VWE: a framework for modularized Virtual Learning Environments
- Applying the Learning Object Concept to the VLE

Fredrik Paulsson and Ambjörn Naeve
KMR group at CID/KTH (Royal Institute of Technology)
Lindstedtsvägen 5
100 44 Stockholm, Sweden
frepa@nada.kth.se

1. Introduction

The concept of Learning Objects has gained wide spread acceptance in the world of education. The main objective of Learning Objects is to provide a modularized model and standards that enhances flexibility, platform independence, reuse of learning content and a higher degree of control for teachers and students.

Learning Objects have been around for a number of years now and the terms, definitions and meanings of the concept has changed over time. Much of the changes are due to the fact that standards have matured, that implementation has shown that everything didn't work as expected or depending on focus and theoretical perspective. A couple of things have never changed however, Learning Objects are always about modularized content and the focus is on small chunks of fairly context-independent content that can be assembled, reused and is platform- and vendor independent. An important condition in order to realize this is the use of Learning Technology Standards such as IMS[1], IEEE/LTSC[2], SCORM[3] and others. Learning Objects are also about the freedom of teachers and students – the freedom to choose, assemble and contextualize.

The metaphor of Lego™ is often used to describe the characteristics of Learning Objects. The supporters of the Lego™ metaphor claim that anyone should be able to put together a *Learning Module* for a specific pedagogical context – simply by assembling the Learning Objects of their choice. The Lego™ metaphor is often criticized for being to simplified, which has lead to the development of more sophisticated metaphors. One that is commonly used is the metaphor of the atomic Learning Object , first addressed by Wiley in [18] and then refined in [19]. The atomic Learning Object is submitted too much stricter rules and regulations. Not anyone can assemble Learning Objects and every Learning Object cannot be assembled with any other Learning Object – they must have certain attributes and possess certain properties to fit. The atomic view (and similar) makes the e-learning life more complicated, but at the same time more realistic.

A slightly different approach to Learning Objects is taken by Song and Andersson [19]. Their definition of Learning Objects is in some respects similar to the VWE taxonomy, since they mean that Learning Objects should be regarded as decomposable, and that there must be a separation between data, operations and the carrier of the data. They also mean that an object should be described using a set of attributes and relationships to other objects. While they focus mainly on the *internal structure* of Learning Objects and their *relations to other objects*, the VWE taxonomy proposes a *general architecture model* and a taxonomy focused on the *architecture for composing* Learning Objects as well as on *the interaction* between objects.

Most of the discussion on Learning Objects is focused on modularized *content.* This view – about

[1]http://www.imsproject.org/
[2]http://ltsc.ieee.org/
[3]http://www.adlnet.org/

Learning Objects being exclusively about content - is in most cases unchallenged. There are however several problems with Learning Objects that makes it important to broaden the discussion. Many of the problems relates to how Learning Technology Standards are shaped and how the Learning Objects architecture is designed, based on existing Learning Object taxonomies. Two major problems can be identified. The first problem is a problem related to pedagogy and the use of Learning Objects: *why do we have a modularized concept for content, where the strive is to attain maximum pedagogical flexibility, when we at the same time continue to accept non-modularized, inflexible and clumsy Virtual Learning Environments (VLE) that enforce pedagogical constraints and limitations?* It is an impending risk that teachers and learners may have content that suits the pedagogical approach of their flavour, but which they are forced to fit in to a Virtual Learning Environment that doesn't? One basic assumption is that each teacher has her own favourite pedagogical methods and that she must be able to continue to use and enhance it even if she uses e-learning. This assumption is reversible and we can assume that most students have their own favourite methods for learning. Hence, the VLE must be able to support these methods, and we cannot allow for the VLE to put limitations on the pedagogical possibilities created by Learning Objects.

The second problem is of a more technological kind related to architecture and the separation of data (information), presentation (context) and logics (interactivity). Most of the Learning Objects that was studied where typically a Flash-animation, a PowerPoint or a simple Java-applet that implements an architecture where data, application logics and presentation is shamelessly mixed into an architectural mishmash. This raises a couple of questions: What is content? Where does the content end and the VLE start? Should application logics rather be a part of the VLE?

The issues raised are complex and cannot be answered in a simple and obvious way, but hopefully they will initiate an important discussion. We believe that the present concept of Learning Object is too narrow to fulfil the vision of modularization and flexibility. Maybe we must "go the whole hog"? What if we apply the same modularized concepts to the VLE?

In this paper we argue that the concept of modularization and Learning Objects must be expanded to comprise parts of the Virtual Learning Environment as well. In order to accomplish this, there is a need to modify the Learning Objects taxonomy. This is the view is that constitutes the basis for the *the Virtual Workspace Environment (VWE)* project and the VWE Learning Object taxonomy.2. Learning Objects and modularization of the VLE

The project *Virtual Workspace Environment* was initiated in 1998 in order to examine how to transfer the modularized concept from Learning Objects to the Virtual Learning Environment by tieing them closer together.

In order accomplish this different definitions and models for Learning Objects and related concepts was examined in order to derive an altered model that suits a component-based learning architecture where both the content *and* the learning environment is considered within the same model. The resulting model was tested trough the implementation of a framework for construction and use of component-based Virtual Learning Environments and learning content.

Based on existing Learning Technology Standards as well as general technology standards, a prototype for a modularized framework was developed – the VWE. The VWE framework is *service-oriented* and consists of a set of *common services* that are needed for communication and interaction between different modules ("objects").

The main objective of the VWE-project was to develop a concept and a framework for the construction of component-based (or module-based) Virtual Learning Environments that adapt to specific pedagogical contexts, includes all necessary functionality for a VLE and that supports the use of modularized content in a transparent way. A learning environment that is assembled using VWE consists of both functionality *and* content. The metaphor for such a learning environment is a *VWE Workspace*. The VWE workspace is what teachers and learners interact with. A VWE workspace can be personal, shared or both. The components that provides functionality are referred to as *VWE Tools*. VWE Tools can provide any functionality, for example the functionality to communicate and collaborate, the functionality to produce, organize, utilize and manipulate content as well as the functionality for typical Learning Management System (LMS) tasks, such as to register courses, to enlist, assess, and grade students, etc.. What is unique, however, is that both the functionality and the content are assembled in the same manner, based on the same taxonomy and within the same conceptual space.

3. The VWE Learning Object Taxonomy

The development of the VWE framework started out in the same conceptual domain as Learning Objects, using the same Learning Technology Standards, using metaphors that are similar to the Atomic Learning Object metaphor and with the same aims for flexibility, adaptability, reuse, independence of technology- and software platforms etc. A slightly modified Learning Object taxonomy, based on Wiley's taxonomy for the atomic Learning Object[1], was used in combination with a service-oriented architecture model in order to accomplish the goal. Wiley's taxonomy turned out to be suitable as a starting point since it categorises the different types according to their complexity and level of interactivity (and application logic).

The problem with the Atomic Learning Object Model is that its only foundation is Instructional Design Theory and it doesn't really consider architectural and Computer Science aspects, which makes it unsuitable for a concept like VWE in it's original shape.

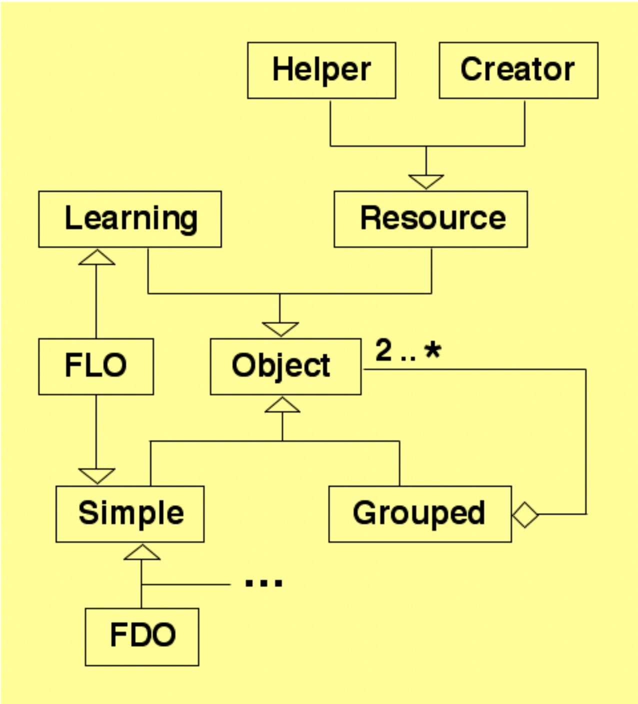


Figure 1. A concept map outlining the VWE Learning Object Taxonomy.

To enable the development of the VWE-framework there was a need to make a clearer definition of different concepts in the part of the *learning architecture,* where the Learning Object plays an important role. The analysis gave four basic categories of constituents that serves as building blocks:

Simple Learning Object

Simple Learning Objects are the smallest pieces of content that can be isolated and used as building blocks. A Simple Learning Object is an arbitrary *digital building block* that is described for use in a learning context. It is typically a picture, an animation, a text, an XML-file etc. A Simple Learning Object can be equivalent to a *Fundamental Learning Object* - described by Wiley and Nelson as the most fundamental Learning Object [20] - or it can be a *Fundamental Data Object* that is *not* a Learning Object by definition, but still relevant in a specific context.*Resource Object*
The Resource Object has been added to the VWE LO taxonomy in order to allow separation of content, application logics and presentation – as shown in figure 1. The Resource Object is the building block that adds functionality (application logics) to the VLE as well as to the content in terms of Learning Objects. There are two different types of Resource Object, which are used in slightly different ways. The first type is the Helper Resource Object, which is used as a support component for content and especially for Simple Learning Objects. Examples of such use is an explorer/viewer for chemical molecules, for example using the Chemical Markup Language (CML), or an application that interprets and renders tests, for example using the IMS QTI specification. In this way the Resource Objects can be used for making Simple Learning Objects usable in a learning context through the construction of Grouped Learning Objects (see below). The second type of Resource Object is the Creator Resource Object, which is used for adding functionality to the VLE, such as whiteboard functionality, authoring tools or tools for teacher/student planning etc. The Creator Resource Object can be used as stand-alone - which may be the case with a whiteboard,- where it can even be used to produce new Simple Learning Objects. It can also be used together with Fundamental Data Objects*,* such as student data, using the IMS Reusable Definition of Competency or Educational Objective (RDCEO) [4]. The Resource Object is also responsible for acting as a link between the Learning Objects and the rest of the Virtual Learning Environment, which means that the Resource Object must implement the required interfaces for interacting with relevant services. A Resource Object may provide both client and server functionality.

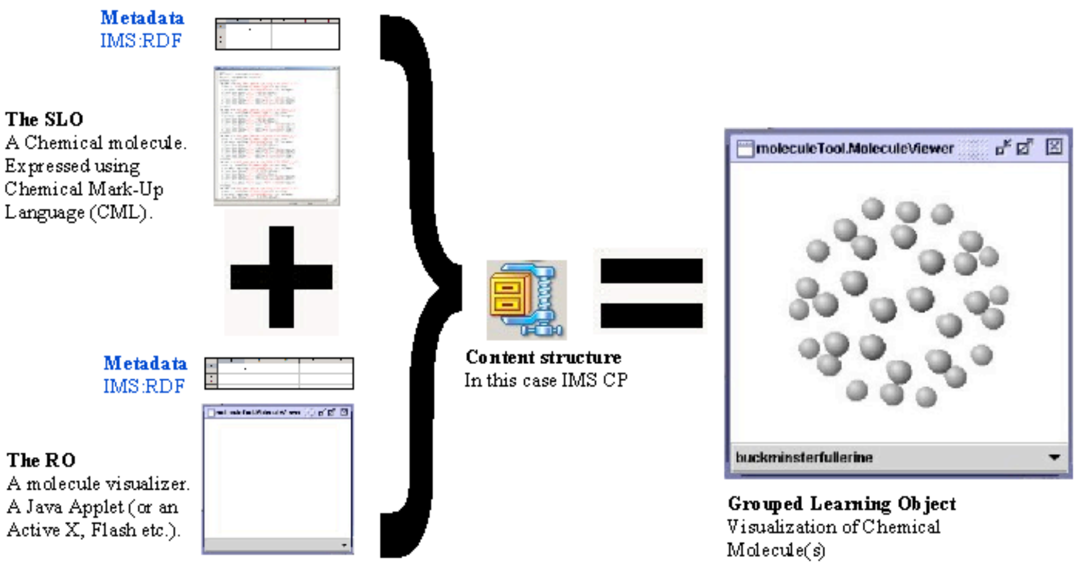
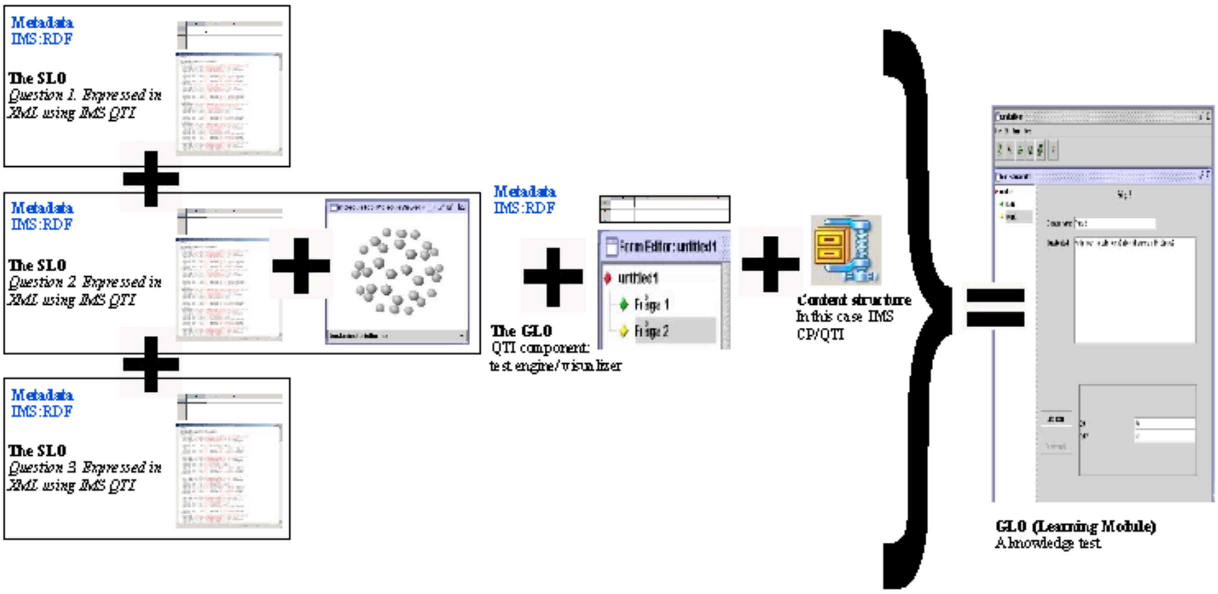


Figure 2 shows the relation between a Fundamental Learning Object – in this case a CML/XML-file and a Helper Resource Object – in this case a generic CML Viewer. These are kept together by an IMS Content Packaging stucture to form a unit that is a Grouped Learning Object.

The Grouped Learning Object
At the next level of granularity in the VWE LO taxonomy there is the *Grouped Learning Object*. A Grouped Learning Object is the result of the combination of two or more Objects, such as e.g.

Simple Learning Objects and Resource Objects. It is at the level of the Grouped Learning Object that the pedagogical context of the content starts to be shaped. The Grouped Learning Object embraces the rest of Wiley's Atomic Learning Objects taxonomy in the sense that all of the remaining types of Learning Objects (*Combined-Closed Learning Object, Combined-Open Learning Object, Generative-Presentation Learning Object and Generative-Instructional Learning Object* [19]) can be assembled from Simple Learning Objects together with Resource Objects. The Grouped Learning Object can be regarded to be at the same level of granularity as the *Shareable Content Object* (SCO) defined in SCORM [8]. In the same manner as the Shareable Content Object



, the Grouped Learning Object represents the lowest level of granularity that can be tracked by the VLE - or LMS which is the term used by SCORM [1].*Figure 3 shows the relation between several VWE Objects – in this case test questions which are Grouped Learning Objects - assembled using a Fundamental Learning Object and a Helper Resource Object. The second question uses another Grouped Learning Object to visualize chemical molecules. The result of the assembled and sequenced questions is a Learning Module – a knowledge test.*

Learning Module

The final level of granularity is the Learning Module. A Learning Module is a collection of Grouped Learning Objects that are prepared for a specific learning context. They may contain content as well as parts of the functionality that constitutes the VLE itself. This is the level that concerns students. A Learning Module is typically an isolated part of a course such as e.g., a case scenario, one of the seven steps in Problem Based Learning (PBL) [3] or anything else that a teacher or learner decides to regard as a clearly defined and isolated part of the learning experience. The Learning Module is actually a sort of Grouped Learning Object as shown in figure

The relationship between the different levels of granularity and their implementation in VWE is illustrated in figure 1 and figure 2. Figure 3 shows a concept map describing the VWE Learning Object Taxonomy.

4. The VWE architecture

There is a need for a general architectural model in order to implement Learning Objects according to the modified VWE taxonomy. The reason for this is that the new taxonomy addresses a common architecture as well, and the communication between Resource Objects and other parts of the Learning Environment (including Learning Objects based on other Resource Objects) becomes vital. The VWE learning architecture can diagrammatically be divided into three main parts: *VWE Services, VWE Kernel and VWE Tools*.

4.1 VWE Services

The *VWE Services* are needed in order to allow different components of the VWE workspace to interact with the VWE Objects. VWE has four basic services that are all implemented by most Resource Objects through a simple Web Service interface. [2]:

User Service. The User Service handles issues concerning users (e.g. learners, teachers and others), such as personal data, access and rights. The User Service is linked to a log-in Service, which may be linked to a local catalogue service.

Tool Service. The Tool Service keeps track of *VWE Tools*. Each VWE Tool is linked to a specific instance of a VWE Workspace. A VWE Tool is typically a Resource Object or a Grouped Learning Object .

Workspace Service. The Workspace Service handles common issues related to workspaces. Each user has access to one or more workspaces. The structure of a workspace is described with an IMS Content Packaging [5] structure as well as with IMS Metadata, using the IMS RDF-binding [REF: Nilsson et al, http://kmr.nada.kth.se/el/ims/metadata.html].

Message Service. The Message Service is used for communication between different components of a workspace. Communication occurs between different VWE Tools and/or VWE Objects . The communication is handled through passing SOAP messages via the Message Service, which functions as a mediator between collaborating tools.

File Service. The File Service is actually a distributed file storage, which stores resources and metadata. The File Service is transparent to the type of resources, and it is used for storing user files, learning content, VWE Tools etc. The File Service uses Semantic Web technology and is based on the SCAM[4] system.[11] [14]. This means that the VWE File Service can be directly connected to other archive systems and Brokerage Services for Learning Objects. The effect of this is that an instance of a VWE Workspace is not isolated and exclusively dependent on what is stored in its local storage. Learning Objects and Resource Objects can be seamlessly discovered and retrieved from other archives, such as archives in a P2P based *Edutella network* [7] in which VWE can be set up as a peer. All VWE Services have been implemented using Web Service technology. This choice was made in order to obtain a service interface that is as standard-based as possible and at the same time avoids the problems that might be caused by firewalls and other bottlenecks in the learning infrastructure.

4.2 VWE Kernel

The VWE Kernel is a light-weight Java application that is downloaded to the browser as VWE is initialized. It is a “middle-layer” that handles the communication between the workspace, the tools on the client (the web browser), and the server-side services.

4.3 VWE Tools

VWE Tools are the most central from the user’s perspective. VWE tools provide the functionality as well as the interactivity and presentation to the content. Most of the VWE-tools are Java-based and therefore executable in a web browser. However, it would be fully possible to use other browser-based technologies, such as ActiveX or Flash, in order to implement the VWE Service

[4]http://scam.sourceforge.net/

interfaces. The model that is facilitated by the VWE Learning Object Taxonomy enables functionality (tools) to be “installed” in a workspace at any time in the same fashion as new content can be added to a traditional LMS.

5. Conclusions and future work

Our work shows that it is possible to extend a Learning Object based model to embrace not only learning content, but the virtual learning environment as well, making it possible to construct a completely modularized learning environment that works in the same way as – and together with - Learning Objects for modularized content.

A model where the Learning Object concept is extended to include, at least parts of the virtual learning environment, provides a much higher level of flexibility and strengthens the characteristic of Learning Objects in terms of reusability, modularization and de-contextualization. The experience gained from the VWE project and the modified VWE Object Taxonomy indicates that the Learning Object concept and taxonomy cannot be based solely on instructional design theory [19], but must also consider various architectural design aspects. There is a need to unite fundamentals from computer science and pedagogical aspects such as instructional design and methodology in order to find the extricating mix for Learning Objects. One obstacle is that existing Learning Technology Standards are not enough developed for this. Several of the specifications are still immature and in some cases still untested. Specifications such as IMS Content Packaging are limiting in the way that they are only able to describe very simple package structures, but more sophisticated specifications such as *IMS Learning Design [21]* is interesting for future development. There is a need for additional standards, especially regarding architecture and interfaces for learning architecture. Future research should continue to address the problem that the Learning Object concept still is too fuzzy, which has a restraining effect. There is a need to specify how concepts like objects, components and modules interrelate to each other as well as to different standard specifications. There is also a need for more clearly defined architectural guidelines and best practice, where issues such as layering and interaction between components are addressed. Our work has resulted in some ideas in this area as well as some suggestions for a slightly altered object taxonomy that makes some of their interrelations between a bit more distinct.

The main advantage of an architecture / framework such as VWE is that it enables the same conceptual model for the *entire* learning environment. The separation of data from application logics and presentation throughout the whole learning environment makes it possible to support various types of Learning Objects and related components in order to construct Learning Objects - as well as Virtual Learning Environments - that adapt better to most learning contexts. The modular approach together with the use of standards and interoperability frameworks, such as the Schools Interoperability Framework (SIF[5]) [15], facilitates the integration and interaction with other systems. It is relatively easy to develop a Resource Object that mediates between an external system and the VWE Message Service and that can be used by Learning Objects to interact with external systems – such as library systems or systems for student administration.

The primary reason for developing VWE as a demonstrator was to get a proof-of-concept for an alternative Learning Object Taxonomy. Of course, VWE is just one of potentially many ways to do this, and there are still several unsolved problems. One of the more challenging ones is to replace the VWE interface with a suitable standard. The current VWE demonstrator uses SOAP and Web Service technology together with Java RMI [22] , which is not good enough. Web Service technology is not powerful enough and creates overhead, while RMI is too Java specific. Since the overall objective is to provide a general model, it is important that the resulting architecture should be as transparent and independent as possible.

There is an ongoing development within the learning architecture area, where similar problems are

[5]http://www.sifinfo.org

addressed. One of the most exiting projects is the work done in the O.K.I project [17] at MIT and especially O.K.I OSIDs and the way they are intended to be used [6]. This is very similar to the VWE Services, and O.K.I OSIDs will be evaluated for future use in VWE. Another important and related project is Sakai [23]

A general problem affecting VWE is the lack of interoperable, sophisticated systems for metadata markup, archiving, search and retrieval, as well as for sequencing of learning resources (Resource Objects, Learning Objects and Learning Modules in the case of VWE). This reduces the flexibility and power of the VWE Learning Objects Taxonomy as well as the VWE framework itself, by preventing the existence of powerful, distributed networks of learning resources. One way out of this could be an increased use of Semantic Web technology, which can better support distributed metadata and Semantic interoperability. This is shown by the work done by the Edutella team [9] and the Knowledge Management Research Group[6] at KTH [7] [11] [10].

References

[1] ADL. (2001), Shareable *Content Object Reference Model (SCORM) Version 1.2* Retrieved October, 2003, from http://www.adlnet.org/card-index?fuseaction=DownFile&libid=40&bc=false

[2] Berglund, M. (2002), *the Virtual Workspace Environment – Technology White paper.* Unpublished manuscript

[3] Engdelius, H. Problembaserat lärande, Lund: Studentlitteratur 1999.

[4] *IMS Reusable Definition of Competency or Educational Objective - Best Practice and Implementation Guide*. (2002). Retrieved 25 November, 2004, from http://www.imsglobal.org/competencies/rdceov1p0/imsrdceo_bestv1p0.html

[5] Consortium, I. G. L. (2004). *Content Packaging Specification*. Retrieved 2004-11-01, 2004, from http://www.imsglobal.org/content/packaging/index.html

[6] Kahn, J. (2004). Managing Complexity and Surviving Technology Change: MIT.

[7] Kraan, W. (2003, November 05, 2003). *Using SHAME to fill your SCAM*. Retrieved May 12, 2004, from http://www.cetis.ac.uk/content2/20031105152216

[8] *Making Sense of Learning Specifications & Standards: A Decision Maker's Guide to their Adoption*. (2002, 2002-03-08). Retrieved January 15, 2005, from http://www.masie.com/standards/S3_Guide.pdf

[9] Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., et al. (2002). *Edutella: A P2P Networking Infrastructure Based on RDF.* Paper presented at the 11th World Wide Web Conference (WWW2002), Hawaii, USA.

[10] Nilsson, M., Palmér, M., & Naeve, A. (2002). *Semantic Web Meta-data for e-Learning - Some Architectural Guidelines.* Paper presented at the 11th World Wide Web Conference (WWW2002), Hawaii, USA.

[11] Palmer, M., Naeve, A., & Paulsson, F. (2004, 10 May 2004). *The SCAM Framework: Helping Semantic Web Applications to Store and Access Metadata.* Paper presented at the the European Semantic Web Symposium 2004, Heraklion, Greece.

[12] Paulsson, F. (2001). *The Virtual Workspace Environment. VWE for Learning - Tomorrows learning environment today?* Paper presented at the I3 -2000, Jönköping, Sweden.

[13] Paulsson, F. (2002, November 25-27). *The Learning Object: bridging the gap between vision and reality.* Paper presented at the NetLearning 2002, Jönköping, Sweden.

[14] Paulsson, F. & Naeve. A. (2003). Standardized Content Archive Management – SCAM. *IEEE Learning Technology newsletter, 5*(1), 40-42.

[6] http://kmr.nada.kth.se

[15] Schools Interoperability Framework Implementation Specification Version 1.1. Washington, DC: Software & Information Industry Association.

[16] Song, W. W. (1999). *Metadata for the management of electronic documents in the governmental organisations and learning objects in the learning domain* (Research report No. SITI 99:03). Kista: SITI, SISU.

[17] Thorne, S., & Kahn, J. (2003). O.K.I.™ Architectural Concepts: MIT.

[18] Wiley, D. A. (1999). *The Post-LEGO Learning Object*. Retrieved 16 March, 2002, from http://wiley.ed.usu.edu/docs/post-lego.pdf

[19] Wiley, D. A. (2002). *The Instructional Use of Learning Objects.* In D. A. Wiley (Ed.), (First ed., pp. 15-19). Bloominton: Agency for Instructional Technology and Association for Educational Communications & Technology.

[20] Wiley, D. A., & Nelson, L. M. (1998, 2004-06-14). *the Fundamental Object*. Retrieved 14 January, 2005, from http://wiley.ed.usu.edu/docs/fundamental.html.

[21] *IMS Learning Design Information Model. Version 1.0 Final Specification*. (2003). IMS Global Learning Consortium Inc.

[22] Govindaraju, M., Slominski, A., Choppella, V., Bramley, R., & Gannon, D. (2000). *Requirements for and Evaluation of RMI Protocols for Scientific Computing.* Bloomington, IN: Indiana University, Department of Computer Science.

[23] *SAKAI Project*. (2004). Retrieved 20 February, 2005, from http://sakaiproject.org/