Each participant using the repository could contribute new learning objects for the repository and the access to learning object is free, available to anyone. Naturally, combinations between these two approaches have been developed, for example, the repository could be accessed only inside organisation’s intranet.

A considerably different approach is to use peer-to-peer technology to create, store and maintain learning objects. One such approach is the POOL (http://www.edusplash.net) project (Portal for Online Objects for Learning), which is funded by Canada’s Research and Innovation Network. Hatala and Richards (2002) state the benefits of peer-to-peer approach as: “for learning objects this means that individual instructors, if provided with the standard metadata and communication protocol, can develop and store their materials so that others may directly search and access their public materials, or become aware of semi-public materials which the individual may wish to negotiate consideration for use. Individuals may also store private materials that are under development, or are not intended for mass consumption”.

7.1 Existing learning material repositories

In this chapter two freely available teaching and learning material collections or repositories available on the web are reviewed. These collections consist of metadata describing the content while the actual material is stored on the same web server or somewhere else on the web. Usually the metadata descriptions provide access to complete courses or lectures, not to 'reusable learning objects'. Basically, most existing repositories on the web are a collection of sophisticated links to teaching resources, e.g., lecture slides, not to actual learning objects within the repository. These learning material collections are evaluated here on the basis of reusability (copyright issues, usage potential etc.). For a more detailed survey of learning object repositories see, e.g., (Neven & Duval, 2002). Digital libraries that have become one of the main sources of information for learning content creation are also discussed.

7.1.1 Merlot

MERLOT (Multimedia Educational Resource for Learning and Online Teaching, http://www.merlot.org) is a collection of learning materials designed primarily for faculty and students of higher education. Access to materials is open to all, but only members of the Merlot community can add their own materials in the collection. Membership to the Merlot project is free of charge.

Merlot project conducts structured peer reviews of all materials. The purpose of the peer review process is to allow users decide if the materials will work on their courses. This naturally helps other people when they are browsing for usable material within the repository. Each review is conducted by at least two higher education faculty members. Reviews are divided in three dimensions: quality of content, potential effectiveness, and ease of use.

Figure 6. Merlot project (http://www.merlot.org).

When submitting material the author submits a web form describing the material. Required information is the title of the material, the content URL, type of material, and at least one subject category. Subject categories are based on the Library of Congress classification system used by most higher education libraries in the U.S. Optional information includes e.g., material description, target audience, author information, and technical information. Merlot projects also uses the LOM metadata descriptions.

Merlot repository holds only metadata describing the materials; the actual teaching materials are stored elsewhere on the web. Therefore, the teaching resources described within Merlot project have been implemented with a wide variety of technologies. Merlot does not define the format of the teaching materials, neither does it restrict the pedagogical usage of the resources.

Compared to normal search engines, Merlot makes it possible to search only for teaching materials that have been reviewed by other users, not just by the authors. Currently (September 2003) there are more than 9000 materials available through Merlot, more than 1000 peer reviews, and more than 15000 people have joined the community.

From the reusability point of view the resources that can be found through Merlot are not very reusable in the sense that the materials are spread all over the web and the licensing regulations or costs involved with use have to be checked individually, along with the quality and “trustworthiness” of materials. A teacher can add a link to these resources, but it’s not (usually) allowed to modify and store them on some other web server.

7.1.2 MIT OpenCourseWare

MIT (Massachusetts Institute of Technology) OCW (OpenCourseWare) project (http://ocw.mit.edu/) is a large-scale publishing project targeted for educators, students, and self-learners around the world. OCW is a publication of MIT course materials and it is freely available to anyone. Access to OCW does not require any registration. Currently, material from 500 MIT graduate and undergraduate courses is available on the OCW.

Figure 7. MIT OpenCourseWare pages (http://ocw.mit.edu/).

Materials on the OCW pages are grouped by MIT academic departments and the course subjects vary from anthropology to nuclear science. The ‘Advanced Search’ option makes it possible to search for words or phrases found in metadata description fields (e.g., keywords) or the course sections (e.g., exams or study materials). Along with the MIT Libraries the MIT OCW has developed a set of metadata to enable accurate searches across course content materials. Each course is structured into sections (e.g., syllabus, readings, assignments, lecture videos). Materials are mostly available in html- or pdf-format.

Unlike in the Merlot repository approach, the materials found on OCW are freely available for non-commercial use with no charge. On the OCW pages the use of materials is described as: “course materials offered on the MIT OCW Web site may be used, copied, distributed, translated, and modified, but only for non-commercial educational purposes that are made freely available to other users under the same terms defined by the MIT OCW legal notice.” More precisely, materials on OCW are distributed under the Creative Commons licence terms (see http://creativecommons.org/licenses/by-nc-sa/1.0/).

As Wiley (2002) puts it, “MIT obviously recognises that the institution’s primary value is not in it’s content as much as it is in the social interactions that it facilitates.” In my opinion this approach on free access to information and emphasis on social aspects of learning shows that the assumptions of socio-cultural theory on learning (see chapter 1) have been accepted even at the organisational level of MIT. In this sense the course materials is just a subset of the MIT education and web is not a substitute for the whole academic education experience.

From the learning object point of view the materials on the pages are not divided into actual learning objects that are individually described by metadata and reusable in different contexts. Rather, in OCW every course itself is a learning entity, described by metadata, and divided into subsections (e.g., lectures). When thinking about modifying the materials for the purpose of building a new course, the materials have to be copied and pasted as any traditional source of information. However, the quality and quantity of the materials is substantial, and the possibility to freely modify and redistribute the materials should make the OCW a very interesting source of information for many people working at higher educational sector. At this point this seems to be by far the largest source of approved higher education teaching materials. Since all resources are actually used in MIT education and created exclusively by members of the MIT Faculty, the ‘trustworthiness’ of the teaching materials on OCW should be widely approved.

7.2 Repository technologies

When learning material is assembled into a repository, a suitable technological solution for maintaining repository content and necessary functionality has to be chosen and implemented. One specification for implementing a repository is provided by IMS.

7.2.1 Digital repositories interoperability – IMS specification

The IMS Global Learning Consortium has released a specification concerning the creation of learning object repositories, the IMS interoperability specification (http://www.imsglobal.org/digitalrepositories). The purpose of the specification is to provide recommendations for the interoperation of the most common repository functions. IMS specification defines digital repositories as being “any collection of resources that are accessible via a network without prior knowledge of the structure of the collection”. These repositories may hold the actual learning objects or only metadata that describe learning objects. Learning objects within such repositories are designed to meet other IMS specifications (see section 5.4.3).

IMS Core Functions Information Model (IMS, 2003) defines, e.g., functional architecture design, along with use cases, for most common repository functions. User roles are defined as creator, learner, infoseeker, and agent. Repository functions include, e.g., a learner or an agent searching for material, a creator submitting and modifying existing material, and an agent searching for metadata from multiple repositories and populating its own repository with the new metadata. The IMS interoperability recommendation is not a complete formal technical specification on how to create a learning object repository; rather it tries to define some issues that should be considered when thinking about using learning objects in multiple repositories.

7.2.2 Norwegian education portal

The Norwegian ministry of education and research has established an education portal (http://www.utdanning.no) as part of the Norwegian learning net for the education sector. One aim of the project is better reuse and sharing of learning objects. The portal provides functionalities for searching resources through a common interface using a web browser. For this purpose the portal has to index the resources from different resource providers, and a common vocabulary and categorisation of resources had to be established. (Skår et al., 2003)

Skår et al. (2003) mention some of the chosen technologies and reasons for choosing them:

- XML over HTTP is used as the transportation protocol. This solution ensures interoperability across content providers, and XML at the syntactic level is likely to stay technically supported in the future.

- The LOM standard (see section 5.4.1) was chosen to ensure consistency at the data/domain level. From the LOM standard only 21 elements of the total 90 elements were used to establish a minimalist but still flexible framework for describing resources. Emphasis was on elements for classification of learning resources, since they are used to categorise learning resources into a navigational structure.

- The portal and search interface was developed with IBM Websphere Portal Server (http://www.ibm.com/websphere) and Verity K2 (http://www.verity.com/) as the search engine.

Similar national education portal initiatives that have exploited the LOM standard have previously been established in, e.g., Canada (http://www.cancore.ca/) and UK (http://www.curriculumonline.gov.uk).

7.3 Digital libraries supporting learning content production

One of the main obstacles in creating teaching material is to find relevant and trustworthy source for information on the subject in question. Traditionally the best way has been a visit to a library where scientific publications, journals, and books can be borrowed and used as information resources. Along with the traditional libraries several organisations and consortiums have developed their own digitally deliverable information sources, digital libraries. Quite commonly academic libraries have subscribed to digitally deliverable scientific journals instead of having all essential publications available off-the-shelf.

Scientific articles are also located outside the actual web journals, for example scientists frequently post preprints of their articles on their own personal web sites. The rapid increase in the volume of scientific literature has led to researchers constantly fighting information overload when searching for information. Staying up to date with recently published literature - and actually finding relevant sources - is becoming increasingly difficult, if not impossible. (Lawrence et al., 1999)

7.3.1 CiteSeer

CiteSeer, also known as ResearchIndex (http://citeseer.org/), is a scientific literature digital library that aims to improve the dissemination and feedback of scientific literature available on the web. Typical web search engines have difficulty with keeping up to date and they currently do not index the contents of Postscript and PDF (portable document format) files. One approach to find relevant scientific papers is to use a citation index that catalogues the citations that an article makes, linking the article with the cited works. Many citation indexes of scientific literature, however, depend heavily on human preparation or editing of the citation information. To make the extra editing requirements unnecessary, CiteSeer uses ACI (Autonomous Citation Index) to autonomously create a citation index that can be used for literature search and evaluation. CiteSeer works by downloading papers from the web and converting them to text. It then parses the converted papers to extract the citations and the context in which the citations are made in the body of the paper and stores the parsed information in a database. (Lawrence et al., 1999)

CiteSeer indexes Postscript and PDF research articles on the web and provides many features to allow efficient citation indexing and searching functionalities, e.g., locating related and similar documents, autonomous use of search engines and crawling to locate scientific papers on the web. Citation indexes can also be used to find related articles that may not be found using keywords.

A quick search on CiteSeer with a phrase “learning object” gives 125 hits to scientific papers considering learning objects, although some hits refer to machine learning research. Compared to searching scientific publications via standard search engines (e.g., Google), CiteSeer gives much more accurate results, along with listings of related publications and citations.

7.3.2 Public Library of Science - PLoS

Traditionally access to digital scientific libraries and journals requires paying subscription fees that can sometimes be very high. Commonly university libraries and different organisations have paid these fees, so that their students and employees are able to access the journals. Individuals outside these organisations cannot easily gain access to those journals, and likewise the subscription fees can be too high for several organisations that could benefit from the information. However, recently some widely publicized attempts have been made that may change the concept on access to scientific information considerably.

Public Library of Science (http://www.plos.org) is a non-profit organisation of scientists and physicians committed to making the world's scientific and medical literature a freely available public resource. PLoS focuses on publishing biological and medical research literature. PLoS Biology launched its first issue on October 2003, both in print and online, and the PloS Medicine journal will follow in 2004. Access to all journals is free of charge to anyone. It is assumed that unrestricted access to scientific ideas, methods, results, and conclusions will speed the progress of science and medicine, and will more directly bring the benefits of research to the public (PloS, 2003).

All articles published in the PloS undergo a strict peer review and editorial as in any other scientific publications. However, since publishing, reviewing, and editing articles costs money, in the PloS publishing is seen as an integral part of the research process, and an subscription fee has to be paid for all published articles to cover these expenses. Presently the publication fee is $1500.

As is the case with the MIT OCW learning resources, all articles within the PloS are scientifically reliable, making it a valuable source for learning content creation. However, currently the PloS covers only biological issues, and medical articles will be published in the year 2004.

7.4 Software Agents

Software agents can be used to support learning, or especially content creation, in several ways, e.g., by notifying the learner of relevant learning content, or possibly even by allowing computer agents to "automatically and dynamically compose personalized lessons for an individual learner" (IEEE, 2003). Agents could also support the learning content creation process by notifying an individual when, for example, some research related content appears in a digital library, or when a learning object with certain descriptions is submitted into a learning object repository.

Agent-oriented computing can be understood as a paradigm for software engineering, as the object-oriented computing paradigm. Agent-oriented approaches can make the modelling, designing and building complex distributed software systems easier than other software engineering approaches (Jennings, 1999).

Jennings (1999) has defined software agents as entities that have the following characteristics:

- Clearly identifiable problem solving entities with well-defined boundaries and interfaces.

- Situated (embedded) in a particular environment - they receive inputs related to the state of their environment through sensors and they act on the environment through effectors.
- Designed to fulfil a specific purpose - they have particular objectives (goals) to achieve.
- Autonomous - they have control both over their internal state and over their own behaviour.
- Capable of exhibiting flexible problem solving behaviour in pursuit of their design objectives - they need to be both reactive (able to respond in a timely fashion to changes that occur in their environment) and proactive (able to opportunistically adopt new goals).

The main difference between agents and objects in object-oriented design is that agents are able to independently and autonomously adjust their behaviour, whereas objects respond to a certain impulse. Moreover, when modelling large systems it is natural to modularise the software components in terms of the objectives they achieve, instead of, e.g., data or objects they possess, as is often the custom practise in object-oriented design (Jennings, 1999).

When designing and modelling complicated software systems, some techniques are needed for tackling complexity in software. Jennings (1999) mentions three methods for tackling software complexity and discusses the suitability of the agent-oriented paradigm for those techniques. These techniques are called decomposition, abstraction, and organisation.

- The most basic technique for tackling any large problem is to divide it into smaller, more manageable chunks that can be dealt with in relative isolation, i.e., decompose the problem. In complex systems, these chunks that encapsulate their own control have to deal with other chunks to fulfil their objectives. However, in complex and distributed systems it is impossible to predict or analyse all the possibilities for interactions between components at design-time. This leads to conclusion that components need the ability to initiate and respond to interactions in a flexible manner. Such components are more precisely software agents.

- The process of defining a simplified model of the system that emphasises some of the details or properties, while suppressing others, is called problem abstraction. In Jennings' opinion (1999) the most powerful abstractions in software design are those that "minimise the semantic gap between the units of analysis that are intuitively used to conceptualise the problem and the constructs present in the solution paradigm". In complex systems these abstractions naturally correspond to agent metaphors.

- The process of identifying and managing interrelationships between various problem-solving components is called organisation. In agent systems explicit representations are made of organisational relationships and structures. In the agent-oriented design the notion of a primitive component can vary according to the needs of the observer, meaning that, e.g., entire sub-systems can be viewed as a single entity, or alternatively teams or collections of agents can be viewed as primitive components. Furthermore, such structures provide a variety of stable intermediate forms that are, in Jennings' opinion, essential for rapid development of complex systems. Basically this means that individual agents or organisational groupings can be developed in relative isolation and then added into the system in an incremental manner.

Agent-oriented approach offers interesting possibilities for building distributed and complex software systems, but it may not be the best choice to be used for a more conventional task where 'intelligent' and autonomous entities are not needed. Furthermore, software agents that support the learning process or the creation of learning materials in an advanced way are yet to be seen.

8 Case study for creating online course content

At the present moment learning objects are mainly a theoretical concept; practical tools for creating, storing, and using them are still to be developed. However, there are already some applications and methods that support the creation of reusable learning content that is described by metadata. In the next chapter one application meant for creating learning content packages from existing material is tested. For the purposes of structured content creation process one methodology, the topic-case approach, is considered and reviewed.

8.1 Technologies for creating learning objects

To test an application meant for creating learning content that is described by some approved learning object metadata standard, the Reload application was used to create an IMS content package consisting of materials taken from the MIT OCW repository (see section 7.1.2). Content packaging is the process of describing and packaging learning materials, such as an individual course or a collection of courses, into interoperable, distributable packages (Reload, 2003).

ReLoad is an open source content package and metadata editor developed at the Bolton Institute, UK. The main purpose of the ReLoad application is to wrap learning content into a single package and provide a suitable interface for metadata creation. The ReLoad supports several learning object specifications, e.g., the ADL/SCORM (ADL, 2001) and IMS metadata (IMS, 2001b) standards. The IMS Content Package specification (IMS, 2003b) defines a framework for describing and packaging learning material into interoperable, distributable packages. In fact, a learning content package, according to the IMS Content Package specification, consists of physical files (e.g., HTML and pdf documents) and an XML document called the package manifest. The manifest consists of metadata describing the content package, the hierarchy of items (views on the physical files) called the organisation, and a sub-manifest section.

The ReLoad application can also be customised to accept different metadata formats by simply renaming and customising a pre-existing metadata description file included in the application. This way ReLoad could be used for creating, e.g., language specific metadata files.

According to the IMS Learning Design Specification (IMS, 2003c), a specific ‘unit of learning’ consists of an IMS content package along with instructions on the pedagogical usage of the package, the IMS Learning Design element. IMS Learning Design is integrated with an IMS Content Package by including the ‘learning design’ element within the ‘organisations’ element. This is one way to separate pedagogy from content to achieve better reusability.

8.1.1 Creating a content package

The source material (part of an MIT software engineering course) consisted of two pdf-documents that were imported into a content package using the ReLoad application. For the metadata declaration part the IMS metadata specification was used. The physical pdf-documents were imported into the content package and their hierarchy was declared within the organisation element. Figure 8 illustrates the main window of the ReLoad application.

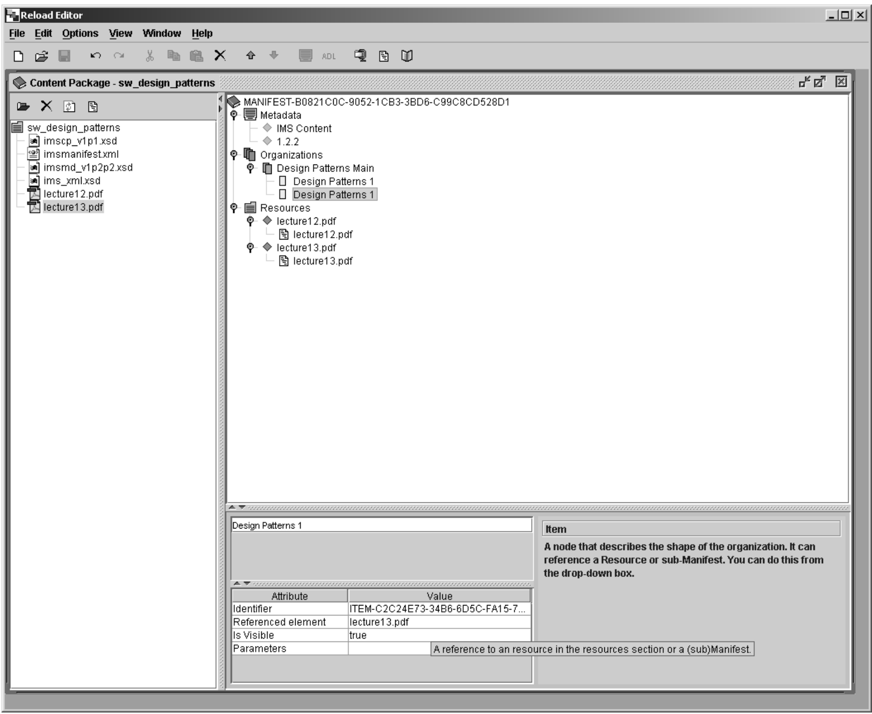

Figure 8. ReLoad application.

After necessary information of the content package is declared, a preview of the result can be viewed. ReLoad generates an HTML document of the content that can be viewed in a web browser. The result of the content package is shown in Figure 9.

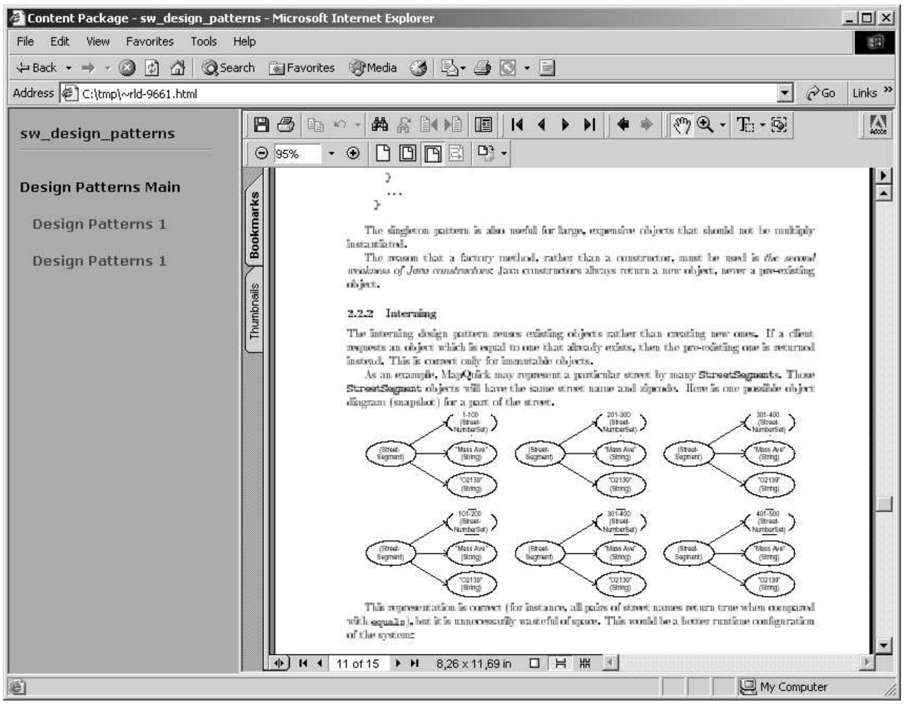

Figure 9. IMS Content Package made with Reload application.

Basically ReLoad creates a frame structure according to the hierarchy that is defined within the manifest document. Learning content management systems and VLEs that support the IMS specifications have the necessary functionalities to interpret this IMS-specific content structure in a similar manner.

8.1.2 Content package metadata

ReLoad supports the creation of metadata with a specific form (see Figure 10). The IMS specification does not define any of the metadata elements as mandatory, so basically the metadata could be just an empty metadata element tag. This is, of course, not the purpose of the metadata section. To achieve interoperability some common practise must be established within an organisation creating content packages, concerning, e.g., mandatory metadata elements and some common vocabulary to be used within the metadata elements describing the content.

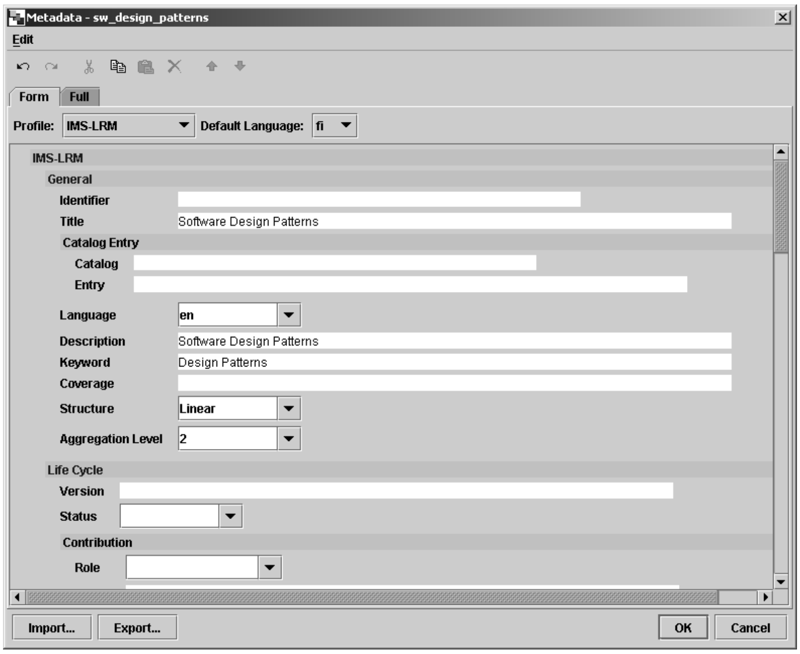

Figure 10. Reload Metadata Form View.

After the metadata elements are declared the whole content package is ready to be saved, zipped, and imported to a learning object (or rather a content package) repository.

8.1.3 Incorporating learning objects to a virtual learning environment

After creating the IMS-specific content package with the ReLoad editor, the next step was to import the content package to a virtual learning environment. University of Jyväskylä uses Optima (http://www.discendum.com), which is a VLE developed in Finland by Discendum Oy. However, Optima does not support importing IMS Content Packages into the system, so in this case the pdf-files have to be organised again within Optima. The manifest file, containing all metadata on the content package, is not much of use within Optima since Optima only opens the file in a browser but does not have support for using the metadata to support course creation. For the whole IMS Content Packaging to be a useful technology, VLE producers need to add necessary functionality into their applications. However, the ReLoad editor and the necessary metadata implementation seems to be a step into the direction of creating distributable and well defined learning content.

8.2 The topic-case approach for creating course content

The topic-case approach is an approach for capturing the necessary steps for creating web courses using a well-structured content-based development method. It aims to improve existing methods for web course content design by applying software engineering metaphors to make the content implementation process more iterative and incrementive. The whole process of creating learning material should also be learning-centric, instead of organisation-centric (Hiltunen, Kärkkäinen, 2003). A somewhat similar structured process for content management using the Rational Unified Process is introduced in (McIntosh, 2000).

Eventually an application with suitable functionalities to support the whole topic-case implementation process should be developed. In the mean time, available (open source if applicable) applications can be used to implement web course material using the method. The topic-case approach concentrates mostly on defining relations among subject topics and supporting the construction of web course material based on these topics, rather than defining the metadata issues and the interoperability and reusability of learning objects discussed in this paper. The approach aims at actual implementation of a web course structured by chunks of learning content that are defined by topic-cases. The topic-cases represent descriptions (i.e. metadata) and 'user guide' of the core content areas (topics) of the course subjects.

8.3 Topic-case development process phases

In general, the topic-case development process contains five phases: background study, content design, pedagogical design, technical design, and realisation and assessment. All phases also allow incremental and iterative development of the web course, so the process does not necessarily have to be done in a linear (e.g., waterfall) fashion (see, e.g., Royce, 1987). Moreover, each phase should be supported by a suitable software application that does not however restrict users' actions considerably, i.e., it does not force the user into an explicitely designed way of creating course content.

8.3.1 Phase 1: Background study

The first phase of the topic-case process is the background study. Basically this means answering the general questions (e.g., why to develop a web course, how to use web or VLE, time and resources) and, possibly, using the concept mapping technique for creating a general view on contents of the course. Concept mapping is a technique for 'brainstorming', realising concepts and their relations by using a visual notation. The user chooses a topic of interest and recurses down into different aspect on the subject. The difference to traditional techniques (e.g., linear writing) is not in the content but in the presentation of the context. After the background study phase the baselines of the course along with timetable and resource allocations should be realised (Hiltunen, Kärkkäinen, 2003).

To apply the topic-case approach to actual teaching material, content from an existing course on Virtual Learning Environments at University of Jyväskylä (lectured in spring 2003) was used. Therefore, the actual background study phase was reduced to graphically mapping the core concepts of the course. For concept mapping purposes I used FreeMind (http://freemind.sourceforge.net/), which is a free mind-mapping tool, licensed under GNU Public License (see http://www.gnu.org/copyleft/gpl.html). The user interface of Freemind (see Figure 11) is very simple, it only takes a few minutes to learn the basic functionalities of the application.

Freemind stores mind-maps as XML documents, so it is relatively easy to use the results in other applications. An extraction of the XML-syntax of the mind-map illustrated in Figure 11 is given in Example 8.

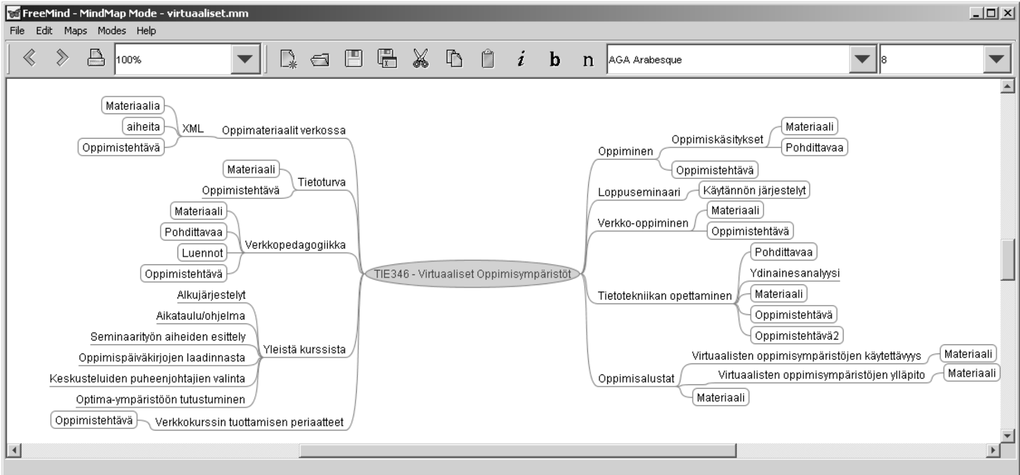

Figure 11. FreeMind mind-mapping tool.

```
<map>

<node TEXT="TIE346 - Virtuaaliset Oppimisympäristöt">

<node TEXT="Oppiminen">

<node TEXT="Oppimiskäsitykset">

<node TEXT="Materiaali" FOLDED="true">

<node TEXT="von Wright, Johan: Oppimiskäsitysten historiaa ja pedagogisia
seurauksia, luvut 3 ja 5, Opetushallitus, 1992. "/>

</node>

<node TEXT="Pohdittavaa" FOLDED="true">

<node TEXT="Palauta mieleesi jokin hyvä oppimiskokemus - mieluiten
yliopisto-opiskelujen ajalta.  "/>

</node>

…

</node>

</node>

</node>

</map>
```

Example 8. Freemind XML mapping.

For example, the child elements of the <map> node in Example 8 could be extracted from the document using XSL Transformations tools (see section 4.6) to form the skeletons for individual topics that are refined later on.

The FreeMind tool does not quite fit into the concept of the topic-case approach, since nodes from different branches cannot be linked together. In fact that is not the purpose of FreeMind; FreeMind is meant for supporting cognitive information processing by producing tree-structural data. However, presently FreeMind is one of the most sophisticated freely available open source software applications that provide graphical user interface for mind mapping.

8.3.2 Phase 2: Content design

During the content design phase the basic content of the web course is designed and documented. This phase is divided into two activities: describing the topics of a course on a general level and finding relations between individual topics. Individual topics in the topic-case approach are documented using suitable forms that capture necessary attributes (i.e. title, description, learning objective) defining the content and objectives of the topic. Documenting a topic begins from initial descriptions defining the core concepts that were realised during the 'background study'–phase. The initial descriptions include, e.g.,:

- Name of the course
- Date and author
- Topic-case name
- Summary of the topic case
- Materials engaged with the topic
- Learning objectives
- Relations to other topic-cases

The next step is to define relations between different topics. At this point topics that are very similar are merged into a single topic-case. Relations between different topic-cases define the basic contentual hierarchy of the web course. The stereotypes that define these relations indicate which topic-cases are pre-requisites for other topics and should therefore be implemented first. In the topic-case approach different topics and their relations are linked together by using qualifiers <<advances>> and <<requires>>. The qualifier <<requires>> indicates what knowledge is required before a certain topic-case can be accomplished properly, <<advances>> indicates the knowledge that would be useful to be acquired, but is not compulsory for the following topic-case. Basically the qualifiers function as guidelines on how the topic-cases can be sequenced later on. For example, the concept ‘teaching’ requires concept ‘pedagogics’, and concept ‘pedagogics in VLE’ advances the concept ‘pedagogics’.

8.3.3 Phase 3: Pedagogical design

Pedagogical design is often disregarded in web course design, though teaching in a VLE should be based on principles of human learning (Hiltunen & Kärkkäinen, 2003). The topic-case approach tries to ensure that a suitable pedagogical approach is used throughout the web course by defining the pedagogical design as an individual design phase. In this phase advisable teaching and learning activities, along with suitable assignments, are declared.

The pedagogical design phase extends the topic-case form created in the previous phase with the following attributes:

- Actor(s): involved in the topic-case (teachers and students)
- Description of activity in topic-case
- Pedagogical solutions in topic-case (advisable teaching and learning actions and assessments)

Each topic-case can include more than one pedagogical solution if necessary. In this way, e.g., different learners, learning situations, and learning environments can be supported.

8.3.4 Phase 4: Technical design

In the technical design phase the decisions concerning technical issues (e.g., software platform, media, user interface) are made. However, the implementation of a web course strongly depends on availability of resources and tools, so no detailed technical decisions are defined in the topic-case approach. Furthermore, the VLE platform in use affects the technical design phase considerably. Most VLE platforms enable importing material, but usually either in the platform's own format, or in HTML. However, if reusable chunks of learning content (learning objects) become a popular way for creating learning material, VLE producers are likely to add support for learning objects (e.g., IMS Content Packages, see section 8.1) into their software products. The topic-case approach does not restrict the choice of possible learning object standards (e.g., LOM, IMS) that could be used for describing the topic cases. Naturally, if an application using the topic-case approach is to be implemented, some existing standards and technologies should be evaluated and exploited.

8.3.5 Phase 5: Realisation and assessment

The final implementation of the web course consists of completing the individual topic-cases using the chosen pedagogical and technical solutions. The assessment means reviewing and evaluating the whole process in every phase, as well as reviewing finished topics and contents and their suitability for user, i.e. the technical, pedagogical, and contentual skills required by the user.

The technical assessment is based on five usability arguments that can be used to evaluate the technical functionality of the topic and the whole course consisting of topics. The assessment questions are, e.g., the easiness to learn to use the web course and pleasantness and efficiency of using the course. The pedagogical assessment is meant to ensure that all topic-phases have a meaningful pedagogical solution that is consistent with the underlying pedagogical models for the web course. The contentual assessment means checking that all selected topic-cases have been implemented properly and relations between topics have been defined correctly.

8.4 Analysis

The topic-case approach should eventually be supported by an application that allows the user to construct individual topics and define their relations using specific forms designed for this purpose. Constructed topic-cases and their metadata should be automatically saved to a web server, where other users could browse for existing topics and create their own topic-cases. Basically such an application is a flexible interface to a learning object repository where learning content is maintained. The topic-case forms that are used as the basis for the construction of a web-course should also be automatically prefilled to allow the user to bypass repetitive phases in the process.

In the topic-case approach, individual topic-cases can be realised as individual learning objects described in this thesis. As the basis for the topic-case metadata description fields, some existing learning object metadata standard is a natural choice, since by using existing standards (see section 5.4) it is possible to benefit from experiences gained in other repository initiatives, and more importantly, gain interoperability with other repositories (see, e.g., Skår et al., 2002). Existing learning object metadata standards, e.g., the IMS LOM specification (see section 5.4.3), cover all the topic-case description fields mentioned in the topic-case approach. For the pedagogical design phase, for example the IMS Learning Design Specification (IMS, 2003c) offers possibilities for describing educational implementation of individual learning objects.

Perhaps the most natural way for constructing a learning object repository is to use a specific web server as the central storage space for learning objects and build all functionality to the server end, allowing users to use a web browser as an interface to the repository (see, e.g., Skår et al., 2002). This significantly reduces problems that are caused by standalone software solutions, e.g., Java versions and computer performance issues. Existing freely available solutions for these purposes are, e.g., Apache Foundation's Cocoon project (http://cocoon.apache.org/), Zope (http://www.zope.org/), and Fedora (http://www.fedora.info), which is an open source digital repository management system.

9 Conclusions

Reusable learning objects as the basis for teaching material creation is still a developing concept. Especially necessary tools for creating this kind of material are still to be developed. Basically the learning content and metadata should be stored in a repository using some markup language (e.g., XML) to achieve interoperability and reusability. Generating XML documents describing the content and metadata without suitable tools is definitely too complicated and laborious, even for technical experts.

It seems that the 'e-Learning community', which has been promoting several different technological approaches meant to support learning throughout the years, has collectively adapted the concept 'learning object' as the basis of future learning content production. This is understandable, since the learning object concept seems very attractive; using the same content many times naturally helps to make the content creation process more effective than writing everything from the scratch. At the present moment there are several definitions and standards that can be used for describing learning objects and their metadata. If the LO concept is to succeed, it is likely that only a few of these definitions survive. Based on the experience from other standardisation efforts, open standards and tools, if any, are going to be widely accepted within the learning object movement in the future.

Slogans like "write once, use everywhere" have often been used in web content creation processes promoting new ways to produce information. However, creating reusable content has in many cases turned out to be very expensive. Tools for creating content using some generic markup language are often expensive and difficult to use. In software engineering the slogan has been twisted to "write once, debug everywhere", describing the difficulties that software reuse has come across with different platforms and contexts. At the extreme, learning objects are seen as pedagogically sound chunks of information that can be sequenced and reused without any significant personal effort, or even be sequenced by software agents to support learners' individual goals and preferences (see IEEE, 2002).

However, according to modern conceptions of learning, knowledge cannot be transferred from human (or computer) to a person intact. Behind the learning object metaphor is an assumption that sequenced learning objects (learning packages including technically expressed pedagogy) contain ready-made knowledge that just needs to be delivered to the learners who, after going through the pre-sequenced learning objects, automatically achieve skills and knowledge as designed by the learning object authors. This closely resembles the programmed instruction of the 60's, which did not function as planned.

If the concept 'learning object' is reduced to a piece of information along with necessary metadata that is created and stored for some purpose (meaning that a 'learning object' is stripped of any pedagogical aspects and references to other objects), it clearly resembles several other attempts to produce and reuse modular information at organisational level. For example, 'content management' and 'knowledge management' try to solve the complexity of creating, storing and retrieving information efficiently and making the 'tacit knowledge' transparent at organisational level (see, e.g., McIntosh, 2000).

As Häkkinen (2002) has stated, the main problem of instructional design has been its isolation from other fields of teaching, learning and technology. In addition to that I would say that the problem with the learning object approach, being a subset of instructional design technology, has been its isolation from other fields of content management and creation, mainly business-oriented approaches. Instructional designers and educational researchers have come up with the concept of reusability and defined a set of new concepts, but have not necessarily addressed the difficulties that reusability brings along (see, e.g., Szyperski, 1998 & Friesen, 2003). Furthermore, pedagogy as a whole is a complicated issue, and combining it with the concept of reusability makes things even more complicated. Even though pedagogical issues have been considered within the LO movement, it is not clear at this point how to combine technology and pedagogy in a meaningful way using learning objects. For example, it is yet to be seen if the IMS Learning Design Specification (IMS, 2003c), claiming to provide a "generic and flexible language designed to enable many different pedagogies to be expressed", is generic and usable enough to be used for expressing pedagogies in different contexts.

Paradoxically, efficient use of learning objects supporting content creation requires a learning object repository, a database where these objects, described by metadata, are stored and retrieved from. The creation of such a repository requires that learning objects can easily be created and stored into the repository with a suitable application. If the workload for creating a learning object exceeds significantly the workload for creating learning material as usual, it is unlikely that teachers would use such an application. Therefore the learning object should be a relatively simple piece of information so an application for creating such an entity could be easy to use. Or at least the application should not force the user to do extensive amount of work when submitting a learning object into a repository, for example to fill in dozens of mandatory metadata description fields. Otherwise the creation of these reusable learning material entities, learning objects, should be left for technical personnel and subject specialists. This means that compromises have to be made in order to achieve an established practise of creation and use of learning objects.

As a conclusion I would say that the concept of reusability should also be thought of as reshaping existing learning objects, or information objects, so that the learning object repositories would function as valuable and trustworthy information sources for teachers and learners in their individual goals. For example, the MIT OCW project introduced in this thesis is a good example of a valuable source of learning material. There is definitely need for organisational, national, and global learning material repositories. These repositories are managed by content management systems, and these to function need to exploit the technologies and common standards discussed in this thesis. However, technology should support various forms of teaching and learning, not control them.

References

ADL (2001). Sharable Content Object Reference Model Version 1.2: The SCORM Overview. Available: http://www.adlnet.org (Accessed 11.12.2003)

ASTD (2003). ASTD's e-learning magazine. Available: http://www.learningcircuits.org/ (Accessed 11.12.2003)

Bannan-Ritland, B., Dabbagh, N., Murphy, K. (2001). Learning Object Systems as Constructivist Learning Environments: Related Assumptions, Theories and Applications. Available: http://reusability.org/read/ (Accessed 11.12.2003)

Berners-Lee, T. (1997) Web Architecture: Metadata. Available: http://www.w3.org/DesignIssues/Metadata.html (Accessed 11.12.2003)

Birbeck, M., Duckett, J., Gudmunsson, O. G., Kobak, P., Lenz, E., Livingstone, S., Marcus, D., Mohr, S., Pinnock, J., Visco, K., Watt, A., Williams, K., Zaev, Z., Ozu, N. (2001) Professional XML 2^{nd} Edition. Wrox Press.

Celebrate project web pages. (2003) http://celebrate.eun.org/ (Accessed 11.12.2003)

Downes, S. (2003) Design, Standards and Reusability. Available: http://www.downes.ca/ (Accessed 11.12.2003)

Educational Object Economy Foundation (2003). Available: http://www.eoe.org (Accessed 11.12.2003)

Ertmer, P. A. & Newby, T. J. (1993). Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective. Performance Improvement Quarterly, 6 (40), 50-72.

Friesen, N. (2003). Three Objections to Learning Objects. Online Education using Learning Objects. Editor McGreal, R. London: Taylor & Francis Books Ltd. Available: http://phenom.educ.ualberta.ca/~nfriesen/ (Accessed 11.12.2003)

Gagne, R. (1985). The Conditions of Learning and Theory of Instruction. Fourth Edition, CBS College Publishing, New York.

Gibbons, A., Recker, M. M., Wiley, D. A. (2000) A reformulation of learning object granularity. Available: http://reusability.org/granularity.pdf (Accessed 11.12.2003)

Hatala, M., Richards, G. (2002) Global vs. Community Metadata Standards: Empowering Users for Knowledge Exchange. Available: http://www.edusplash.net/upload/sw-hatala-richards-final.pdf (Accessed 11.12.2003)

Hillman, D. (2003) Using Dublin Core. Available: http://dublincore.org/documents/usageguide/ (Accessed 11.12.2003)

Hiltunen L., Kärkkäinen T. (2003) Topic-case driven approach for web course design. (Preprint)

Häkkinen, P. (2002) Challenges for design of computer-based learning environments. British Journal of Educational Technology, 33(4), 465-474.

IEEE (2002): Learning Object Metadata Working Group. Available: http://ltsc.ieee.org/wg12/index.html (Accessed 11.12.2003)

IMS Learning Resource Meta-Data Best Practice and Implementation Guide (2001a). Available: http://www.imsglobal.org/metadata/imsmdv1p2p1/imsmd_bestv1p2p1.html (Accessed 11.12.2003)

IMS Learning Resource Meta-Data XML Binding (2001b). Available: http://www.imsglobal.org/metadata/imsmdv1p2p1/imsmd_bindv1p2p1.html (Accessed 11.12.2003)

IMS Learning Resource Meta-Data Information Model (2001c). Available: http://www.imsglobal.org/metadata/imsmdv1p2p1/imsmd_infov1p2p1.html (Accessed 11.12.2003)

IMS Digital Repositories Interoperability – Core Functions Information Model Version 1.0 Final Specification (2003). Available: http://www.imsproject.org/digitalrepositories/driv1p0/imsdri_infov1p0.html (Accessed 11.12.2003)

IMS Content Packaging Specification (2003b). Available: http://www.imsproject.org/content/packaging/index.cfm (Accessed 11.12.2003)

IMS Learning Design Specification (2003c). Available: http://www.imsglobal.org/learningdesign/index.cfm (Accessed 11.12.2003)

Jennings, R. (1999). Agent-Oriented Software Engineering. Available: http://www.ecs.soton.ac.uk/~nrj/download-files/cairo.pdf (Accessed 11.12.2003)

Jonassen, D., Mayes, T., & McAleese, R. (1993) A manifesto for a constructivist approach to uses of technology in higher education. Available: http://apu.gcal.ac.uk/clti/papers/TMPaper11.html (Accessed 11.12.2003)

Lawrence, S., Gile, C. L., Bollacker, K. (1999). Digital Libraries and Autonomous Citation Indexing. IEEE Computer, Volume 32, Number 6.

McIntosh, M. (2000). Content Management Using the Rational Unified Process®. Rational Software White Paper. Available: http://www.rational.com/media/products/rup/TP164.pdf (Accessed 11.12.2003)

Neven, F., Duval, E. (2002) Reusable learning objects: a survey of LOM-based repositories. Proceedings of the tenth ACM international conference on Multimedia (pages 291-294). ACM Press, New York, NY, USA. ISBN: 1-58113-620-X

Noy, N., McGuinness, D. (2001) Ontology Development 101: A Guide to Creating Your First Ontology. Available: http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf (Accessed 11.12.2003)

PloS (2003). Public Library of Science. Available: http://www.plos.org (Accessed 11.12.2003)

Tennyson, R. D. (1994) Automating Instructional Design, Development and Delivery. Berlin : Springer, corp.

Reload (2003). ReLoad editor User Guide. Available: http://www.reload.ac.uk/guide.html (Accessed 11.12.2003)

Resource Description Framework (RDF) Model and Syntax Specification, Available: http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/ (Accessed 11.12.2003)

reusability.org (2003) Available: http://reusability.org/ (Accessed 11.12.2003)

Royce, W.W. (1987) Managing the development of large software systems: concepts and techniques. International Conference on Software Engineering, Proceedings of the 9th international conference on Software Engineering, pages 328-338. IEEE Computer Society Press, Los Alamitos, CA, USA. ISBN:0-89791-216-0

Schunk, D. H. (2000) Learning theories: an educational perspective. Prentice-Hall, Inc.

Shah, U., Finin, T., Joshi, A., Cost, R. C., Mayfield, J. (2002) Information Retrieval On The Semantic Web. Conference on Information and Knowledge Management. Proceedings of the eleventh international conference on Information and knowledge management, pages 461-468. ACM Press, New York, USA. ISBN:1-58113-492-4

Sjunneson, J. (2003) Metadata for learning objects: overview, prospects and test. Masters thesis Uppsala University, Sweden. Available: http://www.skeptron.ilu.uu.se/broady/dl/dl-proj.htm (Accessed 11.12.2003)

Skår, L., Heiberg, T., Kongsli, V. Reuse learning objects through LOM and XML (2003). Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (pages 78-79). Anaheim, CA, USA. ISBN:1-58113-751-6

Szyperski, C. (1998) Component Software - Beyond Object-Oriented Programming. Addison-Wesley.

Taivalsaari, A. (1996) On the notion of inheritance. ACM Computing Surveys Volume 28, Issue 3, pages 438-479. ACM Press, New York, USA. ISSN:0360-0300

Ungar, D., Smith, R. B. (1987) Self: The power of simplicity. ACM SIGPLAN Notices, Volume 22, Issue 12, pages 227-242. ACM Press, New York, USA. ISSN: 0362-1340

Wiley, D. (2000) Learning Object Design and Sequencing Theory. Available: http://wiley.ed.usu.edu/docs/dissertation.pdf (Accessed 11.12.2003)

Wiley, D. (2002) The coming collision between automated instruction and social constructivism. Available: http://wiley.ed.usu.edu/docs/collision_09.doc (Accessed 11.12.2003)

Wiley, D. (2003) Learning Objects: Difficulties and Opportunities. Available: http://wiley.ed.usu.edu/docs/lo_do.pdf (Accessed 11.12.2003)

W3C (2000) Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation. Available: http://www.w3.org/TR/REC-xml (Accessed 11.12.2003)

W3C (2002) XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition), W3C Recommendation. Available: http://www.w3.org/TR/xhtml1/ (Accessed 11.12.2003)