SeLeNe – Self E-Learning Networks

IST-2001-39045

WP2 Deliverable 2.1:

E-Learning Standards

Miltos Stratakis
Institute of Computer Science,
Foundation for Research and
Technology, Hellas (FORTH-ICS)
{mstratak@ics.forth.gr}

Vassilis Christophides
Institute of Computer Science,
Foundation for Research and
Technology, Hellas (FORTH-ICS)
{christop@ics.forth.gr}

Kevin Keenoy
School of Computer Science and
Information Systems, Birkbeck
College, University of London
{kevin@dcs.bbk.ac.uk}

Aimilia Magkanaraki
Institute of Computer Science,
Foundation for Research and
Technology, Hellas (FORTH-ICS)
{aimilia@ics.forth.gr}

Abstract

Learning objects are increasingly seen as key to a technology-based revolution in education and training, even to an emerging global knowledge economy. An international effort is underway to formulate standards that will enable Learning Object (LO) reuse and exchange among multiple educational settings, instructors, courses and institutions. The scope of this deliverable is to appropriately define the notion of learning content available on corporate networks or the Internet, as well as, to provide the core information modeling constructs of e-Learning applications. Hence, we survey several e-Learning standards and consider the feasibility of expressing them using Semantic Web languages, like RDF/S. We conclude our deliverable with the most important pedagogical limitations of existing standards, which are crucial for deploying a full-fledged Self e-Learning Network.

Heraklion, Crete

July 30, 2003

The SeLeNe Project

Life-long learning and the knowledge economy have brought about the need to support a broad and diverse community of learners throughout their lifetimes. These learners are geographically distributed and highly heterogeneous in their educational backgrounds and learning needs. The number of learning resources available on the Web is continuously increasing, thus indicating the Web's enormous potential as a significant resource of educational material both for learners and instructors.

The SeLeNe Project aims to elaborate new educational metaphors and tools in order to facilitate the formation of learning communities that require world-wide discovery and assimilation of knowledge. To realize this vision, SeLeNe is relying on semantic metadata describing educational material. SeLeNe offers advanced services for the sharing, and collaborative creation of learning resources, facilitating a syndicated and personalised access to such resources. These resources may be seen as the modern equivalent of textbooks, comprising rich composition structures, "how to read" prerequisite paths, subject indices, and detailed learning objectives.

The SeLeNe Project (IST-2001-39045) is a one-year Accompanying Measure funded by EU FP5, running from 1st November 2002 to 31st October 2003. The project falls into action line V.1.9 CPA9 of the IST 2002 Work Programme, and is contributing to the objectives of Information and Knowledge Grids by allowing access to widespread information and knowledge, with e-Learning as the test-bed application. The project is conducting a feasibility study of using Semantic Web technology for syndicating knowledge-intensive resources (such as learning objects) and for creating personalized views over such a Knowledge Grid.

Executive summary

This deliverable (2.1) is part of the SeLeNe Workpackage 2 having two main objectives:

- To study existing metadata schemas for describing educational material.
- To identify learners' and instructors' expectations of e-learning environments, and define the functional requirements of SeLeNe.

Self e-learning networks rely heavily on machine processable and semantically intensive metadata describing the meaning, usage, accessibility, quality and validity of available educational resources. This workpackage is studying the most commonly used types of learning objects and the corresponding description granularity that the metadata must support in order to enable their syndication and personalisation. Identifying the appropriate description granularity of educational resources requires a clear understanding of the semantics of existing educational metadata schemas, as well as their relationship to domain/task specific taxonomies dynamically constructed by SeLeNe users for their own learning purposes. Deliverable 2.1 is reporting on these issues.

Workpackage 2 is also studying learners' and instructors' expectations of Web-based e-learning environments. This has been combined with an analysis of state-of-the-art Learning Management Systems in order to ascertain the functional requirements of SeLeNe, both from the learner's and the instructor's perspective. Deliverable 2.2 is reporting on these issues.

The scope of this Deliverable 2.1 is to introduce an appropriate definition – for SeLeNe's purposes – of learning objects available on corporate networks or the Internet, as well as to provide the core information modelling constructs of e-Learning applications. We survey several standard specifications for metadata (i.e., descriptive information) associated with learning objects, and we particularly focus on the semantics of individual learning objects at different levels of granularity (e.g., modules, courses, lectures, etc.), as well as their interrelationships. We also consider the feasibility of expressing existing metadata specifications using the RDF/S Semantic Web language, and indicate the compatibility of different e-Learning standards. We conclude our deliverable with the most important limitations of existing standards to capture the semantics of learning objects from a pedagogical viewpoint, which are crucial for deploying a full-fledged Self e-Learning Network.

Revision Information

Revision date	Version	Changes
January 7, 2003	0.9	Detailed outline
February 1, 2003	1.0	First draft proposal
May 1, 2003	1.1	Second draft edited and submitted to Project Coordinator
July 30, 2003	1.2	Third, final version

Table of Contents

1. Introduction

Learning has always been an important instrument for organizations of any kind to enhance the skills of their members. By exploiting the potential provided in the electronic era, **e-Learning** constitutes a distributed, learner-oriented, personalized and non-linear/dynamic learning process [16], whose aim is to provide on-demand, task relevant educational material. Indicative of its wide acceptance and application is the continuously growing interest expressed from various user communities (e.g., educational, corporate, etc.) across the world and the fact that organizations rely more and more on e-Learning to support the learning processes they pursue. The objective of an e-Learning environment is to facilitate the accessibility and presentation of digital material - not necessarily primarily designed for educational purposes - to a wide spectrum of audiences with diverse educational backgrounds and requirements: corporate staff, learners, instructors or academics. A critical parameter of this process that should be paid attention to is *time* (or lack of it), which essentially comes down to the provision of not only appropriate learning material content (highly specified and not too general), but also to a powerful mechanism for organizing and customizing such material according to user profiles and business demands.

More specifically, e-Learning applications are based on the transmission of learning content across various computing environments and platforms; hence, what has to be specified is a structure unit suitable for this interoperation. This "learning unit" is called a **Learning Object** (LO) and its attributes as well as structure will be subsequently presented. In the context of SeLeNe, we designate as a *Learning Object* any digital object (or physical object disposing a digital surrogate), which may function as a means for learning and is essentially what instructors and learners - or other participants in the learning process - are sharing. However, the worldwide interest in implementing e-Learning services raises important interoperability issues. One feasible solution is to adopt some standard ways for the creation and transmission of e-Learning content across the Web. To answer this need, an international effort is underway to propose standards concerning LO modeling, exchange and reuse among multiple educational settings, instructors, courses and organizations. For instance, we can distinguish the European consortium ARIADNE[1], the American IMS[2], ADL[3] and AICC[4] and the international efforts IEEE LTSC[5], ISO/IEC – JTC1/SC36[6], CEN/ISSS[7], DC[8] and W3C[9], which, till now, have produced several standards with strong interrelationships. The purpose of this deliverable is to exhibit these interrelationships and present what is currently available to instructors, learners and academics to locate and (re)use digital educational material, as well as identify limitations of existing metadata standards to fully support the e-Learning vision.

---

[1] http://www.ariadne-eu.org/

[2] http://www.imsproject.org/

[3] http://www.adlnet.org/

[4] http://www.aicc.org/

[5] http://ltsc.ieee.org/

[6] http://jtc1sc36.org/

[7] http://www.cenorm.be/isss/

[8] http://dublincore.org/

[9] http://www.w3.org/

The present deliverable is organized as follows: Section 2 revisits existing definitions of LOs and proposes a working definition in the context of the SeLeNe Project. It also identifies the different granularity levels into which one LO could be decomposed. Furthermore, it refers to representational aspects of LOs, such as their main descriptive attributes and conceptual representation levels. Section 3 presents the main standardization efforts and their interrelationships, focusing on standards for LO metadata and content structure modeling. Section 4 concludes this deliverable by identifying the most important limitations of existing specifications to capture the semantics of LOs from a pedagogical viewpoint, which are crucial for deploying a fully-fledged Self e-Learning Network.

2. Learning Objects

Although the appeal of Learning Objects to a wide audience is inarguable, there is a conceptual confusion primarily stemming from the definition of what essentially constitutes a "*Learning Object*" (LO). This pervasive confusion exists mainly between LOs as content for learning, as pointers to learning resources or as metadata about such resources, as well as whether this metadata is part of the LO or resides in a different database [4]. Apparently, this confusion has led to various proposals for defining a LO, each of which takes a disparate approach for perceiving LOs with respect to their nature (digital or not, web accessible or not, etc.), content, structure, scope, pursued learning objectives, etc. Some of these definitions refer only to digital objects, but almost all proposals include every digital or physical object with learning content.

One of the simplest views of LOs is as re-usable components of courses. These components can vary in size (e.g., a diagram, a question, an exam paper, a lecture), but they are generally thought of as being smaller than an entire course. Few people would argue with this definition, but it is general enough to be of use. Thus, more formal definitions have been proposed. For instance, according to the Learning Technology Standards Committee[10] (LTSC) of the IEEE (Institute of Electrical and Electronics Engineers):

A Learning Object can be any entity, digital or not, that can be used or referenced in technology-supported learning.

The Learning Objects Network Inc.[11] (LON) uses an equally broad definition:

Learning objects are small stand-alone "chunks" of information designed to be easily reused and repackaged to meet the needs of different audiences. They typically are designed to achieve a certain narrow learning objective and may contain an assessment to determine success against that objective. Learning objects may reflect varying degrees of granularity ranging from as large as a chapter in a book, a case study, or an interactive courseware topic, to smaller items such as a single pedagogical concept (teaching the boiling point of water, for example).

This ubiquity of LOs is not a bad thing in itself. It is true that anything that has existed can be put to some educational use, so maybe we should allow the definition of LOs to be as broad as this. Furthermore, having as broad as possible a definition allows for the widest use of the respective standard, therefore leaving to the organizations the decision of what will essentially constitute a LO in their educational context.

In general, we can classify LOs into two main categories:
1) **Physical LOs**, which are non-digital entities, like a simple text or a workbook. These objects must have a "digital surrogate" if they are to be referenced on the Web, e.g., a book that is not available online as a whole could have a URL reference in a Web page, which is the digital surrogate of the book.
2) **Online LOs**, which are directly related to a web environment. For example, an online .gif image, a Web page or a Java applet recommending a Web browser could be considered online LOs.

---

[10] http://ltsc.ieee.org

[11] http://www.learningobjectsnetwork.com/Concepts.htm

In a Self e-Learning Network, the LOs described by metadata are those available on the Web (online LOs), so it is proposed that we use the term "*Learning Object*" to mean "*Learning Object available on the Web*". Although regarding LOs solely as digital, on-line resources restricts the spectrum of what can be characterized as a LO, one of the key advantages of web-based resources as educational tools is that they can be used simultaneously by many users, unlike traditional resources, such as textbooks and worksheets, where each learner needs a separate copy. Therefore, a working definition (which borrows from Simon and Quemada 's definition of "educational material" [15]) for use in SeLeNe could be something like:

Learning objects are electronic, sharable chunks of reusable learning content, available on the Web.

This definition includes both atomic and complex web-based LOs at all levels of granularity –as identified in the next section, but excludes physical objects, such as textbooks and CD-ROM's, which are not sharable and cannot be stored in the kind of distributed learning-object repositories envisaged in the SeLeNe project. Furthermore, it focuses on LO *reusability*; an approach underlying a vision of computer-based systems for locating and retrieving digital materials to (re)use in instructional situations. It is important to recognize that reusability implies more than one-time access, so LO technology implicitly entails some form of knowledge or document management, and therefore some kind of consistent indexing scheme [4], as well as explicit encoding scheme coping, for instance, with learning level, style and environment, reading level, learning objectives and activities, etc. The following sections report on how each of the most generally accepted educational metadata standards deals with each of these issues.

2.1   Granularity of Learning Objects

We can observe that in real life there is a wide spectrum of learning content with respect to its size and scope. Learning content could range from a single slide to a PhD certificate and can be used for different goals, e.g., as a lesson, a course or a simple notation. These facts impose a flexible representation of a LO with respect to the size and scope of its learning content. This has led to the distinction of different *granularity levels* of a LO. The general granularity level hierarchy of LOs is shown in Figure 1, where we present an example of the second-year course CS252 (Object-Oriented Programming) of the Computer Science Department (CSD) at the University of Crete.

The first and simplest level (lowest level) is the ***information object*** or ***component*** and represents every object that can be used in the learning process, like a photograph, a 3D image, a simple text or a video clip. For example, a single component can be a video clip from a talk. This video clip becomes a LO when a lesson is added to it. As we can see in Figure 1, many different LOs can be created from one single component. For instance, from the video clip above, one could create lessons in history, media studies and many other subjects. In the CS252 example, a simple .ppt slide is a component. We can observe that LOs of this granularity level cannot contain other LOs. For this reason, they are also characterized as *atomic* LOs. Respectively, the LOs belonging to the remaining granularity levels can contain other LOs and are considered to be *composite* LOs.
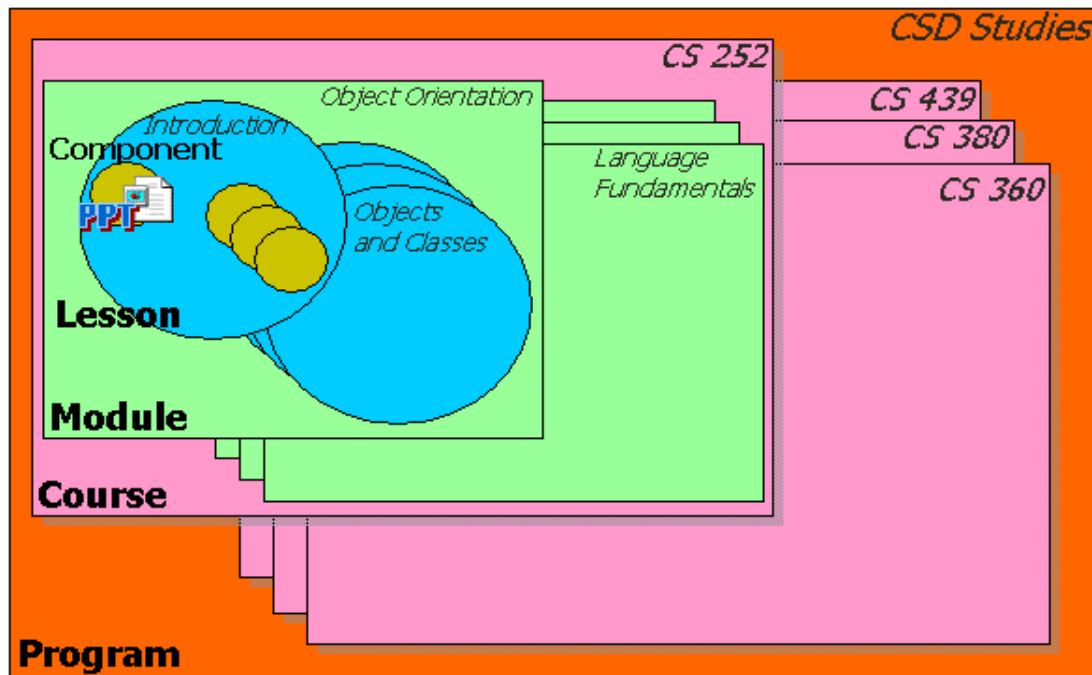
Figure 1: Learning Object Granularity Levels (CS252 Example)

More specifically, the second granularity level is that of the ***Lesson***. A lesson is a grouping of components with a specific theme and its duration is typically less than 90 minutes. In the CS252 example, the slides of the introduction in object-oriented programming constitute the components of the lesson "Introduction" of the CS252 course. The third granularity level is called ***Module***, with a module constituting a LO of less than 10 hours of learning duration. Modules are longer learning experiences or groupings of lessons. In the CS252 example, the grouping of the lessons "Introduction", "Objects and Classes" and other related lessons with introductory information in OOP comprises the module "Object Orientation" of the CS252 course. If lessons are longer than 10 hours or if they consist of more than one module, they are considered to be a ***Course*** (fourth level). That is, a course is a grouping of a large number of lessons or a grouping of modules. In our example, it is obvious that the group of all lessons or the group of all modules of CS252 represents the course itself. Lastly, a ***Program*** represents the fifth and higher granularity level. A program is a group of courses that leads towards a certificate or a diploma. In the example of Figure 1, we can observe that the grouping of all courses at the Computer Science Department constitutes the "CSD Studies" program leading to the CSD Graduate Diploma.

2.2   Conceptual Representation of a Learning Object

LOs are characterized by distinguished relationships among them or other entities applied to them. An approach to the conceptual representation of LOs is shown in Figure 2, which depicts a simplified model for describing LOs using attributes with information about their content as well as their pedagogical value.
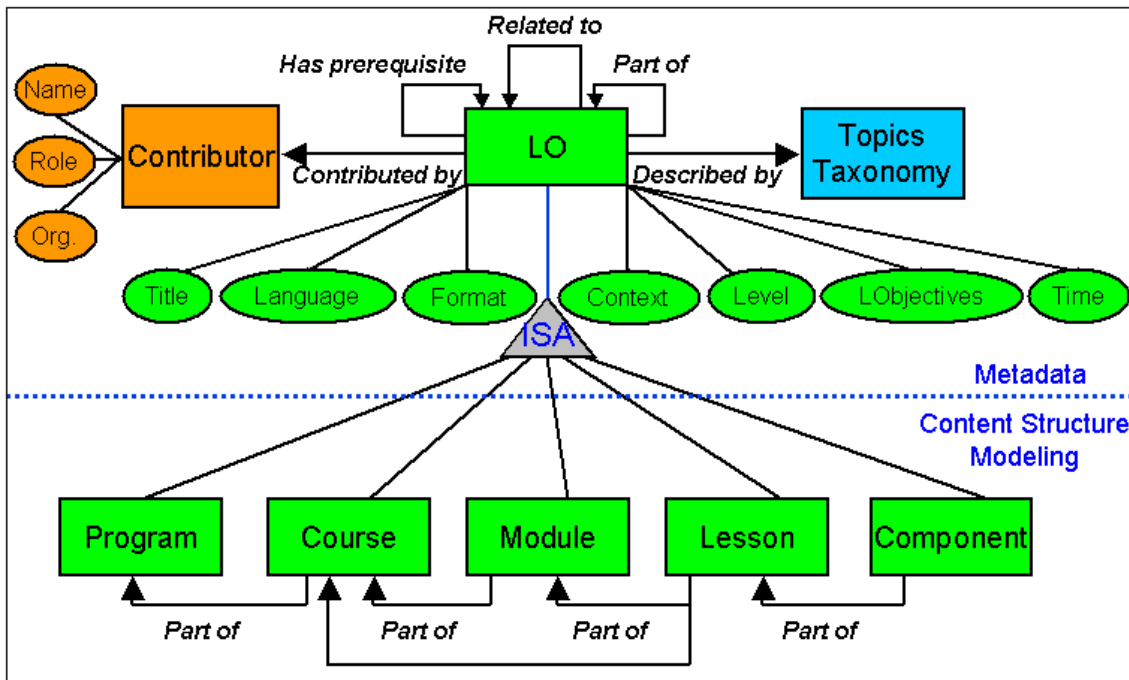
Figure 2: Conceptual Representation of LOs

2.2.1 Main descriptive attributes of Learning Objects

As we can observe in Figure 2, there are five main kinds of attributes used to describe a LO:

1) *Contributor of a LO:* a contributor of a LO is primarily a person or an organization. There is a standard way of referencing this relation with DC elements (*DC.contibutor, DC.creator, DC.publisher*). All other standards just add inconsequential information about the contributor. For example, as we will examine in Section 3.2.1, IEEE LOM expands the set of the contributor roles to include more specific role definitions.

2) *Subject Registration of a LO:* the subject of a LO can refer to a term in a Topics Taxonomy. This attribute has not yet been stabilized to a specific form, such as the ACM Computing Classification System[12], and every standard uses its own taxonomy function. For example, DC has a specified element called *subject*, while IEEE LOM provides a whole *classification* category.

3) *Relationships between LOs:* every LO can be related to another LO by multiple kinds of relations, such as a *part of* relationship capturing learning material composition trees, a *prerequisite* relation capturing learning dependency graphs or a *related to* relation representing correlation networks. This is a consequence of the observation that a whole learning process (like a course) and its sub-components (like lessons, videos or presentations) or other relative processes (like a prerequisite course of a course) could be LOs.

4) *Subsumption (ISA) relations between a LO and its granularity* levels: when learning material does not appear in isolation, *structure* is needed to encompass a set of LOs in an educational unit. Every existing LO could belong in one of the general granularity levels, namely *Component, Lesson, Module, Course* and *Program*. These levels should be interconnected with *part of* relations in order to build a

---

[12] http://www.acm.org/class/1998/homepage.html

complete educational unit comprising these levels. Therefore, a *Component* is a part of a *Lesson*; a *Lesson* is a part of a *Module* or *Course*; a *Module* is a part of a *Course* and a *Course* is a part of a *Program*.

5) *General descriptive attributes*: we can also identify attributes describing the *content* (e.g., *title*, *language*, *format*) and *pedagogical value* (e.g., *context*, *level*, *learning objectives*, *time*) of a LO. As learning or educational objectives, we characterize the teaching-related objectives instructors aim to achieve by using the provided educational material in an educational environment [14]. In other words, a learning objective is an explicit statement of what the learner is intended to demonstrate after the learning has been completed [4]. The educational *context* and *level* refer respectively to the typical target audience context and its educational level (e.g., undergraduate studies, second year), while *time* refers to the typical learning time required to achieve the educational objectives.

The set of these attributes constitute the minimal set of attributes needed to describe a LO in order to facilitate its efficient indexing and retrieval.

2.2.2 Conceptual representation layers of Learning Objects

The conceptual representation of LOs can take place in two layers. These layers are presented in Figure 2 and are distinguished by the blue dotted line. More specifically, we can identify:

1) The *Metadata* (upper) layer: This level comprises the most significant descriptive properties of LOs of various granularity levels. That is, the LO attributes (e.g., title, language, format, context, level, learning objectives and typical learning time) and the LO relationships (e.g., "Related to", "Has prerequisite", "Part of", "Described by" and "Contributed by").

2) The *Content Structure Modeling* (lower) layer: This level comprises the composition structure of LOs at various granularity levels in order to exchange and share LOs between platforms and applications.

There is a tight relationship between Metadata and Content Structure Modeling. In Content Structure Modeling, Metadata are used for describing the LOs being structured. On the other hand, Metadata standards rely on the representation of the composition of LOs, as provided by the various Content Structure Models. A significant statement for Metadata and Content Structure Modeling is that both can define LO *prerequisites*. We will examine how prerequisites are defined in these layers in Section 3.

3. E–Learning standards

E-Learning metadata standards constitute formal specifications of the descriptive terms used to semantically annotate educational material of all kinds. In this section, we will present the main information concepts involved in e-Learning standards. More specifically, an e-Learning standard *may be*:

1) *Formal* or "*de jure*", when a specification's status is designed and certified by an accredited body, like IEEE or ISO
2) "*de facto*", when the standard is adopted and used by the majority of users.

An e-Learning standard *is needed for*:

1) *Durability*. There is no need for modification as versions of system software change
2) *Interoperability*. We can have interoperability across a wide variety of hardware, Operating Systems, Web browsers and Learning Management Systems. This is the most significant advantage of the presence of an e-Learning standard
3) *Accessibility*. The operations of indexing and tracking can be handled to meet our demands
4) *Reusability*. There is possible modification and use by many different development tools.

An e-Learning standard *addresses*:

1) *Learning Object descriptions*, with the use of *metadata*
2) *Content sharing/packaging*, using interoperable *Content Structure Models*.

In the next two subsections, we explore the characteristics of e-Learning metadata standards referring both to the descriptions and the Content Structure Modelling of LOs.

3.1   Main e-Learning standards

From the variety of the e-Learning standards proposed from time to time, we can distinguish the following six, widely accepted metadata standards listed below:

1) **IMS** (Instructional Management System)
2) **ARIADNE** (Alliance of Remote Instructional Authoring & Distribution Network for Europe)
3) **DC** (Dublin Core)
4) **IEEE/LOM** (Institute of Electrical and Electronic Engineers/Learning Object Metadata)
5) **ADL/SCORM** (Advanced Distributed Learning/Sharable Courseware Object Reference Model)
6) **AICC** (Aviation Industry CBT Committee), where CBT stands for Computer Based Training

Their main focus and interrelationships are depicted graphically in Figure 3, which represents an IMS-centred overview of the main e-Learning standards.
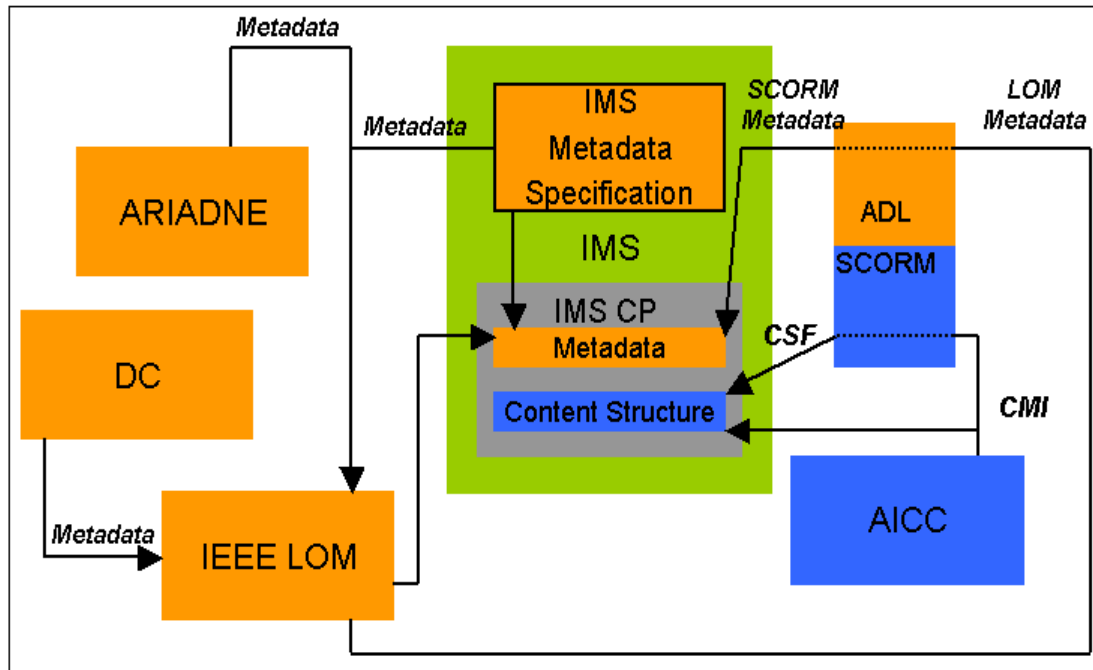
Figure 3: Interrelationships of e-Learning standards

The orange boxes represent the standards for *Metadata* specification (namely, ARIADNE, DC, IEEE LOM, IMS Metadata Specification, ADL) and the blue boxes the standards for *Content Structure Modelling* (namely, SCORM, AICC). We can observe that IMS and ADL/SCORM handle both Metadata specification and Content Structure Modelling. More precisely:

- The arrows labelled *Metadata* connecting IMS, ARIADNE and IEEE LOM represent the joint proposal of the first two standards that led to the creation of the IEEE LOM standard
- The arrow labelled *Metadata* from DC to IEEE LOM shows the contribution of DC on some metadata elements of IEEE LOM
- The arrow named *CMI* represents the AICC CMI (Computer Managed Instruction) Model for content structuring
- The arrow named *CSF* represents the SCORM CSF (Content Structure Format) Model for content structuring
- The dotted line in the lower (blue) part of ADL/SCORM indicates that SCORM CSF is derived from AICC CMI
- The dotted line in the upper (orange) part of ADL/SCORM shows the mapping of IEEE LOM Metadata to SCORM Metadata
- The grey box labelled *IMS CP* in IMS represents the IMS CP (Content Packaging) specification, which comprises a part for Metadata and a part for *Content Structure* organizations
- The arrows pointing at the *Metadata* part of IMS CP indicate the standards that can be used there, like IEEE LOM, IMS Metadata, SCORM Metadata or others
- The arrows pointing at the *Content Structure* part of IMS CP show the standards that can be used there, like SCORM CSF, AICC CMI or others. Note that, IMS CP can also use the *Table Of Contents (TOC)* Model for content structuring.

Subsequent subsections provide a more in-depth description by means of example metadata records of the e-Learning standards mentioned above.

3.2   Standards for Learning Objects Metadata

3.2.1   IEEE LOM

As already mentioned, IEEE LOM [5, 6] has been created from the joint proposal of IMS and ARIADNE to IEEE. It cooperates with DC by using DC elements for the base definition of some LOM elements. For example, the *identifier*, *title*, *language* and *description* elements in LOM are based on the DC elements *DC.identifier*, *DC.title*, *DC.language* and *DC.description,* respectively. This standard:

- specifies the syntax and semantics of LO Metadata using XML DTDs
- provides the attributes required to adequately describe a LO (e.g., element name, data type, definition, vocabulary, field length)
- is focused on a minimal set of attributes for the management, location and evaluation of LOs
- constitutes the most comprehensive metadata standard till now and forms the basis of almost all existing implementations of metadata specifications for LOs.

The *Royal Institute of Technology*[13] in Stockholm provides the RDF/S (Schema) [3, 11] bindings [12] for the last version of LOM that has been released, namely LOM Final Draft v1.0 [5].

More specifically, IEEE LOM specifies nine categories for over 70 metadata elements associated with LOs. These categories are shown in Table 1. All nine categories are *optional*; hence a LOM instance with no values for any of the elements still qualifies as a metadata record for the LOM standard. In this sense, most of the existing implementations of LO metadata conform to IEEE LOM; they generally define some of the LOM elements to be mandatory and extend this set of elements with their own "custom" elements.

General	Groups the general information describing a LO as a whole
Life Cycle	Describes the history and current state of a LO and those who have affected the LO during its evolution
Meta-Metadata	Describes the specific information about the metadata record itself (e.g., who created this record, how and when)
Technical	Describes the technical requirements and characteristics of a LO
Educational	Describes the key educational or pedagogic characteristics of a LO
Rights	Describes the intellectual property rights and conditions of use for a LO
Relation	Defines the relationship between a LO and other targeted LOs, if any
Annotation	Provides comments on the educational use of a LO, who created this annotation and when
Classification	Describes where a LO is placed within a particular classification system

Table 1: IEEE LOM Categories

---

[13] http://www.kth.se/eng/

The LOM standard respects the general granularity hierarchy of LOs depicted in Figure 2. In particular, it has established six granularity levels, which are shown in Table 2:

First level (higher)	*Curriculum*, like Program
Second level	*Course*
Third level	*Unit* (higher level Module)
Fourth level	*Topic* (lower level Module)
Fifth level	*Lesson*
Sixth level (lower)	*Fragment*, like Component

Table 2: IEEE LOM Granularity Levels

3.2.1.1 IEEE LOM CS252 example

In order to exemplify the use of IEEE LOM for annotating learning material, we present a metadata record conforming to the IEEE LOM Final Draft v1.0 [2]. This metadata record refers to the example course CS252 offered by the Computer Science Department of the University of Crete. Instead of an XML document, we presented textually the values of the elements ordered by the category they belong to and the index that the LOM Final Draft v1.0 assigns to them:

1.1.1. *General.Identifier.Catalog* "URI"

1.1.2. *General.Identifier.Entry* "http://www.csd.uoc.gr/~hy252"

1.2. *General.Title* ("en","CS252: Object Oriented Programming")

1.3. *General.Language* "en", "gr"

1.4. *General.Description* ("en", "This course analyzes the main principles and characteristics of Object Oriented Programming"), ("gr", "Το μάθημα αυτό αναλύει τις βασικές αρχές και ιδιότητες του Οντοκεντρικού Προγραμματισμού")

1.5. *General.Keyword* ("en", "Object Oriented Programming"), ("gr","Οντοκεντρικός Προγραμματισμός")

1.6. *General.Coverage* ("en", "University of Crete, Spring 2002, Greece"), ("gr", "Πανεπιστήμιο Κρήτης, Άνοιξη 2002, Ελλάδα")

1.7. *General.Structure* ("LOM v1.0", "linear")

1.8. *General.Aggregation Level* ("LOM v1.0", "3")

The Aggregation Level element value depends on the granularity level of the LO. Here, the LO CS252 is a course, so the value is 3 (Level 3).

2.2.	*Life Cycle.Status*	("LOM v1.0", "final")
2.3.1.	*Life Cycle.Contribute.Role*	("LOM v1.0", "author")
2.3.2.	*Life Cycle.Contribute.Entity*	vCard of Dr. Vassilis Christophides
2.3.3.	*Life Cycle.Contribute.Date*	"2002-02-10"

3.2.1.	*Meta-Metadata.Contribute.Role*	("LOM v1.0", "creator")
3.2.2.	*Meta-Metadata.Contribute.Entity*	vCard of Miltos Stratakis
3.2.3.	*Meta-Metadata.Contribute.Date*	"2002-11-18"
3.3.	*Meta-Metadata.Metadata Schema*	"LOM v1.0"

Caution: LOM v1.0 means the LOM Final Draft v1.0 Schema.

3.4.	*Meta-Metadata.Language*	"en"

4.1.	*Technical.Format*	"text/html", "application/pdf"
4.3.	*Technical.Location*	"http://www.csd.uoc.gr/~hy252"

4.4.1.1. *Technical.Requirement.OrComposite.Type*

("LOM v1.0", "browser")

4.4.1.2. *Technical.Requirement.OrComposite.Name*

("LOM v1.0", "netscape communicator")

4.4.1.1. *Technical.Requirement.OrComposite.Type*

("LOM v1.0", "browser")

4.4.1.2. *Technical.Requirement.OrComposite.Name*

("LOM v1.0", "ms-internet explorer")

5.1.	*Educational.Interactivity Type*	("LOM v1.0", "expositive")
5.2.	*Educational.Learning Resource Type*	("LOM v1.0", "index")
5.3.	*Educational.Interactivity Level*	("LOM v1.0", "3")

Interactivity Level 3 means that this LO has medium interactivity with the learner.

5.4.	*Educational.Semantic Density*	("LOM v1.0", "4")

Semantic Density 4 means that this LO has high semantic density, since it consists of refined pdf documents.

5.5.	*Educational.Intended End User Role*	("LOM v1.0", "learner")
5.6.	*Educational.Context*	("LOM v1.0", "higher education")
5.7.	*Educational.Typical Age Range*	

("en", "suitable for university learners")

5.8.   *Educational.Difficulty*   ("LOM v1.0", "3")

Difficulty 3 means that CS252 has medium difficulty for the typical intended target audience.

5.9.   *Educational.Typical Learning Time*   "P4M" (that is 4 months)

5.11.   *Educational.Language*   "en", "gr"

6.1.   *Rights.Cost*   ("LOM v1.0", "no")

6.2.   *Rights.Copyright and Other Restrictions*   ("LOM v1.0", "no")

7.1.   *Relation.Kind*   ("LOM v1.0", "haspart")

7.2.1.1. *Relation.Resource.Identifier.Catalog*   "URI"

7.2.1.2. *Relation.Resource.Identifier.Entry*

                    "http://www.csd.uoc.gr/lectures/CS252Basics.pdf"

7.2.2. *Relation.Resourse.Description*

            ("en", "Java Programming Basics is an introductory lesson of the CS252 course"),

            ("gr", "Τα βασικά χαρακτηριστικά του προγραμματισμού σε Java είναι εισαγωγικό μάθημα στον Οντοκεντρικό Προγραμματισμό")

7.1.   *Relation.Kind*   ("LOM v1.0", "requires")

7.2.1.1. *Relation.Resource.Identifier.Catalog*   "URI"

7.2.1.2. *Relation.Resource.Identifier.Entry*   "http://www.csd.uoc.gr/hy150"

7.2.2. *Relation.Resourse.Description*

            ("en", "CS150 is a prerequisite course of CS252"),

            ("gr", "Το HY150 είναι προαπαιτούμενο του Οντοκεντρικού Προγραμματισμού")

In the previous Relation category instance we defined a *prerequisite* of CS252. This is the standard way that IEEE LOM defines LO *prerequisites*.

7.1.   *Relation.Kind*   …

7.2.1.1. *Relation.Resource.Identifier.Catalog*   …

7.2.1.2. *Relation.Resource.Identifier.Entry*   …

7.1.   *Relation.Kind*   …

7.2.1.1. *Relation.Resource.Identifier.Catalog*   …

7.2.1.2. *Relation.Resource.Identifier.Entry*   …

In this metadata record we have multiple *Relation* category instances for the definition of the multiple relations of the CS252 Course with other LOs. The kinds of relations that IEEE LOM Final Draft v1.0 provides are shown in Table 3.

Relation Kind	*Value space in IEEE LOM*
Is Part Of	Ispartof
Has Part	Haspart
Is Version Of	Isversionof
Has Version	Hasversion
Is Format Of	Isformatof
Has Format	Hasformat
References	References
Is Referenced By	Isreferencedby
Is Based On	Isbasedon
Is Basis For	Isbasisfor
Requires	Requires
Is Required By	Isrequiredby

Table 3: IEEE LOM Relation Kinds

Let us now see how CS252 can be described using terms from the *ACM Computing Classification System*[14]. The above IEEE LOM metadata record continues as follows:

9.1. *Classification.Purpose*   ("LOM v1.0", "educational level")

9.2.1. *Classification.Taxon Path.Source*   ("en", "ACM")

9.2.2.1. *Classification.Taxon Path.Taxon.Id*   "D.1.5"

9.2.2.2. *Classification.Taxon Path.Taxon.Entry*

>   ("en","Object-oriented Programming")

9.3. *Classification.Description*

>   ("en", "A university second-year course introducing OOP")

9.4. *Classification.Keyword*   ("en", "OOP courses")

This *Classification* category instance places the subject of CS252 in a particular term within *ACM* taxonomy with ID *"D.1.5"* and label *"Object-oriented Programming"*.

---

[14] http://www.acm.org/class/1998/

3.2.1.2 IEEE LOM Metadata: What do they actually provide?

Figure 4 shows the undergraduate courses of CSD and their *prerequisite* interrelations, if any. The course to which an arrow ends is a *prerequisite* of the course from which this arrow starts. An IEEE LOM metadata record of CS252 can define the *prerequisite courses* of this course (with Relation.Kind = Requires) and the *courses that CS252 is a prerequisite course* (with Relation.Kind = Is Required By). In Figure 4, the area this metadata record applies to is highlighted with a red-coloured, dashed rectangle, where:

- CS150 designates the course "Programming in C"
- CS359 designates the course: "Web Programming".

As we can observe, Figure 4 represents a *Directed Acyclic Graph (DAG)* of the *prerequisite* relations of the courses at the CSD Undergraduate Program.

Practically, we rarely say, "*CS252 course requires the "Variables and Functions Definitions" lesson of the CS150 course*". We usually say, "*CS252 course requires CS150 course*". Therefore, we can consider that the *prerequisite* relations are usually applied to LOs of the same granularity level. However, in the case of prerequisites between modules and lessons, we can have statements, such as "The "*Language Fundamentals*" module requires the "*Designing Classes*" lesson" or "The "*Searching and Sorting*" lesson requires the "*Language Fundamentals*" module". Hence, the above consideration is incomplete, as long as there are prerequisite relations between LOs of two different granularity levels, *Module* and *Lesson*. In fact, if we take a closer look at *Module* and *Lesson* granularity levels, we will find out that these two levels are *approximate*. Actually, LOs of both levels could be direct divisions of a *Course*. Therefore, we can extend our consideration to this: The *prerequisite* relations are usually applied to LOs of the same or approximate (case *Module* and *Lesson*) granularity level. With our new extended consideration and the observation made in Figure 4, we can assert that with the use of IEEE LOM metadata we can provide a *DAG* of the *prerequisite* relations between LOs of the same or approximate granularity level.

Figure 5 presents the structure of CS252 through its lessons using IEEE LOM Metadata Relation Kinds. The blue arrows represent the IEEE LOM "*Has Part*" relations between the course and the lessons this course comprises and the red arrows represent the IEEE LOM "*Requires*" relations (aka the *dependencies*) between these lessons. As we can observe, the blue arrows yield a *tree* representation of CS252. Therefore, a *composite* LO (e.g., a course) containing LOs of lower granularity levels (e.g., lessons) can be represented by a *tree* using IEEE LOM Metadata. Also, keeping in mind the conclusions made from Figure 4, we can easily figure out that the red arrows give us a DAG representation of the *prerequisite* relations amongst CS252 lessons. This new DAG is considered to be a lower level DAG than the one in Figure 4, as it contains LOs of lower granularity level (the granularity level *Lesson* is lower than the granularity level *Course*).

Generalizing the above observations, with the use of IEEE LOM Metadata we can create a tree representation for every composite LO and DAGs of the prerequisite relations between:

- LOs of the same (or approximately the same in case of *Module* and *Lesson*) granularity level, which are contained in a composite LO
- LOs of the same (or approximately the same in case of *Module* and *Lesson*) granularity level that the composite LO belongs to.
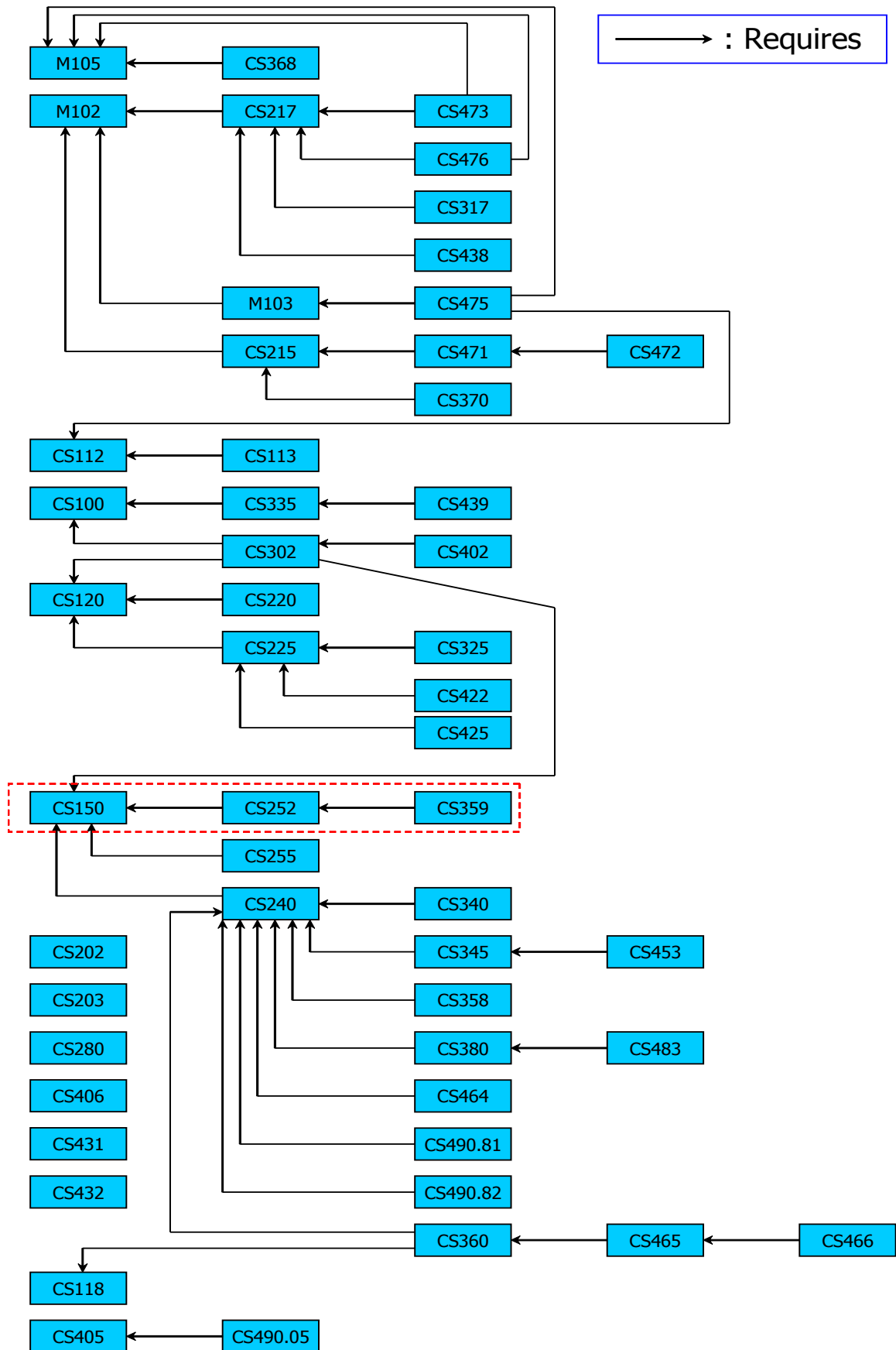
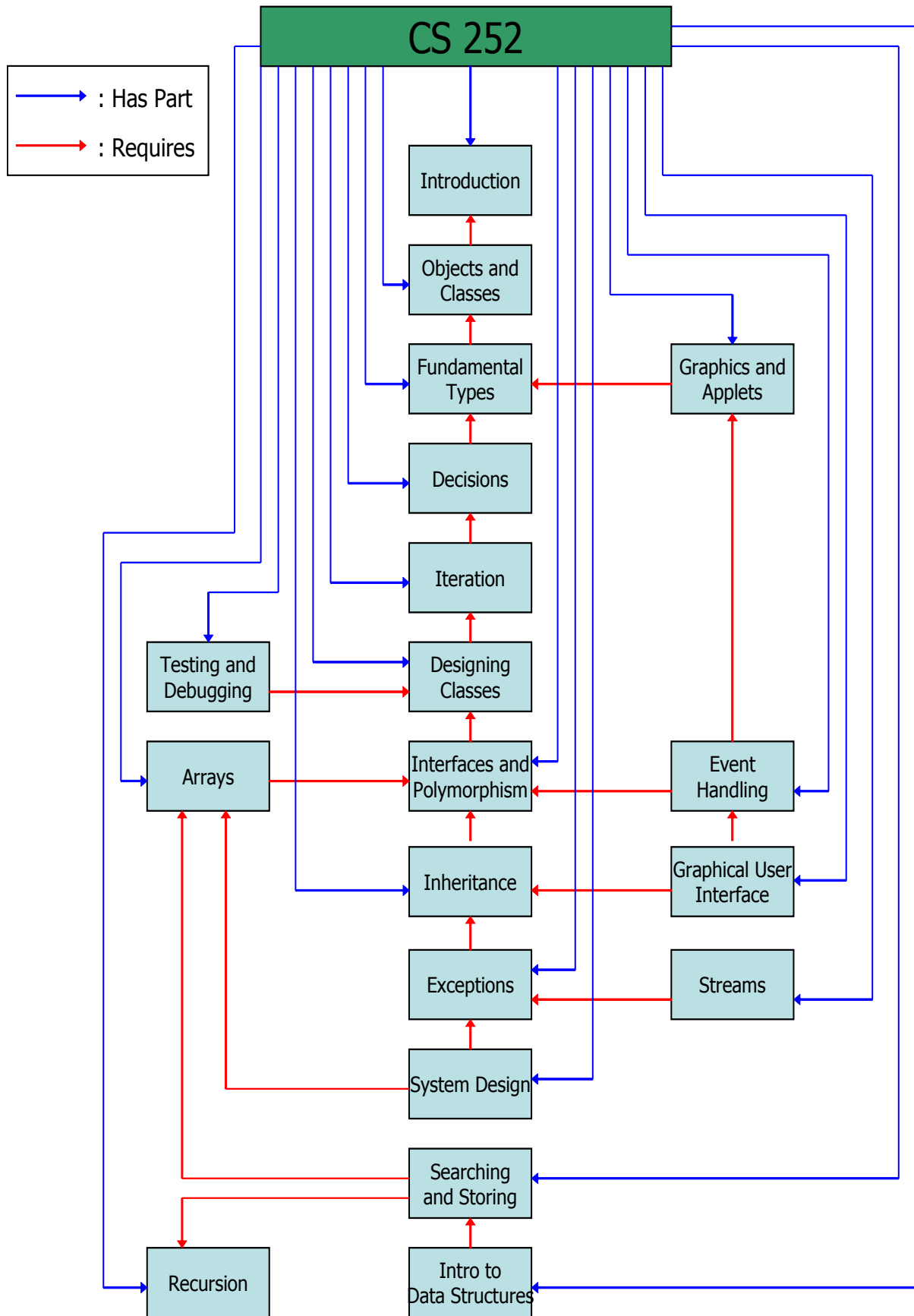***Figure 4: The prerequisites of CSD Undergraduate courses***

Figure 5: CS252 lessons and their dependencies

Figure 6 exhibits a graphical representation of two composite LOs that belong to the granularity level *Program*. The blue and the red arrows stand for the "*Has Part*" and "*Requires*" relations of IEEE LOM Metadata, respectively. Examining Figure 6, we can observe that the blue arrows create a *tree* representation for each of the two programs. These two *trees* are separated by black-coloured, dashed rectangles. Also, we can observe that the red arrows in Figure 6 create DAGs of five different levels:

1) The *Program* level DAG: This DAG contains the *prerequisite* relations between *Programs*. In Figure 6, the one DAG of this level is represented by an aqua-colored, dashed rectangle.

2) The *Course* level DAG: This DAG contains the *prerequisite* relations between courses of the same program. In Figure 6, the two *DAGs* of this level are represented by sky blue-colored, dashed rectangles.

3) The *Module/Lesson* level DAG: This *DAG* contains the *prerequisite* relations between the lessons, if any, and the modules, if any, of the same course. The lessons in this DAG are not part of the modules. In Figure 6, the five DAGs of this level are represented by green-colored, dashed rectangles.

4) The *Module to Lessons* level DAG: This DAG contains the *prerequisite* relations among the lessons of the same module. In Figure 6, the two DAGs of this level are represented by orange-colored, dashed rectangles.

5) The *Component* level DAG: This DAG contains the *prerequisite* relations between the components of the same lesson. In Figure 6, the thirteen DAGs of this level are represented by pink-colored, dashed rectangles.
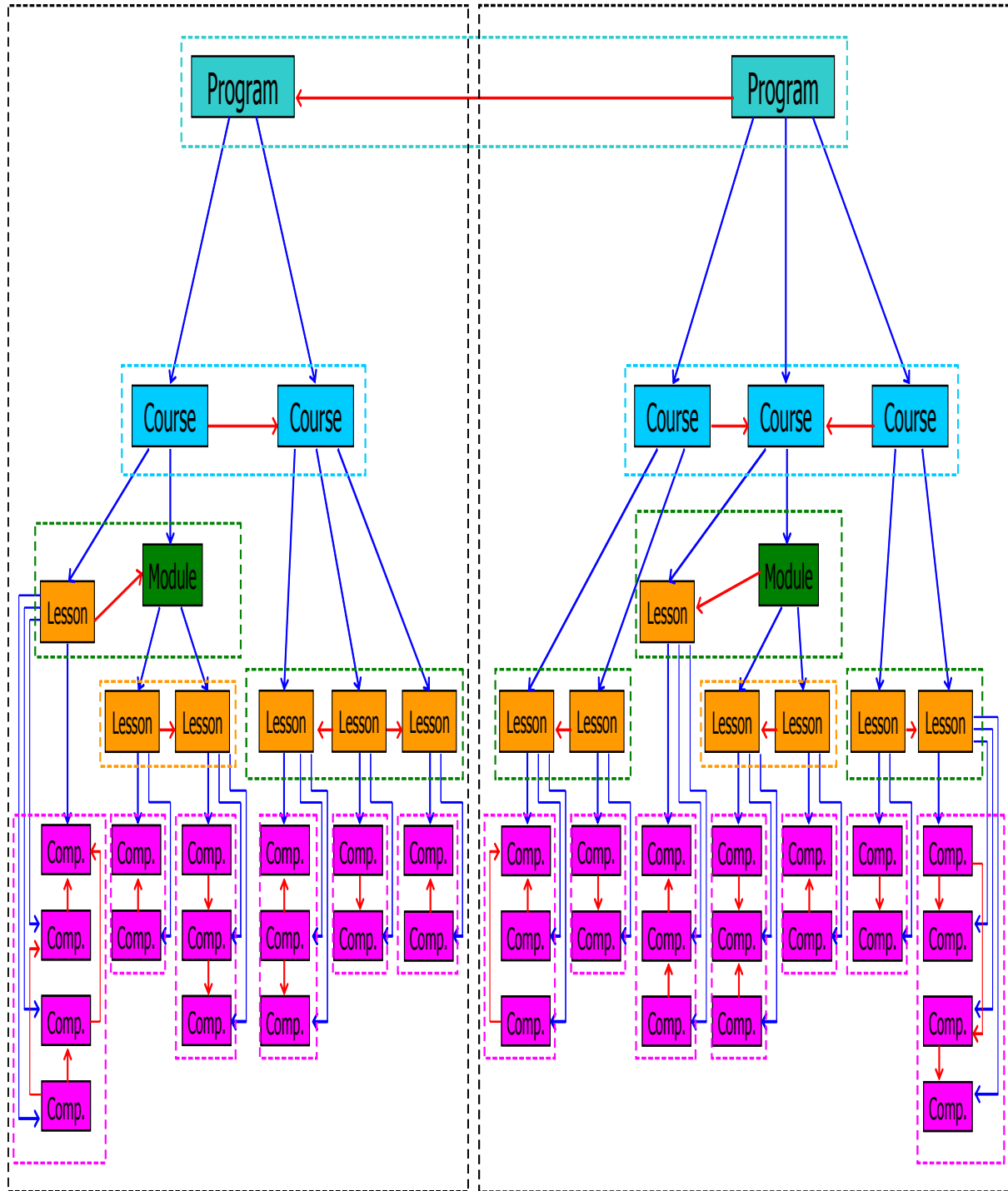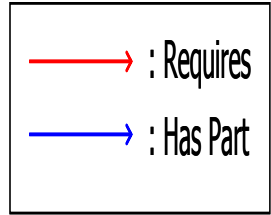
Figure 6: Example of LO trees and DAGs

3.2.2   ARIADNE Metadata

The European ARIADNE project ran from 1996-2000 and, with IMS (Instructional Management Systems), produced a set of recommendations for educational metadata that helped form the basis of the IEEE LOM. People involved in the ARIADNE project have since founded the ARIADNE Foundation, which seeks to build on the achievements of the original project. This metadata standard is trying to resolve two problems:

1) Easy and efficient indexing of LOs
2) Easy exploitation of the metadata by users looking for relevant pedagogical material.

We will present the ARIADNE Metadata Specification v3.2 [2]. According to this specification, there are a number of mandatory categories and an optional category, namely the *Annotations* category. The idea behind ARIADNE making some metadata elements mandatory is to address the conflict that exists between two principles it considers LO repositories should adhere to:

1) That metadata creation by LO authors or indexers should be as easy as possible
2) That search for useful LOs should be as easy as possible.

Thus, the mandatory categories constitute the minimal set of descriptive attributes that should allow for relatively good search capabilities without being too much of a burden to create. The seven categories of the ARIADNE Metadata Specification v3.2 are shown in Table 4:

CATEGORY	*MANDATORY/OPTIONAL*
General information of the resource	Mandatory
Semantics of the resource	Mandatory
Pedagogical Attributes	Mandatory
Technical characteristics	Mandatory
Conditions of use	Mandatory
Meta-metadata information	Mandatory
Annotations	Optional

Table 4: ARIADNE Metadata v3.2 categories

In the table above, the categories *Semantics of the resource* and *Pedagogical attributes* are the most important, as long as they describe the pedagogical value of a LO.

3.2.2.1   ARIADNE CS252 example

In order to present ARIADNE's metadata record structure, we present a metadata record of the course CS252 complying with the ARIADNE Metadata v3.2's most significant categories, i.e., *Semantics of the resource* and *Pedagogical Attributes*. This metadata record representation is analogous to the metadata record representation used in the IEEE LOM Final Draft v1.0 example presented above:

2.1.	*Semantics.Discipline Type*	"Natural, Exact or Engineer Sciences"
2.2.	*Semantics.Discipline*	"Computer Science"
2.3.	*Semantics.SubDiscipline*	"Software Programming"
2.4.	*Semantics.Main Concept*	"Object Oriented Programming"
2.5.	*Semantics.Other Concepts*	"JAVA PL"

3.1.	*PA.End User Type*	"Learner"
3.2.	*PA.Document Type*	"Expositive"
3.3.	*PA.Document Format*	"Hypertext"
3.4.1.	*PA.Didactical Context.Country*	"gr"
3.4.2.	*PA.Didactical Context.Context*	"University Degree"
3.4.3.	*PA.Didactical Context.Level*	"2"

PA is the abbreviation of *Pedagogical Attributes*. Didactical Context Level 2 means that CS252 is a second-year course.

3.6.	*PA.Interactivity Level*	"Medium"
3.7.	*PA.Semantic Density*	"High"
3.8.	*PA.Pedagogical Duration*	129600 minutes
3.9	*PA.Granularity*	"Course or Course Template"

We can note that the differences between the metadata records of ARIADNE and IEEE LOM are imperceptible. However, the pedagogical duration (3.8 *Pedagogical Duration*) of the LO described *must be given in minutes*, a fact that causes frustration when finding the duration of LOs of high granularity levels, like courses or programs. Furthermore, an important drawback of the ARIADNE Metadata v3.2 is the lack of support for describing information about the relations of a LO, like *prerequisites*. The relations among LOs are a significant feature that must be included in a metadata record. Therefore, we can assume that ARIADNE v3.2 is not a complete metadata standard.

3.2.3 IMS Metadata

The metadata specifications of IMS use exactly the same elements as previous versions of IEEE LOM. The latest version of the IMS Metadata specification, v1.2.2 [7], is exactly the same as IEEE LOM Working Draft v6 [6].

Due to the large number of IEEE LOM elements, IMS distinguishes between two different specifications:
1) IMS *Core* (20 LOM elements, which are a reduced set of fundamental metadata)
2) IMS *Standard Extension Library* or IMS-*SEL* (the remaining LOM elements).

3.2.4 SCORM Metadata

ADL's Sharable Courseware Object Reference Model (SCORM) [13] uses IEEE LOM Metadata. It maps IEEE LOM elements into three learning content elements for providing the missing link between general metadata specifications and specific content models. These three learning content elements are considered to be the granularity levels of SCORM, which are from lower to higher:

1) ***Raw media*** (like Components, Fragments)
2) ***Content*** (like Lessons, Modules, Units)
3) ***Course***

On the whole, ADL SCORM specifies how a user can build a course by aggregating content objects as a course tree; what is the type of these content objects and how they are launched and interact with a learning management system; and what kind of data content objects exchange within a learning management system. *Sharable Courseware Object Reference Model* (SCORM) was changed to *Sharable Content Object Reference Model* (SCORM) for all versions after 1.0.

3.3   Content Structure Modeling

3.3.1 AICC

The Aviation Industry CBT Committee (AICC) is an international association of technology-based training professionals, which develops guidelines for aviation industry in the development, delivery, and evaluation of CBT (Computer-Based Training) and related training technologies. In the AICC e-Learning metadata standard, we can distinguish three types of course elements:

1) ***Assignable Units (AUs),*** the smallest educational elements that can be presented to a student, like an HTML page.
2) ***Blocks,*** which are used for nesting. A block can nest AUs and other blocks, which are called *nested blocks*
3) ***Objectives,*** which are used to define course requisites and represent goals that must be achieved in the course. They may be *simple* or *complex* (when they contain simple or multiple AUs and blocks).

From the above course elements, AUs and blocks constitute AICC's structure elements. Objectives are not structure elements.

3.3.1.1 AICC CMI Guidelines for Interoperability

This AICC specification provides a Content Structure Model used primarily for interoperability. The main idea behind this model is that a course is a collection of AUs, blocks and objectives. A course example in AICC CMI and the identification of its elements is shown in Figure 7:

Assignable Units (Ax)	A1, A3, A4, A9, A17, A22
Blocks (Bx)	B6, B7
Objectives (Jx)	J2, J5

Figure 7: AICC CMI course example (Element identification)

The *Course Structure* Table of the course in Figure 7 is shown in Table 5. This table does not contain objectives, as long as objectives are not structure elements.

Block	*Member*	*Member*	*Member*	*Member*	*Member*
Root	B6	B7	A1	A4	A17
B6	B7	A22			
B7	A3	A9			

Table 5: Course Structure Table of Figure 7

In AICC CMI Guidelines for Interoperability we can define *prerequisites* among the structure elements of a course. For instance, in Figure 7 let AU A4 be a prerequisite of block B6 and B6 be a prerequisite of AU A17. This hypothesis is shown in Figure 8.
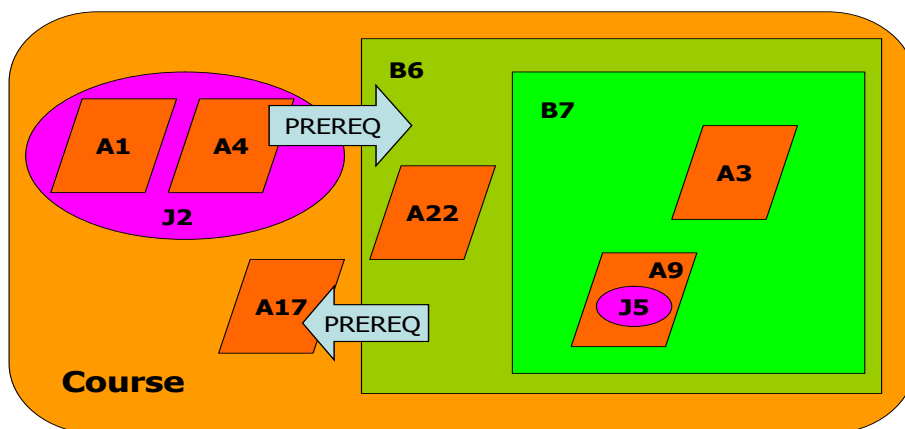

Figure 8: The course in Figure 7 with prerequisites

The *Prerequisite Table* of the course in Figure 8 is shown in Table 6.

Structure Element	Prerequisite
A22	A4
A3	A4
A9	A4
A17	B6

Table 6: Prerequisite Table of the course

As we can observe, in the *Structure Element* column of Table 6, we can only register AUs. In Figure 8, block B6 contains AU A22 and block B7, B7 contains AUs A3 and A9 and AU A4 is a prerequisite of B6, so A22, A3 and A9 have A4 as a prerequisite.

The *Course Structure Table* and the *Prerequisite Table* are two of the files that AICC CMI Guidelines for Interoperability provide for storing the static and dynamic structure of courses exchanged between e-Learning systems. The relation of these files with their type and the corresponding contents is shown in Table 7. When these files are transferred from one platform to other compliant platforms, the course can be recreated straightforwardly.

Name	Content	Type
Course	Basic information about a course, including a textual description	Group/Keyword
Descriptor	System generated IDs, title and descriptions of each element in a course: AUs, Blocks, Objectives and Complex Objectives	Table
Assignable Unit	Information about each AU, including data needed to launch it	Table
Course Structure	Static Course Structure Table	Table
Objectives Relationships	Shows the relationship (if any) of each objective in a course to other objectives, blocks and AUs	Table
Prerequisite	Prerequisite Table. It indicates prerequisites for entering each AU	Table
Completion Requirements	Completion Table. It indicates the requirements for completion of each block or complex objective whose completion cannot be determined by the defaults	Table

Table 7: AICC CMI Course Structure Files

3.3.1.2 AICC Granularity Levels

In order to avoid inconsistency due to the ability of having multiple levels of nesting (blocks into blocks into blocks, etc.), AICC has established a reference hierarchy with ten levels. These levels, shown in Table 8, are considered to be the AICC *granularity levels*.

First level (higher)	*Curriculum*, a grouping of related courses
Second level	*Course*, a complete unit of training
Third level	*Chapter*, a meaningful division of a course. A grouping of subchapters or lessons (high level *block*)
Fourth level	*Subchapter*, a meaningful division of a chapter. A grouping of lessons or modules (middle level *block*)
Fifth level	*Module*, logical group of lessons (one or more). A meaningful division of a course, chapter or subchapter (low level *block*)
Sixth level	*Lesson/AU*. Has three meanings : 1. A meaningful division of learning that is accomplished by a student in a continuous effort 2. A grouping of instructions that is controlled by a single executable computer program 3. A unit of training is a logical division of a subchapter, chapter or course
Seventh level	*Topic*, logical divisions of a lesson
Eighth level	*Sequence*
Ninth level	*Frame/Screen*
Tenth level (Lower)	*Object*, component of a screen or frame

Table 8: AICC Granularity Levels

In Table 8, the *Sequence* and *Frame/Screen* levels are mainly focused on technical specifications of LOs. We can observe that AICC respects the general granularity hierarchy of LOs. Also, we can assert that *Chapters, Subchapters* and *Modules* are essentially the "new" names of the AICC's *blocks*.

3.3.1.3 AICC versus IEEE LOM

Table 9 shows the granularity levels that the AICC and IEEE LOM (IMS) specifications use in several LO functions.

Function	AICC	IEEE LOM (IMS)
Outer Container	Course	Course
Nesting Container	Block	Unit
Content Aggregate	AU	Lesson
Reusable Media Element	Object ?	-

Table 9: AICC vs. IEEE LOM LO Granularity

We can observe that there is an analogy between the granularity levels used by AICC and IEEE LOM in the three first functions. As we have already mentioned, AICC's *Blocks* and IEEE LOM's *Units* represent *Modules* in the general granularity hierarchy of LOs. Also, *AUs* represent *Lessons* in AICC, so the correspondence is obvious. In LO function *Reusable Media*

Element, AICC is considering using *Objects*, while IEEE LOM has not made a suggestion till now.

3.3.2 SCORM Content Structure Format (CSF)

SCORM CSF [13] is derived from AICC CMI but is slightly differentiated. In particular,
- SCORM CSF is applicable not only to complete courses, but also to subsets of courses or to groups of courses
- SCORM CSF renames the term *AU* to *Sharable Content Object (SCO)*, but preserves its meaning.

The main elements of SCORM CSF's DTD, are shown in Figure 9:

Figure 9: SCORM CSF's DTD (Main elements)

The definitions of these elements are given below:
- *Content* element: The root level of *Content Structure* representation. This element represents the learning content being structured
- *globalProperties* element: The properties of the learning content as whole
- *block* element: A grouping of related structural elements. *Blocks* always contain other learning content elements
- *externalMetadata* element: The value of this element refers or points to the location of the metadata describing this learning content
- *curricularTaxonomy* element: The organizational methodology used to construct the learning content
- *identification* element: This element identifies the learning content context-specific information
- *prerequisites* element: An expression indicating what a student must have accomplished before beginning the learning content element. That is, the learning

content elements that a student must complete before beginning a block or a SCO. This element constitutes the standard way SCORM CSF defines LO prerequisites

- *sco* element: Like AU, SCO is the smallest element of instruction or testing presented to a student
- *blockAlias* element: A reference to a previously defined block. This element permits one block to be used more than once within a learning content
- *scoAlias* element: A reference to a previously defined *sco*. This element permits one SCO to be used more than once within a learning content
- *timeLimit* element: The time values or actions associated with this *sco* in this context
- *launch* element: The information needed by a Learning Management System (LMS) to launch a *sco*
- *masteryScore* element: The values to be used in this learning content context for tracking score within a *sco*

As we can observe from Figure 9:
- The *Content* element could be a course, a subset of a course or a group of courses
- The *globalProperties* element contains the data about the *Content* element
- The *block* element defines the structure of the *Content* aggregation
- A *Content* element constitutes one *block* element and zero or one *globalProperties* element (we have a question mark (?) in *globalProperties* element)
- The *sco* element constitutes a *scoAlias* element or an element which constitutes zero or multiple *externalMetadata* elements, one *identification* element and zero or one *prerequisites, timeLimit, launch* and *masteryScore* elements
- The green highlighted boxes represent the elements that lead us to the distinction of the existing relations between LOs in SCORM CSF. These are the *prerequisite* relation (the *prerequisites* element) and the *reference* relation (the *blockAlias* and *scoAlias* elements).

The most significant observation made from Figure 9 is the frequent use of the *prerequisite* relation between LOs. This is very important as in practice, learning contents usually have prerequisites or are prerequisites of other learning contents

3.3.3   IMS CP Information Model

For Content Structure Modelling, IMS provides the IMS CP Information Model [8, 9, 10]. This model:
- Is based on a set of data structures to provide content interoperability in an Internet environment
- Encapsulates a complete course, with its related metadata, in a single file
- Is very flexible in the representation of the learning content
- Co-operates significantly with other organizations.

The key element of this model is the *package*. Figure 10 shows a pictorial representation of the IMS CP Information Model.
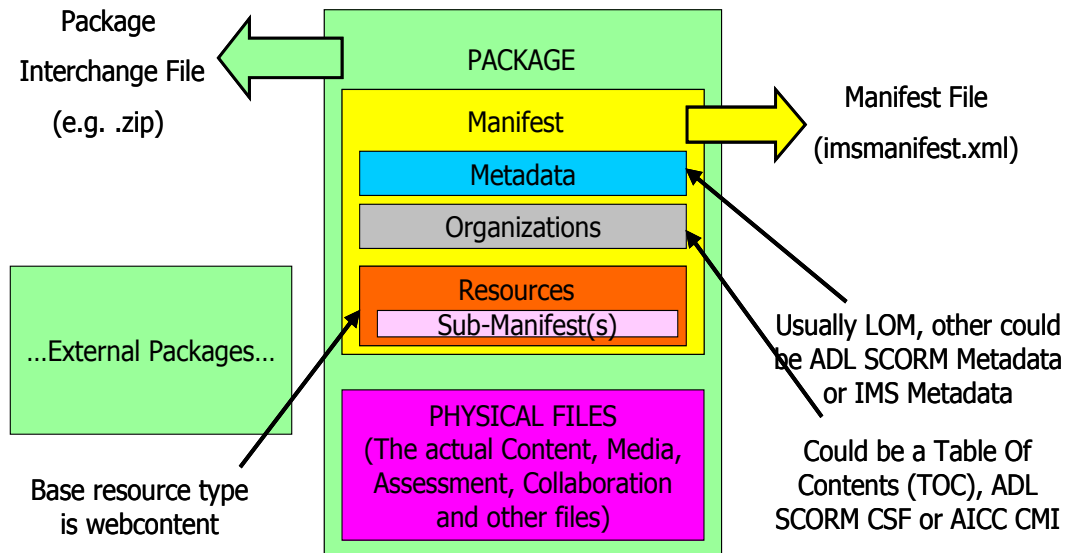
Figure 10: IMS CP Information Model

In the IMS CP Information Model, the *Package* element contains two main elements:

- A *manifest* element, an XML document with stable name (*imsmanifest.xml*) that describes the encapsulated contents and their organization.
- The *actual educational contents*, described in the *manifest*, such as media and text files, assessment objects or other data in file form.

The *Manifest* element consists of:

- *Metadata:* An XML element describing the *manifest* as a whole. For the metadata description, we can use IEEE LOM, IMS Metadata, SCORM Metadata or other metadata standards
- *Organizations:* An XML element describing zero or more static ways of organizing (aka, structuring) the instructional resources for presentation. For the structuring, we can use Table Of Contents (TOC), AICC CMI, SCORM CSF or other proposals
- *Resources:* A section which contains references to all of the actual resources and media elements needed for a manifest, including metadata describing the resources, references to any external files and sub-manifests that can contain zero or more logically nested manifests.

A *Manifest* has flexible scope and facilitates aggregation or unfolding. A package always contains a top-level manifest that can include sub-manifests associated to a part of the contents encapsulated into this package. For example, a content developer who wants to move multiple courses could create a manifest for each course and then aggregate all of them in a single package, with a top-level manifest describing the collection of courses as a curriculum.

Packages can be interpreted as logical directories, which contain:

- A manifest file (*imsmanifest.xml*)
- All format control documents for the manifest (e.g., DTD, XSD)
- A set of sub-directories containing the physical files
- The single file (e.g., .zip), where the logical directory of the *package* is placed, is named the *Package Interchange File*.

3.3.3.1 IMS CP Information Model CS252 Example

This subsection provides a metadata record of the *Manifest* of CS252 in IMS CP Information Model v1.1.2 [9]. For this record, we are using the IMS CP Schema v1.1 [10]. IMS CP Information Model v1.1.2 is not the latest version, as IMS CP Information Model v1.1.3 [8] has already been released. Instead for an XML document of the *Manifest* metadata record, we follow the same presentation style used for the Metadata records of IEEE LOM and ARIADNE above.

0.1.	*Manifest.Identifier*	The unique ID of this manifest
		(e.g., MANIFEST-CS252)

1.1.	*Metadata.Schema*	"IMS Content" (IMS CP Schema)
1.2.	*Metadata.SchemaVersion*	"1.1"
1.3.	*Metadata.{IMS Metadata}*	The IMS Metadata describing CS252
		(e.g., LOM 1.0 or IMS 1.2.2)

2.2.1. *Organizations.Organization.Identifier*

The unique ID of this organization element (e.g., TOC1)

2.2.3. *Organizations.Organization.Title*   "Lectures"

As we can notice, CS252 comprises a Table Of Contents (TOC) called "Lectures" that contains the lessons of this course. Next, we present the annotation of the lessons constituting the TOC "Lectures".

Caution: Org is the abbreviation for the Organization element.

2.2.4.1. *Organizations.Org.Item.Identifier*

The unique ID of this item (e.g., TOC1_ITEM1)

2.2.4.2. *Organizations.Org.Item.IdentifierRef*

The reference to the ID of this item in the *Resources* section (e.g., TOC1_RESOURCE1)

2.2.4.3. *Organizations.Org.Item.Title*

"Introduction: Programming Languages and Paradigms"

2.2.4.1.	*Organizations.Org.Item.Identifier*	TOC1_ITEM2
2.2.4.2.	*Organizations.Org.Item.IdentifierRef*	TOC1_RESOURCE2

2.2.4.3. *Organizations.Org.Item.Title*

"Java Programming Basics: Types, Variables, Operators"

2.2.4.1.	*Organizations.Org.Item.Identifier*	…
2.2.4.2.	*Organizations.Org.Item.IdentifierRef*	…

2.2.4.3. *Organizations.Org.Item.Title*   …

2.2.4.1. *Organizations.Org.Item.Identifier*   …
2.2.4.2. *Organizations.Org.Item.IdentifierRef*   …
2.2.4.3. *Organizations.Org.Item.Title*   …

Now, we are going to define the second TOC used in CS252, named "Assisting Lectures", which contains the assisting lessons of CS252. After this action, we import the lessons that constitute this TOC in a similar way as previously.

2.2.1. *Organizations.Org.Identifier*
                The unique ID of this organization element (e.g., TOC2)
2.2.3. *Organizations.Org.Title*   "Assisting Lectures"

2.2.4.1. *Organizations.Org.Item.Identifier*   TOC2_ITEM1
2.2.4.2. *Organizations.Org.Item.IdentifierRef*   TOC2_RESOURCE1
2.2.4.3. *Organizations.Org.Item.Title*
                "The Java Programming Environment"

2.2.4.1. *Organizations.Org.Item.Identifier*   TOC2_ITEM2
2.2.4.2. *Organizations.Org.Item.IdentifierRef*   TOC2_RESOURCE2
2.2.4.3. *Organizations.Org.Item.Title*
                        "Files, Streams, Filters and Strings"

2.2.4.1. *Organizations.Org.Item.Identifier*
2.2.4.2. *Organizations.Org.Item.IdentifierRef*   …
2.2.4.3. *Organizations.Org.Item.Title*   …

After declaring the main and assisting lessons of CS252, we are going to declare the references to them, using the *IdentifierRef* records defined above. This action takes place in the *Resources* section.

3.1.1. *Resources.Resource.Identifier*   TOC1_RESOURCE1
3.1.2. *Resources.Resource.Type*   "webcontent"
3.1.3. *Resources.Resource.HRef*
      "http://www.csd.uoc.gr/~hy252/lectures/CS252Intro.pdf"

3.1.1. *Resources.Resource.Identifier*   TOC1_RESOURCE2
3.1.2. *Resources.Resource.Type*   "webcontent"
3.1.3. *Resources.Resource.HRef*
      "http://www.csd.uoc.gr/~hy252/lectures/CS252Basics.pdf"

3.1.1. *Resources.Resource.Identifier*   TOC2_RESOURCE1
3.1.2. *Resources.Resource.Type*   "webcontent"

3.1.3. *Resources.Resource.HRef*

"http://www.csd.uoc.gr/~hy252/assist/CS252JavaProgrammingEnvironment.pdf"

3.1.1. *Resources.Resource.Identifier*   TOC2_RESOURCE2

3.1.2. *Resources.Resource.Type*   "webcontent"

3.1.3. *Resources.Resource.HRef*

   "http://www.csd.uoc.gr/~hy252/assist/streams_files.pdf"

3.1.1. *Resources.Resource.Identifier*   …

3.1.2. *Resources.Resource.Type*   …

3.1.3. *Resources.Resource.HRef*   …

   The *webcontent* type is the only current type supported by IMS CP Information Model v1.1.2. It is defined as the content that can be hosted or launched by an Internet Browser.

4. Conclusions

The purpose of this deliverable has been to overview the main characteristics of a LO in diverse e-Learning environments, by focusing on the definition, granularity and conceptual representation of LOs. Since several e-Learning standards have been developed to describe LOs and their relationships, we have paid particular attention to the two primary LO representation layers: *Metadata* and *Content Structure Modelling*. For instance, the most significant metadata standard, IEEE LOM, specifies a variety of bibliographic and technical properties of LOs, different relationships among LOs and enables metadata-based exchange, reuse and search of educational material. However, even though the IEEE LOM standard includes an educational category, no information is included to specify which instructional roles are or can be played by a LO within a course. It specifies properties only at a very basic abstraction level and does not support instructional design, like metadata about instructional models and theory and information about the use of LOs in learning processes. Thus, IEEE LOM concentrates on ***what*** should be taught and ***when***, rather than ***how*** to be taught or used for learning.

In fact, instructional principles are *not* addressed so far by the e-Learning standards presented in this deliverable, so learners *cannot* choose courses or lessons in a useful way to satisfy their educational needs. This fact reflects the intended use of these metadata standards as indexing schemes for basic cataloguing and sharing. The instructional framework of a given LO is implicit and it is assumed to be provided by a human agent (instructor). In this context, an educational metadata standard should not tell how to teach, but it should be able at least to provide information on *how* to specify pedagogical aspects of LOs. Specifying the author and the title of a LO is an *easy* way, but specifying instructional metadata, models and theory is a *better* way. If LOs are marked-up with sufficiently detailed pedagogic metadata (e.g., "this learning object is particularly useful for helping visual learners grasp such-and-such a concept"), and learners have profiles that match up with this (e.g., "this person learns things best when they are presented visually"), then a search utility over a learning object repository should be able to match users' learning styles as well as the content they require.

The metadata standards examined in this deliverable do not provide the means to key each LO to a learning objective in an exploitable way. Learning objectives essentially describe goals of training or education in terms of the knowledge, skills, or expected performance of a learner. Learning objectives may be associated with LOs of any granularity to guide the design of the instructional strategies to achieve these goals. Learning objectives are, hence, attractive to instructors and learners, because they can facilitate the learning process and the preparation of proprietary learning material. Therefore, what is needed – currently not provided by e-Learning standards - is the taxonomic distinction of learning objectives in order to effectively index cognitive, affective and psychomotor objectives. Outcomes may also be cross-referenced to general skills and competencies essential for socioeconomic success (communication skills, problem solving, critical thinking, numeracy, etc). Such an outcome framework facilitates management of learning programs within an organization as well as providing a common language for inter-institutional articulation and LO exchange [4]. To assess whether the learning objectives have been achieved, what is also needed is an assessment taxonomy accommodating not only the types of questions and methods used for assessment (e.g., multiple choice, fill-in-blanks, case study, etc), but also types of instruments (pretests, self-evaluation, assignment, etc), assessment criteria and rubrics, metacognitive factors and integration across multiple objectives. Such indexing enables the provision of alternate assessment strategies for the same content in different circumstances [4].

The Educational Modeling Language[15] (EML) proposed by the Open University of Netherlands has been recently chosen by IMS as the basis of the newly established Learning Design Specification[16] whose aim is "*the development of a framework that supports pedagogical diversity and innovation, while promoting the exchange and interoperability of e-Learning materials*". EML extends IEEE LOM and all other specifications by explicitly applying educational models and theories of learning on LOs and by providing pedagogical roles for the users. Thus, EML and the Learning Design Specification seem to provide the educational context that is lacking in current e-Learning metadata specifications.

Another aspect not confronted by existing e-Learning metadata standards is how to *trust* and *evaluate* the source of LOs. Due to the proliferation of educational material, it is a challenge to build a web of trust that would enable learners to evaluate and credit the authority and authenticity of educational material or annotate all kinds of LOs according to their perception of its value. Another aspect which is not covered by metadata standards is *copyright*. The IEEE LOM standard does not support the specification of usage rights; instead digital rights languages, such as the Open Digital Rights Language[17] (ODRL) and Digital Object Identifier (DOI)[18] could be incorporated into metadata specifications for this particular purpose. The ODRL specification supports an extensible language and vocabulary (data dictionary) for the expression of terms and conditions over any content including permissions, constraints, obligations, conditions, and offers and agreements with holders of digital rights. On the other hand, DOI is a system for identifying and exchanging intellectual property in the digital environment by providing a framework for managing intellectual content, for linking customers with content suppliers, for facilitating electronic commerce, and enabling automated copyright management for all types of media. Using digital management to protect the copyright of LOs in an efficient way is an issue that calls for research activity.

---

[15] http://eml.ou.nl

[16] http://www.imsproject.org/learningdesign/ldv1p0/imsld_infov1p0.html

[17] http://odrl.net/

[18] http://www.doi.org/

5. References

[1] Luis E. Anido-Rifon et al., *Virtual Learning and Computer Based Standardization. Issues and Trends*,

http://www.ics.forth.gr/~mstratak/e-Learning/anidoSurveys.zip

[2] ARIADNE (2002), *ARIADNE Educational Metadata Recommendation – V3.2*,

http://www.ariadne-eu.org/en/publications/metadata/ams_v32.html

[3] Brickley, D., Guha, R.V., *Resource Description Framework Schema (RDF/S) Specification 1.0.* W3C Candidate Recommendation (2000)

[4] Epsilon Learning Systems, *Learning Objects*, http://www.epsilonlearning.com/objects.htm

[5] IEEE (2002), *Draft Standard for Learning Object Metadata*,

 http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf

[6] IEEE (2001), *Draft Standard for Learning Object Metadata*, http://ltsc.ieee.org/wg12/LOM_WD6.pdf

[7] *IMS Learning Resource Metadata Specification v1.2.2 XSD*,

 http://www.imsglobal.org/xsd/imsmd_v1p2p2.xsd

[8] *IMS Content Packaging Specification v1.1.3 XSD*,

http://www.imsglobal.org/content/packaging/cpv1p1p3/validation/xml_schema/imscp_v1p1p3.xsd

[9] IMS (2001), *IMS Content Packaging Information Model: Version 1.1.2 Final Specification*,

 http://www.imsproject.org/content/packaging/cpv1p1p2/imscp_infov1p1p2.html

[10] *IMS Content Packaging Schema (IMS CONTENT) v1.1 XDR*,

 http://www.imsglobal.org/xml/content/packaging/bindings/xml_data/IMS_CONTENTv1p1.xdr

[11] Lassila, O., Swick, R., *Resource Description Framework (RDF) Model and Syntax

Specification*. W3C Recommendation (1999)

[12] Mikael Nilsson, *IEEE Learning Object Metadata RDF binding*, http://kmr.nada.kth.se/el/ims/md-lomrdf.html

[13] SCORM Version 1.0, *Sharable Courseware Object Reference Model Version 1.0*.

http://www.adlnet.org/index.cfm?flashplugin=1&fuseaction=scorm10

[14] B. Simon, *Do e-Learning standards meet their challenges?* In Proceedings des Workshops "Standardisierung im eLearning", Frankfurt, Deutschland, April, 2002.

http://www.wu-wien.ac.at/usr/wi/bsimon/publikationen/e-Learning-standards-bsi.pdf

[15] B. Simon, J. Quemada, *A Reflection of Metadata Standards Based on Reference Scenarios*
 http://www.wu-wien.ac.at/usr/wi/bsimon/publikationen/SimonQuemada-ReflectionOnMetadataStandards.pdf

[16] L. Stojanovic, S. Staab, R. Studer, *eLearning based on the Semantic Web*. WebNet 2001 - World Conference on the WWW and the Internet. Orlando, Florida, USA, Oct. 23-27, 2001.

Appendix

1.   RDF Representation of the CS252 Example

```
<?xml version="1.0" ?>
<rdf:RDF
     xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:rdfs = "http://www.w3.org/2000/01/rdf-schema#"
     xmlns:dc = "http://purl.org/dc/elements/1.1/"
     xmlns:dcterms = "http://purl.org/dc/terms/"
     xmlns:dctype = "http://purl.org/dc/dcmitype/"
     xmlns:vCard = "http://www.w3.org/2001/vcard-rdf/3.0#">

     <rdf:Description rdf:about="http://www.csd.uoc.gr/~hy252">
          <!-- Title (1.2) -->
          <dc:title>
               <rdf:Alt rdf:ID="title">
                    <rdf:li xml:lang="en">CS252: Object Oriented
Programming</rdf:li>
               </rdf:Alt>
          </dc:title>

          <!-- Language (1.3) -->
          <dc:language>
               <dcterms:RFC1766>
                    <rdf:value>en</rdf:value>
               </dcterms:RFC1766>
          </dc:language>
          <dc:language>
               <dcterms:RFC1766>
                    <rdf:value>gr</rdf:value>
               </dcterms:RFC1766>
          </dc:language>

          <!-- Description (1.4) -->
          <dc:description>
               <rdf:Alt rdf:ID="desc">
                    <rdf:li xml:lang="en">This course analyzes the
main principles and characteristics of Object Oriented
Programming</rdf:li>
                    <rdf:li xml:lang="gr">Το μάθημα αυτό αναλύει τις
βασικές αρχές και ιδιότητες του Οντοκεντρικού Προγραμματισμού</rdf:li>
               </rdf:Alt>
          </dc:description>

          <!-- Keyword (1.5) -->
          <dc:subject>
               <rdf:Alt rdf:ID="keyword">
```

```
                        <rdf:li xml:lang="en">Object Oriented
Programming</rdf:li>
                        <rdf:li xml:lang="gr">Οντοκεντρικός
Προγραμματισμός</rdf:li>
                </rdf:Alt>
        </dc:subject>

        <!-- Coverage (1.6) ??? -->
        <dcterms:temporal>
                <dcterms:Period>
                        <rdf:value>start=2002-03-01;end=2002-05-
31;</rdf:value>
                </dcterms:Period>
        </dcterms:temporal>
        <dcterms:spatial>
                <rdf:Alt>
                        <rdf:li xml:lang="en">University of Crete,
Greece</rdf:li>
                        <rdf:li xml:lang="gr">Πανεπιστήμιο Κρήτης,
Ελλάδα</rdf:li>
                </rdf:Alt>
        </dcterms:spatial>

        <!-- Structure (1.7) -->
        <lom-gen:structure rdf:resource="&lom-gen;Linear"/>

        <!-- Aggregation Level (1.8) -->
        <lom-gen:aggregationLevel rdf:resource="&lom-
gen;AggregationLevel3"/>

        <!-- Life Cycle (2) -->
        <!-- Version (2.1) -->
        <lom-life:version>
                <rdf:Alt>
                        <rdf:li xml:lang="en">LOM Final Draft v1.0
Schema</rdf:li>
                </rdf:Alt>
        </lom-life:version>

        <!-- Status (2.2) -->
        <lom-life:status rdf:resource="&lom-life;Final"/>

        <!-- Contribute (2.3) -->
        <dc:creator>
                <lom:Entity>
                        <vCard:FN>Miltos Stratakis</vCard:FN>
                </lom:Entity>
        </dc:creator>
        <dcterms:created>
                <dcterms:W3CDTF>
                        <rdf:value>2000-02-10</rdf:value>
                </dcterms:W3CDTF>
        </dcterms:created>

        <!-- Technical (4) -->
        <!-- Format (4.1) -->
        <dc:format>
```

```
            <dcterms:IMT>
                    <rdf:value>text/html</rdf:value>
            </dcterms:IMT>
      </dc:format>

      <dc:format>
            <dcterms:IMT>
                    <rdf:value>application/pdf</rdf:value>
            </dcterms:IMT>
      </dc:format>

      <!-- Location (4.3) -->
      <lom-tech:location
rdf:resource="http://www.csd.uoc.gr/~hy252"/>

      <!-- Requirements (4.4) ??? -->
      <lom-tech:browser>
            <rdf:Alt>
                    <rdf:li>
                            <lom-tech:NetscapeCommunicator/>
                    </rdf:li>
                    <rdf:li>
                            <lom-tech:MicrosoftInternetExplorer/>
                    </rdf:li>
            </rdf:Alt>
      </lom-tech:browser>

      <!-- Educational (5) -->
      <!-- Interactivity Type (5.1) -->
      <lom-edu:interactivityType rdf:resource="&lom-
edu;Expositive"/>

      <!-- Learning Resource Type (5.2) -->
      <rdf:type rdf:resource="&lom-edu;Index"/>

      <!-- Interactivity Level (5.3) -->
      <lom-edu:interactivityLevel rdf:resource="&lom-
edu;MediumInteractivity"/>

      <!-- Semantic Density (5.4) -->
      <lom-edu:semanticDensity rdf:resource="&lom-
edu;HighDensity"/>

      <!-- Intended End User Role (5.5) -->
      <lom-edu:intendedEndUserRole rdf:resource="&lom-
edu;Learner"/>

      <!-- Context (5.6) -->
      <lom-edu:context rdf:resource="&lom-edu;HigherEducation"/>

      <!-- Typical Age Range (5.7) ??? -->
      <dcterms:audience>Suitable for university
learners</dcterms:audience>

      <!-- Difficulty (5.8) -->
      <lom-edu:difficulty rdf:resource="&lom-
edu;MediumDifficult"/>
```

```
            <!-- Typical Learning Time (5.9) -->
            <lom-edu:typicalLearningTime>
                    <lom:ISO8601>
                            <rdf:value>P4M</rdf:value>
                    </lom:ISO8601>
            </lom-edu:typicalLearningTime>

            <!-- Language (5.11) -->
            <lom-edu:language>
                    <dcterms:RFC1766>
                            <rdf:value>en</rdf:value>
                    </dcterms:RFC1766>
            </lom-edu:language>

            <lom-edu:language>
                    <dcterms:RFC1766>
                            <rdf:value>gr</rdf:value>
                    </dcterms:RFC1766>
            </lom-edu:language>

            <!-- Rights (6) -->
            <!-- Cost (6.1) -->
            <lom-rights:cost rdf:resource="&lom-rights;NoCost"/>

            <!-- Copyright and Other Restrictions (6.2) -->
            <lom-rights:copyrightAndOtherRestrictions
rdf:resource="&lom-rights;NoRestriction"/>

            <!-- Relation (7) -->
            <dcterms:hasPart
rdf:resource="http://www.csd.uoc.gr/~hy252/lectures/CS252Basics03.pdf">
                    <dc:description>
                            <rdf:Alt rdf:ID="desc">
                                    <rdf:li xml:lang="en">Java Programming
Basics is an initiative lesson of CS252 course</rdf:li>
                                    <rdf:li xml:lang="gr">Τα βασικά
χαρακτηριστικά του προγραμματισμού σε Java είναι εισαγωγικό μάθημα στον
Οντοκεντρικό Προγραμματισμό</rdf:li>
                            </rdf:Alt>
                    </dc:description>
            </dcterms:hasPart>
            <dcterms:requires
rdf:resource="http://www.csd.uoc.gr/~hy150">
                    <dc:description>
                            <rdf:Alt rdf:ID="desc">
                                    <rdf:li xml:lang="en">CS150 is a
prerequisite course of CS252</rdf:li>
                                    <rdf:li xml:lang="gr">Το ΗΥ150 είναι
προαπαιτούμενο του Οντοκεντρικού Προγραμματισμού</rdf:li>
                            </rdf:Alt>
                    </dc:description>
            </dcterms:requires>

            <!-- Classification (9) -->
            <lom-cls:accessibilityRestrictions
rdf:resource="http://www.myVocabulary.com/GreekUniversities#UOC.CS252/>
```

```
        </rdf:Description>

        <!-- Meta-metadata (3) ??? -->
        <rdf:Description rdf:about = "http://www.ieee.org/metadata.rdf">
              <!-- Contribute (3.2) -->
              <dc:creator>
                    <lom:Entity>
                          <vCard:FN>Miltos Stratakis</vCard:FN>
                    </lom:Entity>
              </dc:creator>
              <dcterms:created>
                    <dcterms:W3CDTF>
                          <rdf:value>2002-11-18</rdf:value>
                    </dcterms:W3CDTF>
              </dcterms:created>

              <!-- Metadata Schema (3.3) -->
              <lom-meta:metadataScheme rdf:resource="&lom-meta;LOMv1.0"/>

              <!-- Language (3.4) -->
              <dc:language>
                    <dcterms:RFC1766>
                          <rdf:value>en</rdf:value>
                    </dcterms:RFC1766>
              </dc:language>
        </rdf:Description>
</rdf:RDF>
```