ECP 2005 EDU 038098

MACE

D3-6.1 Metadata taxonomy and their integration in MACE

Deliverables number	*D-3.1, D-4.1, D-5.1, D-6.1* *(according to deliverables list)*
Dissemination level	*Public*
Delivery date	*31 May 2007*
Status	*Final*
Author(s)	*Stefan Apelt, Massimiliano Condotta Markus Eisenhauer, Roland Klemke, Herman Neukermans, Christian Prause, Marius Scholten, Marcus Specht, Martin Wolpers*

e*Content*plus

This project is funded under the *e*Content*plus* programme[1],
a multiannual Community programme to make digital content in Europe more accessible, usable and exploitable.

[1] OJ L 79, 24.3.2005, p. 1.

1 Introduction

1.1 Purpose of this document

This document introduces and describes taxonomies for each metadata type in MACE: Usage and social metadata, context metadata, learning process and design metadata and content and domain metadata. Listed in the MACE Description of Work as separate deliverables, they are combined into this document, with each work package forming one of the main chapters.

The purpose of this document is to provide a basis for further project activities by defining candidate metadata attributes that will be used in the enrichment process. This happened in cooperation with requirements analysis activities in work package 2. While the results from work package 2 – described in the deliverables D2.1, D2.2 & D2.3 – define metadata from the user point of view, this deliverable shows what is technically possible. Further efforts on integration and development of the ideas described here will lead to a prototype model of the MACE infrastructure.

1.2 Goals

Making digital contents accessible and enriching digital contents are the core goals in MACE. This is accomplished by enriching learning objects with metadata and thus linking learning objects to certain properties. Using these properties e.g. in search requests to discover learning objects postulates a well-defined classification of metadata. The association of metadata with certain data types allows for proper interpretation of the data and the derivation of new information. Improving the discovery of learning objects under qualitative rather than under quantitative aspects implies the need for personalised search methods. To provide MACE users with personalised information we not only collect metadata associated to learning objects but also enrich so-called MACE-entities, which include learning objects, MACE users and real-world objects, with metadata. By that, a user’s search request can be executed under consideration of his/her personal data. For example, a user’s role as well as certain preferences can influence a search result. Associating competencies with users is another important aspect. Assigning competencies to users in a proper way ensures the quality of user generated information. We address user competencies in detail in this document.

This document is a joint deliverable of the work packages 3 to 6. Notwithstanding the DOW we changed the order of chapters to a more adequate cycle for this document: We start by introducing content and domain metadata (WP6), continue with contextual metadata (WP4), and competence and process metadata (WP5) and end with usage related / social metadata (WP3) each in separate chapters. We define taxonomies for all metadata types, and explain how metadata is gathered and stored with existing learning object metadata. Further, we introduce ideas, how to improve search results by linking learning objects based on interrelations of metadata.

2 Taxonomy of content and domain metadata

2.1 Introduction

Content and domain metadata are provided in several formats using various metadata schemas. We have developed an application profile in order to harmonize the metadata descriptions and unlock the repositories in MACE. The work presented in this section results from the integration of available metadata provided from the MACE providers.

All providers agree to use the presented LOM based application profile. Nevertheless, the MACE consortium understands that the application profile might change during the project. Changes will be caused due to the integration and harmonization with the application profiles for the other kinds of MACE metadata, e.g. contextual and usage metadata. Furthermore, changes will be caused due to the requirements engineering process that is carried out in WP2 and will continue throughout the whole project. As changes to the application profile will occur, it is necessary to ensure the security of harvested and enriched metadata against faulty edits and changes. The respective harvesting infrastructure is set up to ensure a maximum of security in respect to changing and enriching metadata as well as developing application profiles.

The work described here has been or will be published in parts in:

> Herman Neuckermans, Martin Wolpers, Mathias Casaer & Ann Heylighen, Data and Metadata in Architectural Repositories, in: Yu Gang, Zhou Qi, Dong Wei (eds.) Digitization and Globalization, CAADRIA 2007, Proceedings of the 12th International Conference on Computer-Aided Architectural Design Research in ASIA, CAADRIA, Nanjing(China), pp.489-497
>
> Moritz Stefaner, Elisa Dalla Vecchia, Massimiliano Condotta, Martin Wolpers, Marcus Specht, Stefan Apelt, Erik Duval, MACE – enriching architectural learning objects for experience multiplication. Proceedings of the 2nd International Conference on Technology Enhanced Learning, Crete, Greece, September 2007

2.2 Components of the content and domain metadata schema

The content and domain metadata schema is based on the analyses of the learning object providers. The three providers DYNAMO, WINDS and ICONDA provided the mapping of their contents into LOM v1.0 as shown in the table below. The sum of all available metadata is the MACE application profile for content and domain metadata.

Said table specifies the MACE application profile. It is based on the Learning Object Metadata standard (LOM) v1.0 [Duval, 2005] with adaptations and extensions where necessary. The first and second rows give the number and name of the metadata field in accordance to LOM. The following four rows (DYNAMO Project, DYNAMO File, WINDS and ICONDA) outline which provider provides values for the respective metadata fields. The DYNAMO provider is split according to their structure that bases on projects and related media files describing the projects. DYNAMO projects are learning objects in their own right, therefore have their own LOM-based metadata description.

The Explanation row outlines the function of each metadata field, while the value space row outlines the possible values that the respective metadata field can hold. Furthermore, several extensions of LOM as well as the usage of LOM loose for several metadata fields are explained (number and name of these fields are light grey.) These adaptations and extensions are necessary to accommodate the requirements of architecture specific metadata in LOM.

	LOM v.1.0	**Dynamo Pr.**	**Dynamo File**	**Winds**	**Iconda**	**Explanation**	**Value Space**
1	**general**					This category groups the general information that describes this learning object as a whole.	See LOM v1.0
1.1	identifier	x	x	x	x	A globally unique label that identifies this learning object.	See LOM v1.0
1.1.1	catalogue	x	x	x	x	The name or designator of the identification or cataloguing scheme for this entry. A namespace scheme.	See LOM v1.0
1.1.2	entry	x	x	x	x	The value of the identifier within the identification or cataloguing scheme that designates or identifies this learning object. A namespace specific string.	See LOM v1.0
1.2	title	x	x	x	x	Name given to this learning object.	See LOM v1.0
1.3	language	x	x	x	x	The primary human language or languages used within this learning object to communicate to the intended user.	See LOM v1.0

	LOM v.1.0	Dynamo Pr.	Dynamo File	Winds	Iconda	Explanation	Value Space
1.4	description	x	x	x	x	A textual description of the content of this learning object.	See LOM v1.0
1.5	keyword			x	x	A keyword or phrase describing the topic of this learning object.	See LOM v1.0
1.6	coverage	x	x	x	x	The time, culture, geography or region to which this learning object applies. The data is a human readable text. If the coverage is based on any taxonomy, a respective metadata record will be included in category 9.	See LOM v1.0
1.7	structure	x	x	x		Underlying organizational structure of this learning object.	See LOM v1.0
1.8	aggregation level		x	x		The functional granularity of this learning object.	See LOM v1.0
2	**lifecycle**					This category describes the history and current state of this learning object and those entities that have affected this learning object during ist evolution. This category is also used to describe the lifecycle of buildings.	Extended LOM v1.0 value spaces to enable the description of lifecycles of buildings (e.g. project, planning, building, altering, etc.)
2.1	version	x		x		The edition of this learning object.	See LOM v1.0
2.2	status	x	x	x		The completion status or condition of this learning object, extending the LOM value space with architecture specific values, e.g. conversion, extension	Extended LOM v1.0 value space to extend the vocabulary with architecture specific statuses.
2.3	contribute	x	x	x	x	Those entities (i.e., people, organizations) thathave contributed to the state of this learningobject during its life cycle (e.g., creation, edits,publication).	See LOM v1.0

	LOM v.1.0	Dynamo Pr.	Dynamo File	Winds	Iconda	Explanation	Value Space
2.3.1	role	x	x		x	Specifies the kind of contribution in respect to the learning object and extending the value space with architecture specific roles, e.g. architect, constructor, etc.	Extended LOM v1.0 value space to extend the vocabulary with architecture specific roles.
2.3.2	entity	x	x			The identification of and information about entities (i.e., people, organizations) contributing to this learning object. The entities shall be ordered as most relevant first. Use vCard format.	vCard
2.3.3	date	x	x	x	x	The date of the contribution.	See LOM v1.0
3	**meta-metadata**					This category describes how the metadata instance can be identified, who created this metadata instance, and how, when, and with what references.	See LOM v1.0
3.1	identifier	x	x	x	x	A globally unique label that identifies this metadata record.	See LOM v1.0
3.1.1	catalogue	x	x		x	The name or designator of the identification or cataloguing scheme for this entry. A namespace scheme.	See LOM v1.0
3.1.2	entry	x	x		x	The value of the identifier within the identification or cataloguing scheme that designates or identifies this metadata record. A namespace specific string.	See LOM v1.0
3.2	contribute	x	x	x	x	Those entities (i.e., people or organizations) that have affected the state of this metadata instance during its life cycle (e.g., creation, validation). At harvesting time, a vCard instance of the metadata provider organisation will be included.	See LOM v1.0
3.2.1	role	x	x			Kind of contribution.	Extended LOM v1.0 value space to include the role “provider”.

	LOM v.1.0	Dynamo Pr.	Dynamo File	Winds	Iconda	Explanation	Value Space
3.2.2	entity	x	x	x		The identification of and information about entities (i.e., people, organizations) contributing to this metadata instance. The entities shall be ordered as most relevant first. Use vCard format.	See LOM v1.0
3.2.3	date	x	x	x	x	The date of the contribution.	See LOM v1.0
3.3	metadata schema		x			The name and version of the authoritative specification used to create this metadata instance.	See LOM v1.0
3.4	language		x			Language of this metadata instance. This is the default language for all LangString values in this metadata instance. If a value for this data element is not present in a metadata instance, then there is no default language for LangString values.	See LOM v1.0
4	**technical**					This category describes the technical requirements and characteristics of this learning object. It is used to describe the technical properties of digital or building learning objects.	See LOM v1.0
4.1	format		x	x		Technical datatype(s) of (all the components of) this learning object.	See LOM v1.0
4.2	size		x			The size of the digital learning object in bytes (octets). The size is represented as a decimal value (radix 10). Consequently, only the digits “0” through “9” should be used. The unit is bytes, not Mbytes, GB, etc.	See LOM v1.0
4.3	location	x	x	x	x	A, set of metadata fields that is used to locate the object. It may be the location of a digital learning object (e.g. Uniform Resource Locator) or a dataset specifying the location of a real-world object. The exact structure of the metadata set will be defined in cooperation with wps 4 and 5.	Extended LOM v1.0 value spaces with URI, GPS, etc. to be specified by WP4 and WP5.

	LOM v.1.0	Dynamo Pr.	Dynamo File	Winds	Iconda	Explanation	Value Space
4.4	requirement		x		x	The technical capabilities necessary for using this learning object.	See LOM v1.0
4.4.1	OrComposite				x	Grouping of multiple requirements.	See LOM v1.0
4.4.1.1	type		x		x	The technology required to use this learning object, e.g., hardware, software, network, etc.	See LOM v1.0
4.4.1.2	name		x			Name of the required technology to use this learning object.	Extended LOM v1.0 value spaces to extend the vocabulary with architecture specific tools and applications.
4.5	Installation Remarks			x		Description of how to install this learning object.	See LOM v1.0
4.6	Other Platform Requirements		x	x		Information about other software and hardware requirements.	See LOM v1.0
4.7	duration			x		Time a continuous learning object takes when played at intended speed.	See LOM v1.0
4.8	climate	x				The climate in which the building is or will be situated.	Free form text.
4.9	topography	x				Description of the terrain in which the building is located.	Free form text.
4.10	site	x				The urban context of the building.	Free form text.
5	**educational**					This category describes the key educational or G38pedagogic characteristics of this learning object.	See LOM v1.0
5.1	interactivity type	x	x	x	x	Predominant mode of learning supported by this learning project.	See LOM v1.0
5.2	learning resource type	x	x	x		Defining the specific kind of learning object. The most dominant kind shall be first. Describes the educational kind of the learning object, e.g. if it is a building, a project, etc.	Extended LOM v1.0 value spaces to extend the LOM vocabulary with architecture specific types of learning resources.
5.3	interactivity level	x	x	x		The degree of interactivity characterizing this learning object. Interactivity in this context refers to the degree to which the learner can influence the aspect or	See LOM v1.0

	LOM v.1.0	Dynamo Pr.	Dynamo File	Winds	Iconda	Explanation	Value Space
						behavior of the learning object.	
5.4	semantic density			x		The degree of conciseness of a learning object. The semantic density of a learning object may be estimated in terms of its size, span, or—in the case of self-timed resources such as audio or video—duration.	See LOM v1.0
5.5	intended end user role	x	x	x	x	Principal user(s) for which this learning object was designed, most dominant first.	Extended LOM v1.0 value spaces to extend the LOM vocabulary with architecture specific types of users, e.g. architects, etc.
5.6	context	x	x	x		The principal environment within which the learning and use of this learning object is intended to take place.	See LOM v1.0
5.7	Typical Age Range			x		Age of the typical intended user.	See LOM v1.0
5.8	difficulty			x		How hard it is to work with or through this learning object for the typical intended target audience.	See LOM v1.0
5.9	Typical Learning Time			x		Approximate or typical time it takes to work with or through this learning object for the typical intended target audience.	See LOM v1.0
5.10	description	x	x	x		Comments on how this learning object is to be used.	See LOM v1.0
5.11	language	x	x	x	x	The human language used by the typical intended user of this learning object.	See LOM v1.0
6	**rights**					This category describes the intellectual property rights and conditions of use for this learning object.	See LOM v1.0
6.1	cost	x	x	x		Whether use of this learning object requires payment.	See LOM v1.0
6.2	copyright & other restrictions	x	x	x		Whether copyright or other restrictions apply to the use of this learning object.	See LOM v1.0

	LOM v.1.0	Dynamo Pr.	Dynamo File	Winds	Iconda	Explanation	Value Space
6.3	description	x	x	x		Whether copyright or other restrictions apply to the use of this learning object.	See LOM v1.0
7	**relation**					This category defines the relationship between this learning object and other learning objects, if any. This category enables the relation of LOs describing buildings to a LO that describes a project (e.g. in Dynamo.) For example, this relation can be used to express that a building uses a specific stone. In this case, LOs for the building and the stone must exist.	See LOM v1.0
7.1	kind	x	x		x	Nature of the relationship between this learning object and the target learning object, identified by 7.2:Relation.Resource.	Vocabulary based on Dublin Core but extended to enable additional relations like “shows”
7.2	resource		x		x	The target learning object that this relationship references.	See LOM v1.0
7.2.1	identifier	x	x			A globally unique label that identifies the target learning object.	See LOM v1.0
7.2.1.1	catalogue	x	x			The name or designator of the identification or cataloguing scheme for this entry. A namespace scheme.	See LOM v1.0
7.2.1.2	entry	x	x			The value of the identifier within the identification or cataloguing scheme that designates or identifies the target learning object. A namespace specific string.	See LOM v1.0
7.2.2	description		x		x	Description of the target learning object.	See LOM v1.0
9	**classification**					This category describes where this learning object falls within a particular classification system. This category is to be used if LOs are described through extensive classifications. For example, if the LO describes a wall, and the wall is classified within an extensive classification schema like the	See LOM v1.0

	LOM v.1.0	Dynamo Pr.	Dynamo File	Winds	Iconda	Explanation	Value Space
						CI/SFB or Getty catalogues, the respective reference into the catalogue for the LO needs to included in this category.	
9.1	purpose	x				The purpose of classifying this learning object.	Free form text.
9.2	taxon path	x				A taxonomic path in a specific classification system. Each succeeding level is a refinement in the definition of the preceding level.	See LOM v1.0
9.2.1	source	x				The name of the classification system. This data element may use any recognized “official” taxonomy or any user-defined taxonomy.	See LOM v1.0
9.2.2	taxon	x				A particular term within taxonomy. A taxon is a node that has a defined label or term. A taxon may also have an alphanumeric designation or identifier for standardized reference. Either or both the label and the entry may be used to designate a particular taxon.	See LOM v1.0
9.2.2.1	id	x				The identifier of the taxon, such as a number or letter combination provided by the source of the taxonomy.	See LOM v1.0
9.2.2.2	entry	x				The textual label of the taxon.	See LOM v1.0
9.3	description	x				Description of the learning object relative to the stated 9.1:Classification.	See LOM v1.0

2.3 Infrastructure

The MACE infrastructure strives to open up the existing Learning Object Repositories (LORs) to enable the access of Learning Objects (LOs) through MACE tools. Therefore, we rely on a hybrid combination of harvesting metadata from and federating searches to existing content repositories. Additionally, the infrastructure enables the enrichment of learning object descriptions with metadata about their usage including contexts of use, necessary competencies, etc. The approach aims to make the learning objects in all repositories jointly searchable and retrievable.

The technical infrastructure allows searching over the contents of all content repositories based on metadata. In order to enable “semantic interoperability” among LORs, the learning objects are described through the MACE application profile as shown in the previous section.

Existing metadata from the connected repositories is collected via metadata harvesting based on the Open Archive Initiative Protocol for Managing Harvesting OAI-PMH (OAI, 2002). Harvesting in this context means the transfer of the content metadata from the providing repository into the central content metadata repository on a regular basis. Note that only the metadata describing the learning objects is transferred; the learning objects themselves stay in the repository and thus in control of their owner without changing the access conditions. In turn, the central content metadata repository also offers an OAI-PMH interface so that interested content metadata providers can retrieve enriched metadata suitable for their learning objects.

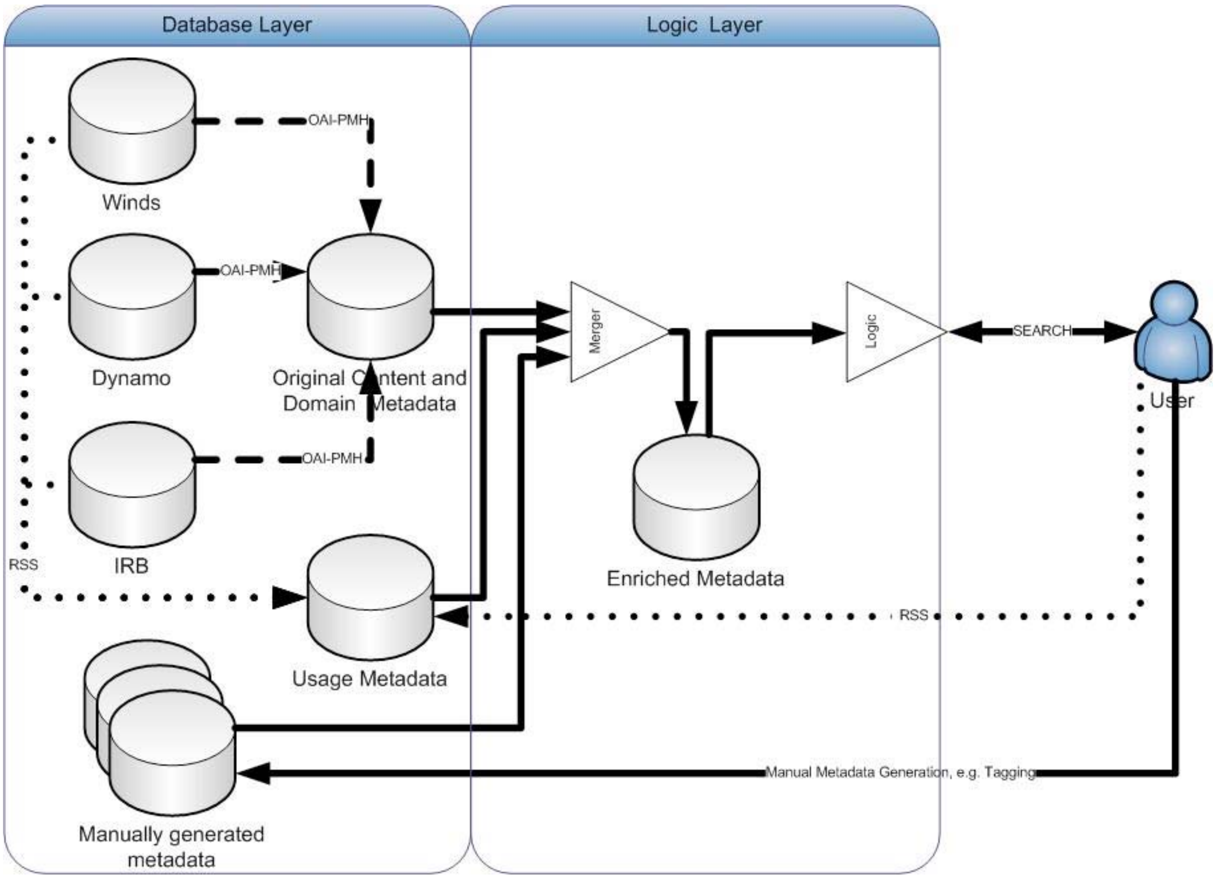

Figure 1: Harvesting metadata from MACE repositories

Figure 1 shows how harvesting works in MACE in principle: Existing metadata de-scribing the learning resources is harvested through the OAI-PMH protocol into the MACE central metadata store for original harvesting metadata. Within the database layer, OAI-PMH is used for harvesting content and domain metadata. The merge logic will combine the metadata originating from the provider with usage metadata of the learning objects (see wp3) and manually generated metadata and store it in the enriched metadata store. We have chosen this storage structure in order to be able to separate the original metadata from enriched metadata. At a later stage, this architecture will enable efficient and effective error recovery procedures, e.g. in the case of faulty enriched metadata.

The enriched metadata store supports a search facility that provides references to available and suitable learning objects. In order to access the learning object, the user accesses the learning resource directly at the provider through the respective mechanisms of each provider.

2.4 Extending LOM v1.0

The above content and domain metadata schema extends the original LOM v1.0 schema in order to be able to capture specific architectural properties. The main difference lies in the distinction of real-world and digital objects in architecture. The MACE consortium has the opinion that digital objects as well as the real-world objects like a building are learning objects. Thus, metadata sets for both types of learning objects have to be represented by the content and domain metadata application profile. In consequence, the LOM schema is modified to suite these needs, e.g. through the extension of the technical location metadata field that captures the physical location of a learning object, it may be accessible through an URL or physically at certain GPS coordinates.

Furthermore, the LOM specification has been extended to suit several contextual metadata that are provided by the MACE repositories. The metadata fields climate, topography and site (LOM category 4 technical) are specifically building oriented: climate metadata describe the climate of the area where the building stands, topography describes the geographical landscape (e.g. hill, mountain, etc.) while the site describes the urban environment within which the building is situated. They are included explicitly because their value space is a flat list of values. If the value space is more complex, e.g. a taxonomy, such metadata fields will be represented in LOM category 9 classification.

Figure 2 will outline the MACE modelling approach:

Figure 2: S. Pietro Basilica (recent)

The learning object is a picture of “S. Pietro Basilica”, the properties describing the content (the Basilica) are:

ITALY, ROME, (location), CITY CENTRE (urban context), PLAIN (geographical context), XVI CENTURY (historical context)…

Furthermore, the properties describing the media (the photo) are:

SEPTEMBER 2004 (the date of the photo), WIKIPEDIA (where the image come from), WOLFGANG STUCK (the author of the Photo)

To model the image of the basilica (Figure 2) in LOM, two LOM instances are necessary: One describes the Basilica and one describes the image that shows the basilica. (see Figure 4 in 19section 3.2 for further explanations on this principle.) The following metadata instances describe the basilica as well as the image of the basilica. Please note that the example is simplified to emphasize the differences to the original LOM v.1.0:

Basilica:

```
LOM.general.identifier = SPietroBasilica

LOM.general.title = S. Pietro Basilica

LOM.general.coverage = XVI Century


LOM.technical.location = Italy, Rome

LOM.technical.topography = plain
```

```
LOM.technical.site = City Center
```

…

Image 1 (Figure 2):

```
LOM.general.identifier = Image1SPietroBasilica

LOM.general.title = Image of S. Pietro Basilica


LOM.technical.location = WIKIPEDIA URL

LOM.technical.format = jpg


LOM.lifecycle.contribute.role = creator/author

LOM.lifecycle.contribute.entity = vCard of Wolfgang Stuck

LOM.lifecycle.contribute.date = September 2004


LOM.relation.kind = shows (lom loose definition)

LOM.relation.identifier.entry = SPietroBasilica (reference to Basilica LO)
```

…

Furthermore, if a second image exists, that shows the basilica in a different time, an additional metadata instance needs to be created:

Figure 3: S. Pietro Basilica (ancient)

The properties of Figure 3 that are referring to the content (the Basilica) are the same. Most of the properties referring to the media (the painting) are different:

1630 (the date of the painting), RENDITION OF ST. PETER'S SQUARE (the name of the painting), VIVIANO CODAZZI (the author of the paint)

Consequently, the respective metadata set is:

Image 2 (Figure 3):

```
LOM.general.identifier = Image2SPietroBasilica
LOM.general.title = Rendition of St. Peter’s Square

LOM.technical.format = jpg

LOM.lifecycle.contribute.role = creator/author
LOM.lifecycle.contribute.entity = vCard of Viviano Codazzi
LOM.lifecycle.contribute.date = 1630

LOM.relation.kind = shows (lom loose definition)
LOM.rleation.identifier.entry = SPietroBasilica (reference to
the Basilica LO)
```

…

Using this approach, it becomes much simpler to distinguish between the digital and real-world objects. Describing all learning objects in LOM provides a number of benefits: (1) The computational handling of LOs is simplified because no translation between metadata schemas is necessary. (2) Real-world objects like buildings are learning objects and therefore might be used in advanced learning scenarios as identified in WP2. (3) Representing real-world objects as well as learning objects supports and eventual simplifies the concise description and usage of architectural context.

3 Taxonomy of context metadata

3.1 Introduction

Over the last years, context has received a great amount of attention in various areas of research. A great variety of definitions and understandings of context exist. In linguistics, people research the context dependent meaning of utterances (see Akman 1999) or the effect that dialogues have on changing context (e.g. Bunt 1994), where context is seen as the personal context of participants in communication.

Some philosophers state that there is no context independent meaning of information at all (Heidegger 1962). Others (Penco 1999) distinguish between metaphysical context (= set of features of the world) and cognitive context (= set of assumptions on the world).

Cognitive scientists research the notion of context in information systems (e.g. Croon 1998, Nardi 1996b) stating that a contextual understanding of information systems implies that there is no clear border between systems and their social surroundings. Other research focuses on the development of experts in certain areas, resulting in the observation, that expertise results from intensive practice combined in context, as opposed to the previous belief in innate talent (see Ericsson & Charness 1997). However, within the cognitive science research community, the meaning and impact of context is not yet agreed on (compare e.g. Ziemke 1997 for a comparison of the meaning of context in cognitivism and enaction).

These different notions of context underlying the different approaches clearly motivate one important insight: what we consider to be context depends on what we want to contextualise. This means, that before context can be represented, the viewpoint from which we look at context has to be known (Klemke 1999; 2000).

In WP4, we want to enrich contents with contextual information to enable users of MACE to use it for searches and recommendations when interacting with the system.

The clear aim of the context taxonomy is to define the context related aspects of the overall taxonomy to be used in MACE, the corresponding metadata schema and its relation to LOM. Contextual metadata will provide a categorisation of entities with respect to similarities in their context metadata and enable a more advanced search than traditional keyword search can offer. The first approach to achieve this goal is to define what “context” means in MACE and to operationalize it. As stated above we will describe the schema and taxonomy to be used for contextual metadata and define which aspects of context will be used and how we are going to obtain them.

3.2 Context in MACE

Generally, context can be defined as "*the conditions and circumstances that are relevant to an event, fact, etc.*" (Collins 1999) or "*the interrelated circumstances in which something exists or occurs*" (Webster's 1996).

The term "context-aware" was first used by Schilit and Theimer to describe mobile computing applications (Schilit and Theimer 1994). Dey provides two quite general definitions that fit with the intention within MACE:

> "*Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.*"
>
> *"A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task.*" (Dey 2001)

In MACE, entities fall into these categories:

- Real world subjects and objects with a relation to architecture (e.g. architects, buildings, places...)
- users of MACE
- digital contents describing either subjects/objects or users (e.g. text, images, multimedia {audio, video, animations})

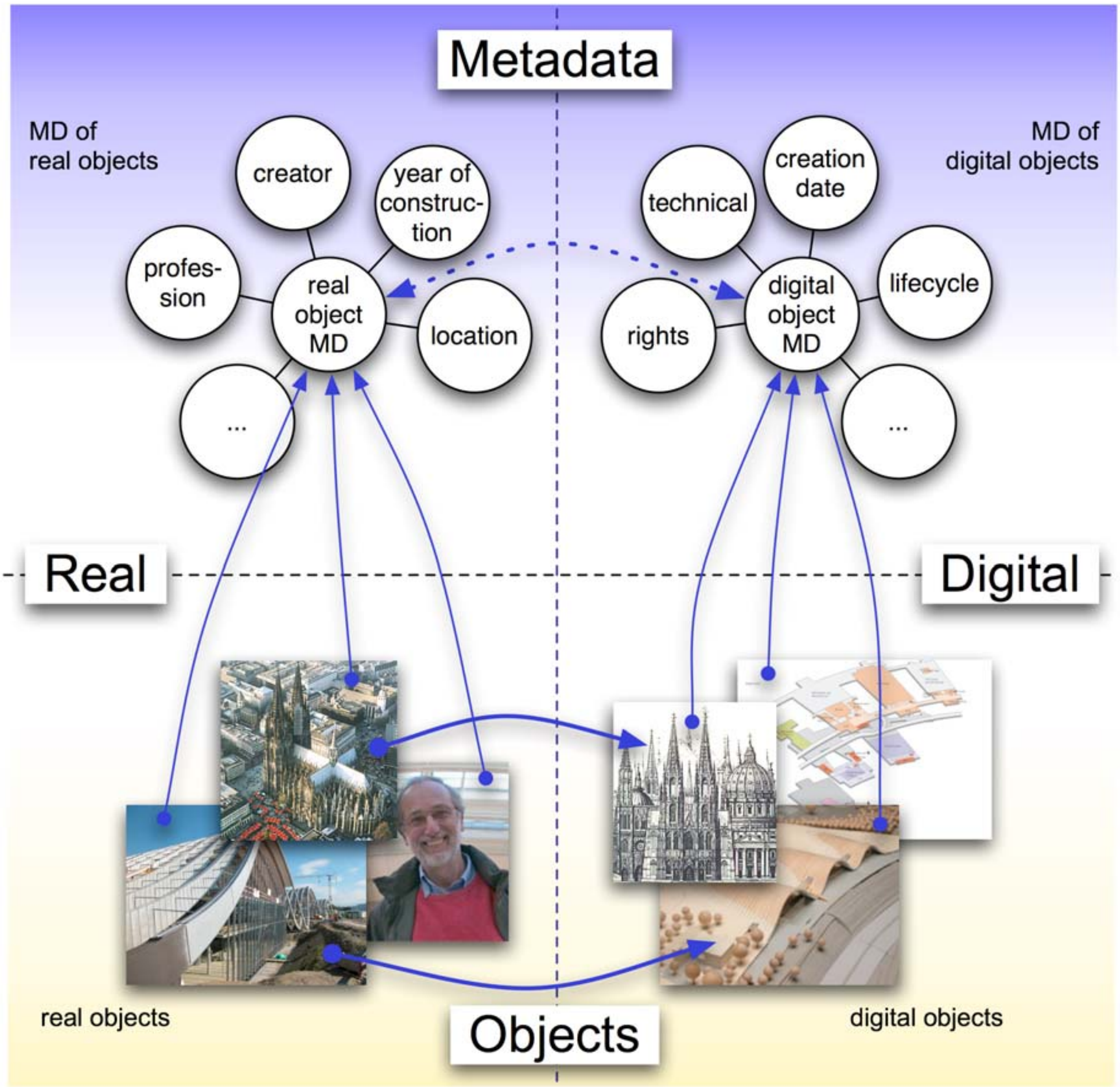

Figure 4 - Real world objects and digital contents

Even though the MACE system will deal with digital contents describing real world objects and not the objects themselves, it makes sense to distinguish between these two categories because they have different metadata associated with them. Consider for example the “Paul Klee Zentrum” in Bern and a photo of it (see Figure 4): the metadata “creation date” associated with the museum may be different from the “creation date” of the photo. Fortunately, the LOM standard (IEEE-LTSC 2002) allows for more than one metadata record per content object. We will make use of that by having different LOM records linked to each other, one for the real world object and one for the digital content:

- For each building or project there is one LOM description that includes the metadata about the building, e.g. its location, the architect, etc. If necessary, there might be several LOM descriptions about the same building, each describing a different state of the building.

For each digital object (e.g. photo, image, movie, drawing, course, etc.) there exists one LOM description that includes a reference to the building shown, the author of the digital object, etc.

3.3 Context types in MACE

Petrelli, Not et al. mention that "…context is by definition unstable and changing. A given configuration of context is valid only at a given time." (Petrelli, Not et al. 2001) We have identified the following types of context, which play a vital role in MACE: Architectural, physical, social, usage, role and technical context. However, not all types are equally important and relevant for each entity:

		Context types					
		Architectural	Physical	Social	Usage	Role	Technical
Entity	Users		x	x	x	x	x
	Contents				x		x
	Real world objects	X	x	x	x		x

Table 1 - Context and entity types

The second table lists enrichment methods for each context-entity combination. The symbols mean the following:

T – Gather **T**racking information about entity

D – **D**erive metadata from other existing Databases

M – **M**anual metadata entry

		Context types					
		Architectural	Physical	Social	Usage	Role	Technical
Entity	Users		T	T,M	T	M	M,T
	Contents				T		M
	Real world objects	D,M	D,M	T	T		M

Table 2 - Enrichment methods

In WP4, we cannot deal with all these types of context. Fortunately, some of these are covered by other work packages. First, we will list and describe each kind and then we explain which ones we will use in WP4 – Contextual Metadata.

3.3.1 Architectural context

Architectural context does not pertain to media, which is the form of the content in semiologic terminology. Architectural context applies to real world objects such as buildings and places. The architectural context of a building depends on the environment of the building. It influences the characteristics of buildings like shape and material. In general, there are two main architectural context types(see Figure 5): one refers to the physical world and its characteristics like location, environment, geographical etc., thus elements directly influencing construction by means of necessity and availability (the city, the desert, the country, etc) and the other one refers to intellectual aspects (i.e. culture).

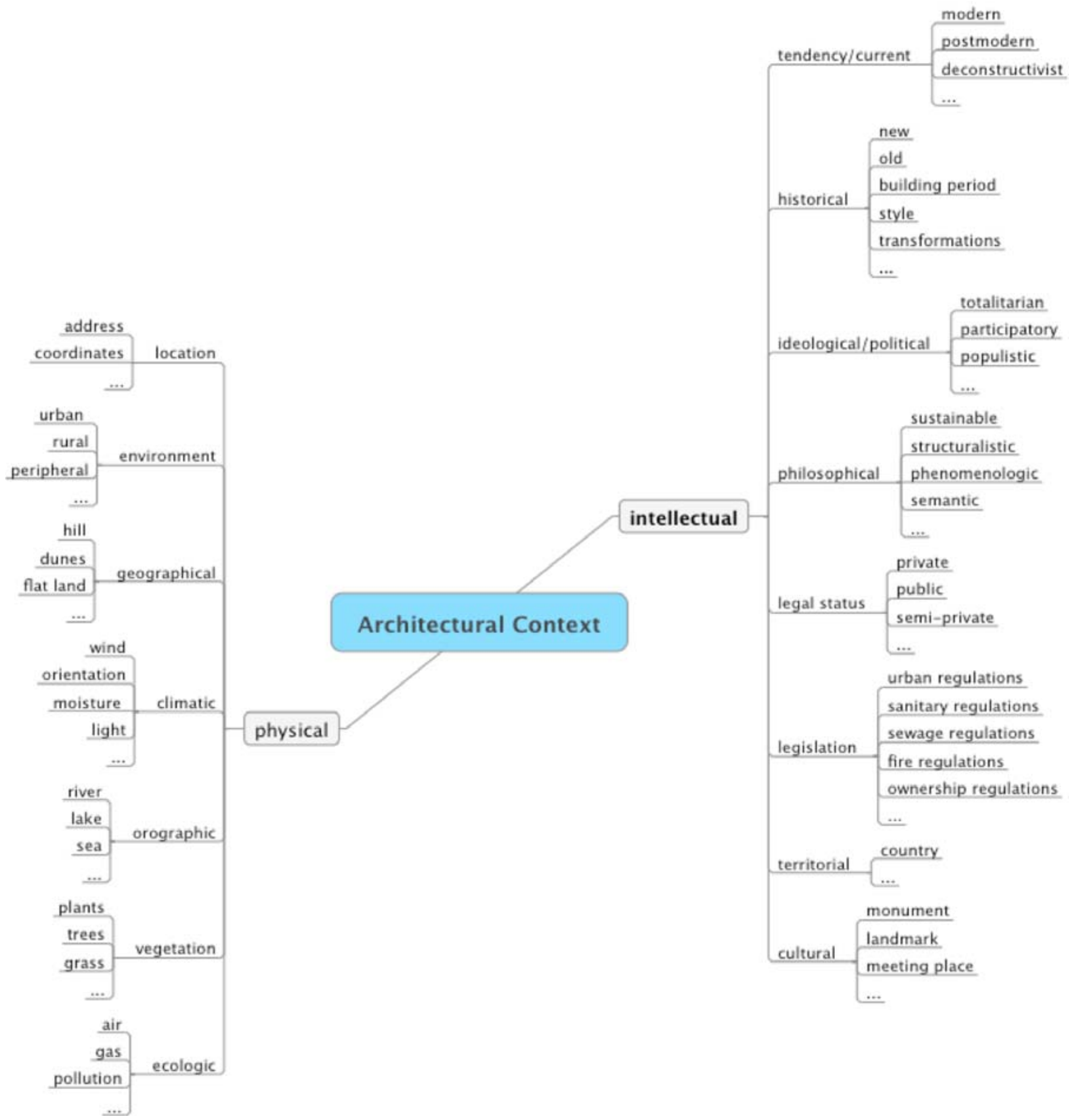

Figure 5: First taxonomy of architectural context

On top of this first taxonomy architectural context can be grouped into three categories:

- Geographic context belongs to the real world context and describes the surrounding of a building in terms of environment conditions. For architects it is interesting to know what these conditions are in order to plan a construction site and ensure protection against the elements.
- Urban context is formed by joining real and cultural environments. It describes a building from the view of town planning and town development as well as the historical setting of a city (modern, ancient city, etc.). Rural and city centre buildings might have different needs (public transport, security, building style, etc).
- Historical and cultural context describes the characteristics of the cultural environment linked to the territory where the building is located. It can be used for example to find information about buildings with similar functions in a different time or to find different buildings located in similar cultural context but in different places (i.e. different cities or countries).

Enriching contents with architectural context is in some cases possible with automated methods. For example, seismic, geological and orographic metadata can be retrieved from specialized databases that record earth activity and provide risk assessment scenarios for a given area.

In other cases, such as for the historical, territorial or urban profiles, manual entry of metadata or manual connection of records in databases will be necessary.

Architectural context for a real world object will be stored as LOM domain metadata record together with digital contents describing the real world object.

3.3.2 Physical context

This type of context is by far the most researched on. In the beginning, physical attributes like time and location were used as synonyms for context (Schilit, Adams et al. 1994).

Physical context contains information about real world objects. Attributes of this context can be time, location and weather conditions. For example, it might be interesting to know whether it rained or snowed when a photo was taken. Other attributes include sensory information such as light and noise levels like “glass façade at day / by night” or “noise level with / without insulating material”.

In MACE we want to enrich contents with physical context metadata to allow for targeted searches based on location and time. Physical context metadata can be gathered automatically by using sensors for tracking user locations. For real world objects specialized databases containing position information can be connected to MACE. Physical context for real world objects will be stored as LOM domain metadata records together with digital contents about the real world object. Physical context of users will be stored in a user profile database.

3.3.3 Social, usage and role context

Usage and social context closely belong together, but are not the same. The usage context of an entity contains information about the number of activities that belong to this entity. Entities can be real world objects, a content object or a user. Usage information could be “visits to it”, “number of views” or “number of searches” respectively. This can be used to determine interesting entities with a high usage count (e.g. “hot topics”, “most active users”, “most frequented places”) and to look for similarities (“other people seeing this content also have seen the following contents”). Usage context will play an important role in MACE-WP3.

Social context is very important in learning scenarios, e.g. looking for peers to ask for feedback or help. (Braun and Schmidt 2007) It builds on usage information, but also includes information about relationships between users. For a student, this can be peers, tutors, independent experts and teachers. Social context also allows users to rate others and their competencies, something that will be done in MACE-WP5.

Most IT systems provide the same behaviour to each user. By taking into account the role of a user, we can create a system that delivers information specifically targeted to certain types of users, depending on their role. A student searching for “glass facades” might be interested in the general concept of applying glass to facades, while a professional user might be interested in standards and part numbers.

By combining social, usage and role context, we will to provide better search results and increase user satisfaction.

Usage context information can be gathered automatically by means of analysing log files and using sensors. Social context information depends on user input (relationships and rating). It might also be partly inferred from user actions. The role context can either be specified by the user or it results from a given scenario.

Information for these contexts will either be stored in user profiles (role, personal interests, and traits) or be derived from existing databases via queries and aggregating the results.

3.3.4 Technical context

Although not directly relevant for the domain of architecture, this type of context is necessary for the operation of the technical infrastructure. It contains metadata on how to process information depending on content types, end user devices and network capabilities available. For example, a user with a mobile phone or PDA will have trouble to use a high-resolution video because of network bandwidth and device capabilities. Or an image in a proprietary format might not be displayable at the user's screen. To avoid these issues, technical metadata have to be applied to contents (formats, file size, etc.) and end user devices.

Technical context information can be gathered mostly automatically from the content or device itself. In some cases it might be necessary to manually add metadata. Technical context will be stored in LOM records and it either directly describes the contents or it will capture IT-technical conditions at real world objects. Technical context for users will be read from profiles of the user or the user's device.

3.3.5 Summary

We have described the types of context that could be interesting to MACE users. We will not deal directly with the educational aspects (social, role and usage) in WP4; they are subject of work packages WP3 (usage related metadata) and WP5 (learning process and learning design metadata), respectively. Architectural context concerning manual data input, will be a subject of work package WP6 (content and domain metadata). Also, most of the technical context can already be captured with existing LOM metadata structures and thus will be subject of WP6.

In work package WP4, our main subject is physical context, that is attributes of the physical surrounding of real world objects or users. We will contribute to architectural context and technical context where we can supply automatically generated metadata.

3.4 Creation and storage of context

Context information already exists in freely accessible databases on the Internet but is not yet connected to the MACE learning objects. However, if we enrich these learning objects with e.g. physical context, we can enable new search queries, that did not yield results before like “find buildings from same architect in the same city”. Our goal therefore is to connect learning objects with context data from external databases.

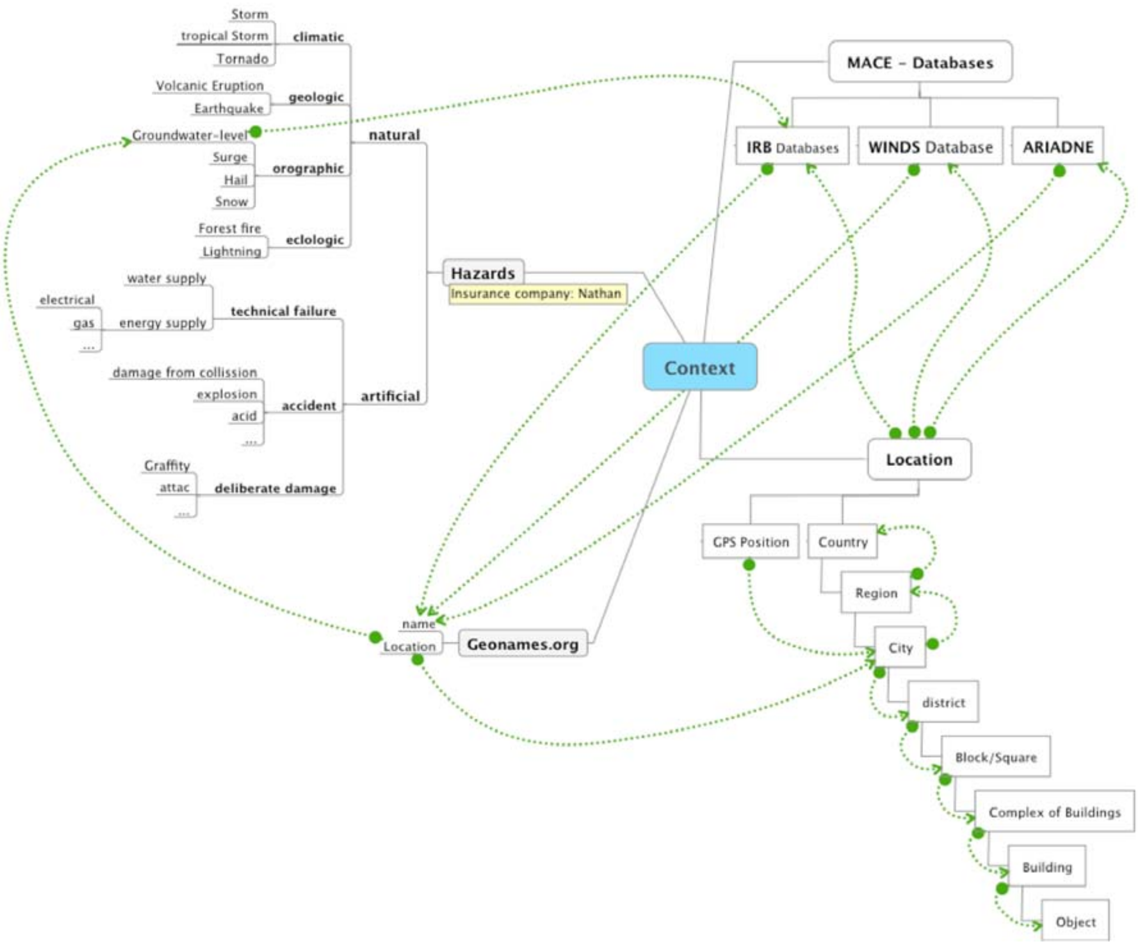

Figure 6: Context Taxonomies and their relation to Mace Databases

Figure 6 defines the context taxonomy to be used in MACE and the interrelations to the respective digital objects within the MACE databases. In WP 4 we will use databases like geonames.org to automatically enrich the location for objects within MACE. Databases of hazards from insurance companies like gfz-potsdam.de or NATHAN[1] (NATural Hazards Assessment Network; see http://mrnathan.munichre.com/) allow access to risk classifications for regions. This data could in turn be used to automatically tag digital objects within

[1] The access to NATHAN, though is still not clear. As this site is owned by the Münchner Rück it depends on their cooperation if we can use it within MACE. We are in contact with them and they need more information for a final decision.

MACE. For example the connection to geonames.org enables the automatic resolution of the location at the granularity of a city. The location taxonomy allows placing queries on the same granularity level or on a more abstract or more detailed level. So a search query for a learning object could automatically be enhanced to deliver as well results based on the location for other learning objects on the same level (e.g. city of Cologne) or on a more detailed (e.g. districts in Cologne) or a more abstract (e.g. in North-Rhine-Westphalia) level. The hazards databases could be used as well to automatically tag learning objects with contextual information like climatic, geologic or ecological context. As explained above, this would in turn lead to new intelligent context-based search queries, based e.g. on learning objects with specific ecological or climatic requirements. It is obvious that the intelligent use of the context taxonomy will lead to an enhanced metadata set of learning objects, that will allow new associations of digital content in architecture, that was not possible before.

As stated in the paragraph above and in Figure 6 we have already identified the following potential candidate databases:

- The geonames.org database contains a huge number of entries (cities, buildings, administrative divisions) together with their GPS location. We can link these entries with full text descriptions from learning objects to make the learning objects location aware
- Hazard databases like gfz-potsdam.de or mrnathan.munichre.com contain information about architectural geographic context. We can link these to e-learning project descriptions.

To store the context information, we extend the LOM schema for MACE with a new category “context” that contains keys like “context.physical.location”.

Type of context	Attribute	Description	Type	Examples
Architectural		Includes aspects related to site from territorial point of view, urban area analysis, historical relevance, existent building and all the relevant geographical factors.		
	urban	Project's area features: infrastructures; productive and residential areas; services	text, enumeration	"rural", "city"
	cultural, historical context and	Historical analysis of urban transformation and	free text	

Type of context	Attribute	Description	Type	Examples
	references	stratification. Specification of historical philosophical, historical-geographical, School, Author and Cultural trends or precedent project cases that have been used as references for a given design project		
	geographical	It specifies the climatic, geomorphologic and geological characteristics of the context, and the presence of natural/geographic emergencies in the area	<natural landscapes> Repertoire of AAT Getty Vocabulary	
Physical		Includes aspects of the physical environment.		
	date/time	The date/time of a record or event.	date/time	"2007-02-28 14:37:12"
	location	The location of a real world building. The location where a content was created. The location of a user.	GPS position	"7.124W, 51.102N"
	weather	Weather conditions at time of content creation.	enumeration	"rain", "snow", "sun"
	light level	Light level at time of content creation.	enumeration	"dawn", "low light"
	noise level	Noise level (for multimedia) at content creation.	enumeration	"quiet", "noisy"
Technical		Includes technical aspects needed by the technical infrastructure.		
	bandwidth	Bandwidth of network connection available.	text	"DSL", "11mbit"
	media format	Media format of a content object	text	"video/mpeg"
	file size	File size of a content object	number	3.234.123
	resolution.width	For a image or video content, its media resolution. For a user device, its display capabilities.	number	400
	resolution.height	For a image or video content, its media resolution. For a user device, its display capabilities.	number	300

Table 3 - Context attributes

Having these attributes at hand, we can link external databases to our learning object repositories, enrich the learning objects with new metadata and incorporate new data into MACE. To do this, we will create filters that map to the API of the respective external database and allow queries to that database. Thus the above attributes can be seen as enablers for connections. Digital objects within MACE databases can be linked with external databases via a match of tags. Filters represent search views on digital objects and select a view on the data according to a specific selection and weighting of tags. Chapter 3.5 will reveal more details on the use of context.

3.4.1 Cost of context creation

We want to maximize our impact of contextual enrichment, which grows with the amount of enriched learning objects. As already implied before, there are three possible ways of content enrichment: automatic, semi-automatic and manual:

- automatic: metadata is attached automatically, no user interaction is required; this is the fastest and most cost-effective way
- semi-automatic: metadata is also attached automatically, but user interaction is required for sanity checks or additional input
- manual: metadata cannot be generated automatically (e.g. expert knowledge) and must be entered manually; this is the slowest and (based on the same amount of metadata) most expensive way

We will try to gather as much metadata as possible in automated ways. Technical metadata and most of the physical metadata can be gathered automatically. For architectural context, a semi-automated method is possible where experts approve of found data (so they don’t have to create it by hand) and link it to contents manually.

3.5 Use of context

The context information gathered and attached to Learning Objects can be used in a variety of ways (Figure 7). It is interesting to note that end users will very seldom have to modify context directly, usually this will be done by the MACE infrastructure in accordance with the user’s actions.

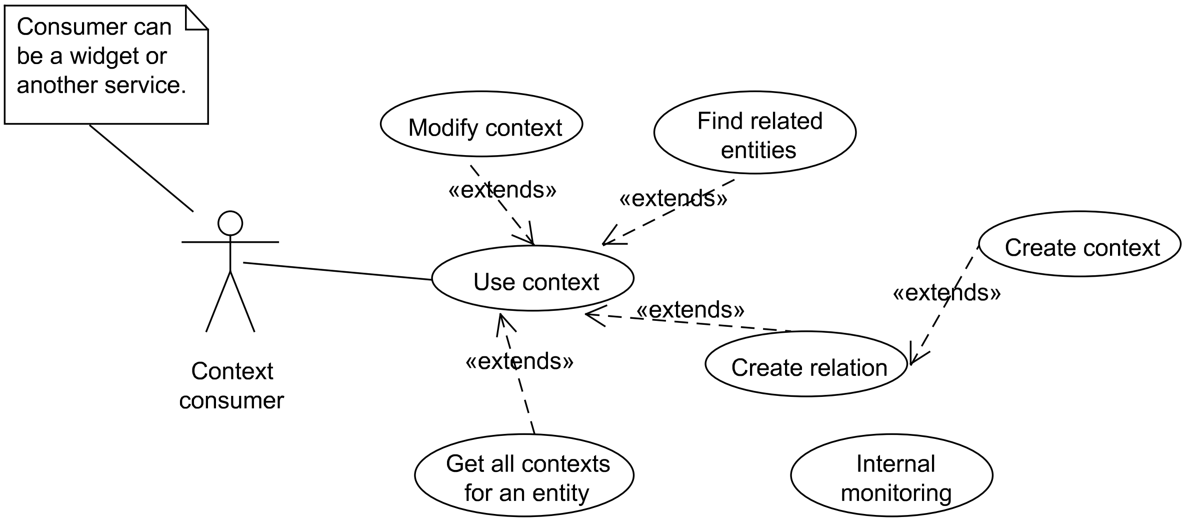

Figure 7 - Context use cases

To allow all queries shown in Figure 7 with all types of context described above, we need a flexible schema that can handle all requests. A first design idea is shown in Figure 8.

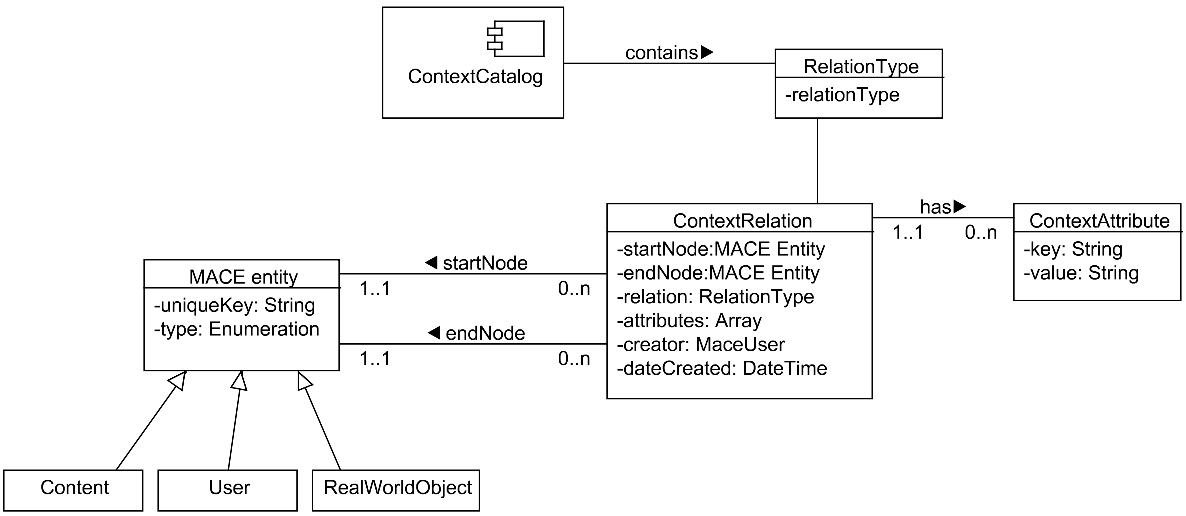

Figure 8 - Context schema

This schema allows managing relations between entities – a set of relations at a given time forms the context, which can be used for improved queries and filtering of results. Relations can be dynamic ("user visits building", "user views content") or static ("content was created at place", "content is related to other content") and allow access to a large mass of underlying data in a unified way. The idea of seeing context management as a set of relations that have attributes themselves was introduced by (Zimmermann and Lorenz 2007).

The management of relations between entities can be seen as specific search queries or filters on digital objects defining a view on the data according to a specific selection and weighting of attributes or tags. The results of a query – as defined by a ContextRelation – are based on the match of attributes or tags pertaining to the context tags of digital objects. The more tags match, the closer the association of digital objects between each other is.

Each ContextRelation referring to a metadata subset is of a certain RelationType and can have dynamic attributes. For example, when a user views a content object, the date/time of that action can neither be stored to the user nor to the content, because it would be overwritten when the next action happens with either the user or the objects. Instead, the date/time is a dynamic attribute and belongs to the ContextRelation – the action itself.

3.6 Summary

The different types of context and their attributes as described above form a taxonomy of contextual metadata which we will use to connect external databases with LORs in MACE. To validate and improve this taxonomy, we will put it to use and evaluate the usage of all attributes after the first prototype of the MACE infrastructure is running. It might then be necessary to extend the taxonomy presented here with additional attributes to allow for more content inclusion.

To measure success we will count hits to context related features and services by monitoring the usage of these services.

4 Taxonomy of competence and process metadata

Work package 5 aims at enriching and making use of competence metadata and process metadata for the learning objects in the MACE federated repositories. In the following section we will describe:

- Standards to model and represent competences and learning processes
- Competence models in the field of Architecture and Design
- The enrichment approach taken by the MACE project

4.1 Definitions of competences and competence metadata

Competences can be defined in a manifold way and there have been functional, cognitive, behaviouristic and lots of other approaches. For a nice overview and integration see (Cheetham & Chivers, 2005). In coordination MACE and the TENCompetence consortium interpret competence as all the factors for an actor to perform in an ecological niche. Performance in that sense includes the specific context that is necessary for the interpretation of a competence. Of course competences include competencies and knowledge that is necessary to put the competence into performance. In the context of previous projects in the field of architecture (like WINDS) competency taxonomies have been defined for the different areas relevant for architecture and design. Nevertheless, there is a high need to standardize those competence taxonomies and descriptions from different perspectives. On the one hand there is a European need for standardisation for the competences described and implemented in the European curricula for architects. On the other hand there is a need in competence driven education to have a shared and common set of competences or at least a common understanding of what competences are and about their role in the educational process.

For MACE several problems are related to competence metadata:

1. Selecting and defining a competence metadata schema that is compatible with the current approaches in ongoing standardisation efforts. Basically the defined schema must allow the import/export from and to existing standards like IMS, and HR-XML and foster the exchangeability of competence taxonomies and parts.
2. Support the definition of competence taxonomies for the domain of architecture and design, and in this context take into account the different approaches and granularity

of competences that are described in the professional development of architectural education. That means that on the one hand the schema must be able to support competence driven applications based on competence taxonomies on a fine granular level describing 200-500 competencies, on the other hand it should be possible to represent higher level approaches like in the European directive describing the profession of an architect with 11 high level competencies.

3. Defining the schema in a way that it supports bottom up and top down approaches for competence taxonomy definitions. Basically the schema should allow the definition of a starting set of competencies but also support the continuous update and be able to manage multiple interpretations of competencies.
4. Enable the easy metatagging of knowledge resources with competence metadata in an approach with little overhead and best as a side effect of using resources.
5. Support the integration of competence metadata with other types of metadata and explore the possibilities for educational applications in enabling the user to explore competence descriptions that are contextualized in a community of practice.

An example for a complex competence could be running a customer project with all necessary skills and knowledge ranging from the recipes and skills for customer contact, project management, to the actual architectural knowledge and competencies. Beside the specific knowledge, competencies, and skills the context in the ecological niche has an important impact and for enabling people to act in such an ecological niche effectively meta competencies are also necessary. A main idea behind the current approach in MACE for competence metadata generation is the fact that competencies are hidden entities and that real world competencies are often described by people having a competence or by artefacts requiring a certain competence for production or usage.

4.1.1 Current standards to model competences

IMS/IEEE RDCEO:

The IMS/IEEE RDCEO specification defines an information model for describing, referencing, and exchanging definitions of competencies, primarily in the context of online and distributed learning. This specification, aims to provide the means for formally representing the key characteristics of a competency, independently from its use in a particular context. It thus aims to guarantee interoperability among e-training systems that deal with competency information, by allowing them to refer to common definitions with commonly recognized values.

HR-XML:

The HR-XML competencies schema allows capturing of information about evidence used to substantiate a competency and ratings and weights that can be used to rank, compare, and otherwise evaluate the sufficiency or desirability of a competency.

The schema is intended as a flexible, practical means to communicate both unstructured competency data (such as that that may be captured from a resume or profile) or structured competency data from a taxonomy.

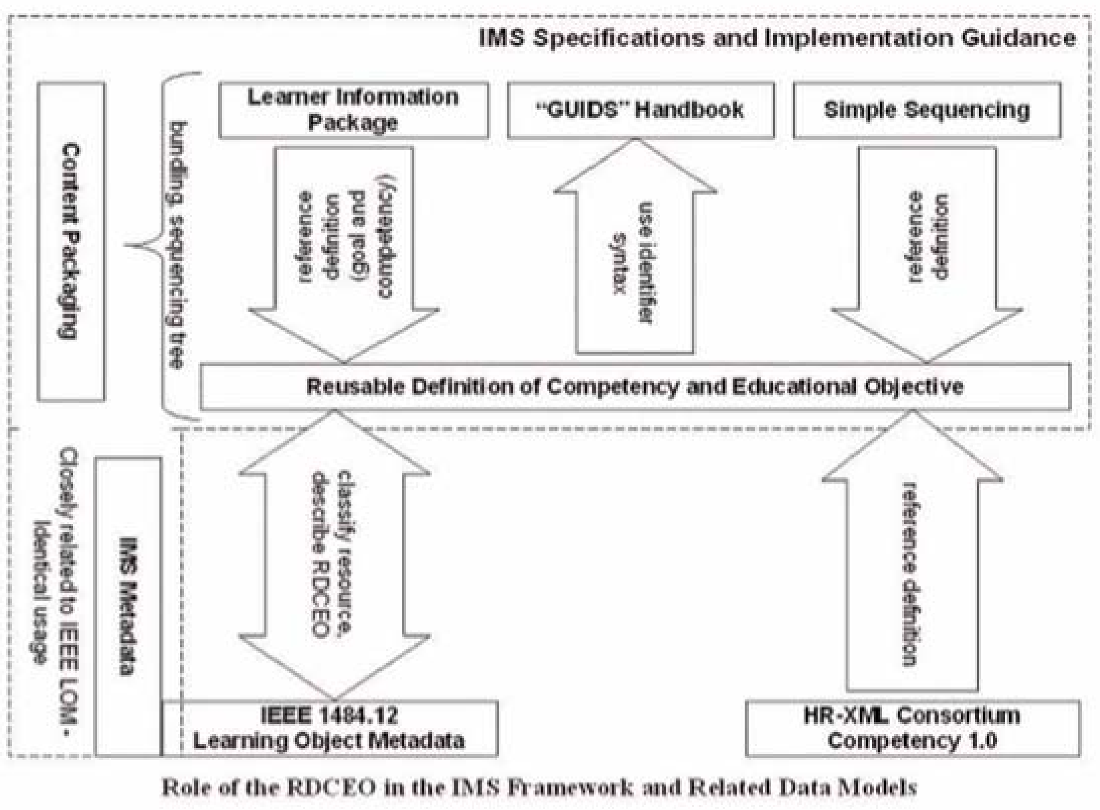

Figure 9: Role of the RDCEO in the IMS framework and related data models

In MACE in a first step all elements of the integrated applications profile should be integrated with the IEEE LOM schema. Therefore we describe the appropriate integration in the last section of this integrated deliverable. An overview of the basic relationships between LOM and competency definitions is given in Figure 9.

4.1.2 Competence models in the field of architecture

In the last decades the field of architecture and civil and building engineering has begun to orient towards a homogenization of the articulation already consolidated in the EU countries. The debate centered on the fields of civil engineering, environmental engineering, planning, architecture and building engineering is growing even more.

In the international field, the civil engineer is intended as that technician which traditionally has the capacities and the competencies of designing, managing and controlling the realization of structures and infrastructures (such as roads and transport infrastructures, water systems design and management, urban and territorial infrastructures, constructions for environment protection and others).

Even schematically, the building engineer is intended as that technician, who traditionally has the capacities and the competencies for building, managing and controlling the realization of constructions that satisfy human needs in the field of housing, health culture, education and so on. She is not so commonly intended, but frequently understood as that technician to whom the management of projects and processes in the building construction, in technological innovation, in experimentations, and the quality control of the building processes and products can be entrusted.

Also schematically, an architect is intended as that technician who is capable of combining specific design capacities in urban and architectural fields with the competencies and the mastery of the techniques relative to building or construction feasibility and programming as long as the competent control of the realization phases regards the aesthetical, functional and technical economical aspects. The architect is intended as a professional figure capable of operating with full title at the European level of architectural design, urban planning and development, monuments restoration, as a technician entrusted with the design, the planning and the execution, the control and the recuperation in the building construction and the architecture field, either as an autonomous professional or as a professional enrolled by corporations, public administrations and others.

In the context of the European project WINDS a competence catalogue for the professional fields of architects was also developed and the competencies where clustered in professional profiles for different areas. This allowed for relating the competencies needed in different professional profiles to each other and show their interrelations. The overall competence framework of WINDS contained 199 competencies which can be seen in the annex. The description of the WINDS competencies is highly targeted towards output and the result of certain activities, i.e. a competency is described with the result it allows to produce.

Furthermore the WINDS project defined different types of competencies:

- Mandatory competencies – personal, interpersonal and business skills common to all routes and compulsory for all candidates. In this category, knowledge and abilities about the capacity to work in team, the capacity to expatiate ideas and concepts with adapted graphical techniques and so on are included. Common competencies are a mix of technical, interpersonal, business and management skills. They are structured in levels and minimum standards which should be achieved. All students on all routes must achieve the defined level of the mandatory competencies.
- As the mandatory competence categories are common to all the educational paths, they don't constitute elements for the specification of the educational path. At times the authors give evidence to the mandatory competencies, because they believe emphasizing such competencies in the course description is significant. Only in these cases can this type of competencies be relevant in the educational offer of the course (e.g. in the case of a basic course).
- Core competencies – primary skills of the professional profiles. The 'core' activities of those practicing within a faculty should fulfill a defined level.
- Optional competencies – selected as additional skill requirements for the chosen route to qualification.

Both the core competencies and the optional competencies can be differently emphasized in the educational offer by an author:

- Generic competence: these are competencies that the teacher in general refers the contents of the course to and their acquisition results from the completed development of the course units by the students. Typical generic competencies are the mandatory competencies that have a generic cultural valence. In the description of the educational offer, the generic competencies are associated with the course but, they are not identifiable in any specific unit or course section;
- Extended competence: these are knowledge and skills resulting from a conceptual or applicative point of view, and therefore are not resolved in a single didactic unit, but interest and characterize more coarse units;

- Specific competence result finalized and circumscribed to one or two didactic units; usually characterized from either theoretical or operative instructional contents.

In the DIRECTIVE 2005/36/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 7 September 2005 on the recognition of professional qualifications for different profession including architects have been described.

The motivation is described as follows:

> *(27) Architectural design, the quality of buildings, their harmonious incorporation into their surroundings, respect for natural and urban landscapes and for the public heritage are a matter of public interest. Mutual recognition of qualifications should therefore be based on qualitative and quantitative criteria which ensure that the holders of recognized qualifications are in the position to understand and translate the needs of individuals, social groups and authorities as regards spatial planning, the design, organizations and realization of structures, conversation and the exploitations of the architectural heritage, and the protections of natural balances.*

In this directive 11 core competences for the profession of an architect are described as follows:

Competence

(a) ability to create architectural designs that satisfy both aesthetic and technical requirements;

(b) adequate knowledge of the history and theories of architecture and the related arts, technologies and human sciences;

(c) knowledge of the fine arts as an influence on the quality of architectural design;

(d) adequate knowledge of urban design, planning and the skills involved in the planning process;

(e) understanding of the relationship between people and buildings, and between buildings and their environment, and of the need to relate buildings and the spaces between them to human needs and scale;

(f) understanding of the profession of architecture and the role of the architect in society, in particular in preparing briefs that take account of social factors;

(g) understanding of the methods of investigation and preparation of the brief for a design project;

(h) understanding of the structural design, constructional and engineering problems associated with building design;

(i) adequate knowledge of physical problems and technologies and of the function of buildings so as to provide them with internal conditions of comfort and protection against the climate;

(j) necessary design skills to meet building users' requirements within the constraints imposed by cost factors and building regulations; first subparagraph.

(k) adequate knowledge of the industries, organisations, regulations and procedures involved in translating design concepts into buildings and integrating plans into overall planning.

4.1.3 References for architectural practice and connected competences in Europe

The “Architects’ Directive” for mutual recognition of architectural qualifications is 85/384/EEC. The full text of the Directive can be found by searching the Internet at europa.eu.int/eur-lex/en/consleg/pdf/1985/en_1985L0384_do_001.pdf. Appropriately qualified architects from EU Member States registered in their own country have been able to register in other Member States since 1987.

Organizations can be found on the International Union of Architects at www.uia-architectes.org websites’. Follow the links to Member Sections. The appropriate national organization should be approached for regulations concerning work within that country.

The RIBA has published a series of country guides on **architectural practice**:

Architectural practice in Europe: France, 1990; *Germany*, 1991; Italy, 1991; *Portugal*, 1991; *Spain*, 1991. CIRIA, the Construction Industry Research and Information Association, produced a series on the construction industry in various countries in the late 1980s. Although dated, these are still useful as a starting point: *The Belgian and Luxembourg construction industry: a guide for UK professionals*, 1991; *The French construction industry*, 1989; *The West German construction industry*, 1990; *The Iberian construction industry*, 1989, (covering Spain and Portugal); *The Italian construction* *industry*, 1990; and *The UK construction industry: a continental view*, 1992. The CIRIA website can be found at www.ciria.org.uk.

The Institute of Building Control produced a series of individual country reviews of **building regulations and technical provisions** throughout Europe and the EFTA countries in 1993, which are updated periodically. Entitled *Review of European Building Regulations and technical provisions*; the following countries are covered: *Austria*; *Belgium*; *Denmark*; *Finland*; *France*; *Germany*; *Greece*; *Ireland*; *Italy*; *Netherlands*; *Norway*; *Portugal*; *Spain*; *Sweden* and *the United Kingdom*.

Information can be searched on-line at www.architecture.com**.** Follow the links from the home page for all items added to stock since 1984. Articles on changes in practice in different countries are included. *World Architecture* produced a series of country reports in the period 1995 – 1996 which dealt with the building industry, major firms and architectural practices, and individual projects: *Benelux (Belgium, The Netherlands and Luxembourg),* June 1996, p. 56- 87; *Czech Republic,* May 1996, p. 46-73; *Franc*e, April 1996, p. 74-104; *Germany,* no 39, 1995, p. 88-111; *Spain,* no 40, 1995, p. 92-99; *United Kingdo*m, no 41, 1995, p. 72-85.

4.1.4 MACE competence metadata schema

As starting points for the MACE competence metadata schema the competence definition schemas from IEEE/IMS RDCEO and HR-XML are relevant as also the domain driven competence definitions from the field of architecture.

IMS/IEEE RDCEO:

The IMS/IEEE RDCEO information model is purposely minimalist and extensible in order to provide the flexibility to different organisations from different sections to describe their own Competence Model. The metadata elements in the Metadata category of IMS/IEEE RDCEO provide means for achieving the extensibility of the corresponding specification.

The IMS/IEEE RDCEO information model contains the following core elements:

- Identifier: A globally unique label that identifies this definition of competency or educational objective. The “Identifier” element consists of two other sub-elements: “Catalogue” and “Entry”.
- Title: A short name for this competency or educational objective. The “Title” may be repeated in multiple languages.

- Description: A narrative description of the competency or educational objective. The “Description” may be repeated in multiple languages.
- Definition: A structured description that provides a more complete definition of the competency or educational objective, using a collection of statements that determine a competency or an educational objective. Typically, such models define a competency or educational objective in terms of a “statement, conditions, criteria”, “proficiency, criteria, indicators”, “standards, performance indicators, outcomes”, “abilities, basic skills, content, process”, and similar sets of statements. The “Definition” consists of two other sub-elements: “Model Source” and “Statement”.
- Metadata: Optional meta-data record that further describe the RDCEO. The meta-data records must be conform to IEEE 1484-12.1-2002 (IEEE Learning Object Metadata).

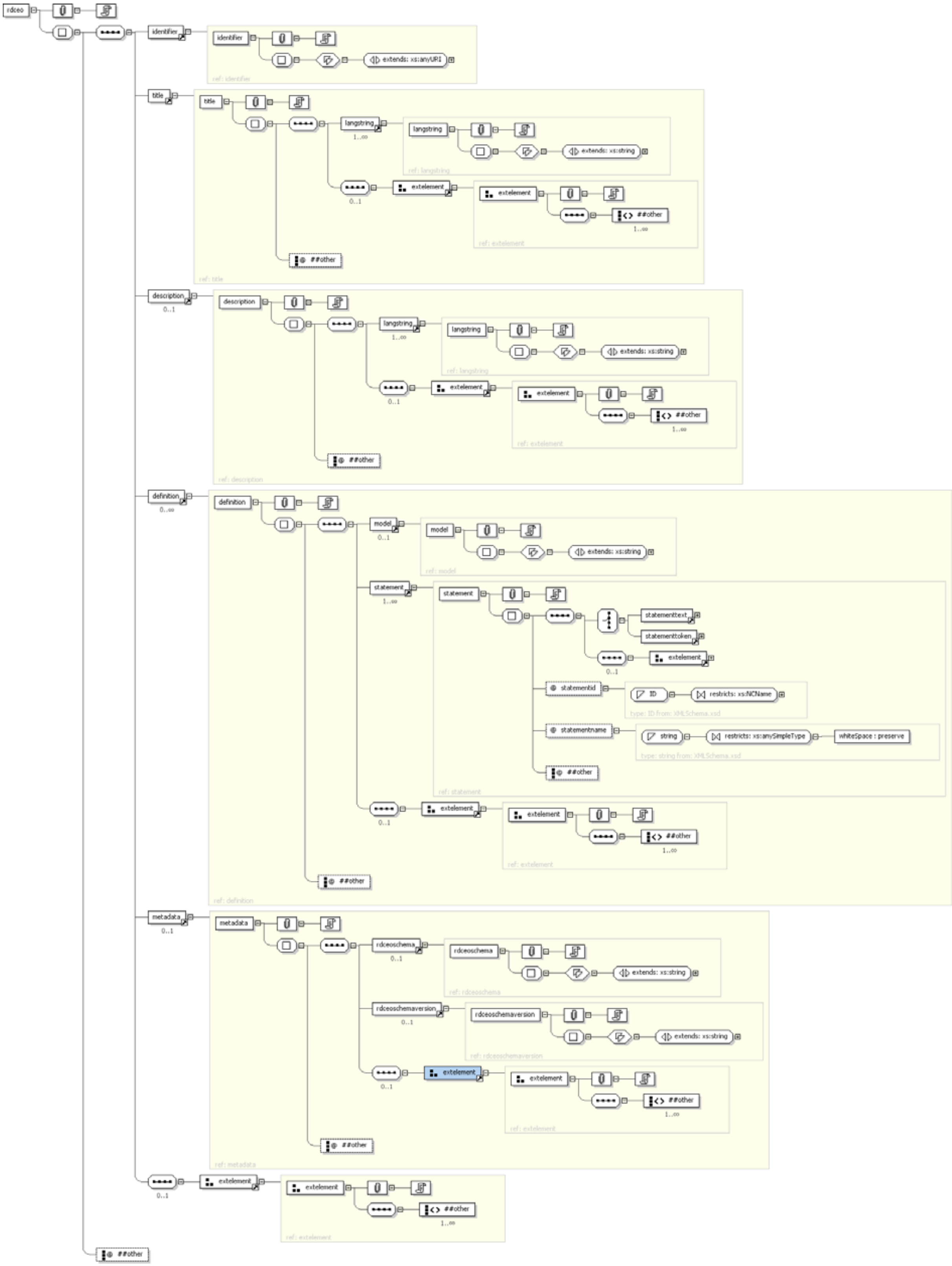

Figure 10: IMS/IEEE RDCEO Schema Visualization

No.	Name	Explanation
0	RDCEO	
1	Identifier	A globally unique label that identifies this Definition of Competency or Objective
1.1	Catalogue	The name or designator of the identification or cataloguing scheme for this entry. A cataloguing scheme.
1.2	Entry	The value of the identifier within the identification or cataloguing scheme that designates or identifies this Definition of Competency or Educational Objective. A namespace specific string
2	Title	Text label of this RCEOD
3	Description	Description of the Competency or Educational Objective
4	Definition	A list of statements according to a particular definition model
4.1	Model Source	Source of the Model being used
4.2	Statement	
4.2.1	Statement ID	A local identifying label for the Statement
4.2.2	Statement Name	Name of the Statement
4.2.3	Statement Text	Text of the Statement
4.2.4	Statement Token	Token value for the Statement
5	Metadata	Embedded Metadata about this RDCEO
5.1	RDCEO Schema	Describes the schema that defines and controls this RDCEO
5.2	RDCEO Schema Version	Describes the version of the above schema.
5.3	{Additional Metadata}	Additional embedded Metadata describing this RDCEO

HR-XML:

One of the HR-XML Competency Workgroup's important design goals was the development of a competency schema that would be relatively simple and sufficiently flexible to be useful within a variety of business contexts. Towards this end, HR-XML's Competency Workgroup wanted to avoid binding its schema to a definition of competency that would require difficult distinctions, such as the differences between "innate" and "learned" characteristics. Likewise, the workgroup did not want to bind the schema to a definition of competency that would limit the schema's usefulness in capturing and exchanging information about behaviourally revealed competencies versus those competencies evidenced by assessments, certificates, or degrees.

The HR-XML data model is purposely simple in order to provide the flexibility to different organisations to describe their own Competence Model within a variety of business contexts. Additionally this standard schema for the exchange of competency data will improve the communication across many HR systems and will simplify data transfer processes, thereby helping HR organizations save time and money.

The HR-XML information model contains the following core elements:

1. Name: A short name for the related competency.
2. Description: A narrative description of the competency.
3. Required: A Boolean used to indicate whether the CompetencyEvidence is mandatory for a particular position or given context.
4. CompetencyId: An identification code assigned to identify or classify the competency.
5. TaxonomyId: A code that identifies the taxonomy of the competency.
6. CompetencyEvidence: A text label that is used to capture information to substantiate the existence, sufficiency, or level of a Competency. CompetencyEvidence might include test results, reports, performance appraisals, evaluations, certificates, licenses, or a record of direct observation, such as a report given by a former supervisor or other employment reference.
7. CompetencyWeight: A text label that allows the capture of information on the relative importance of the Competency.
8. Competency: Competencies can be recursive. A competency may include other competencies. One competency might be decomposed into several component competencies, each of which might be separately measurable.
9. UserArea: Personal information about the individual that holds the defined competency.

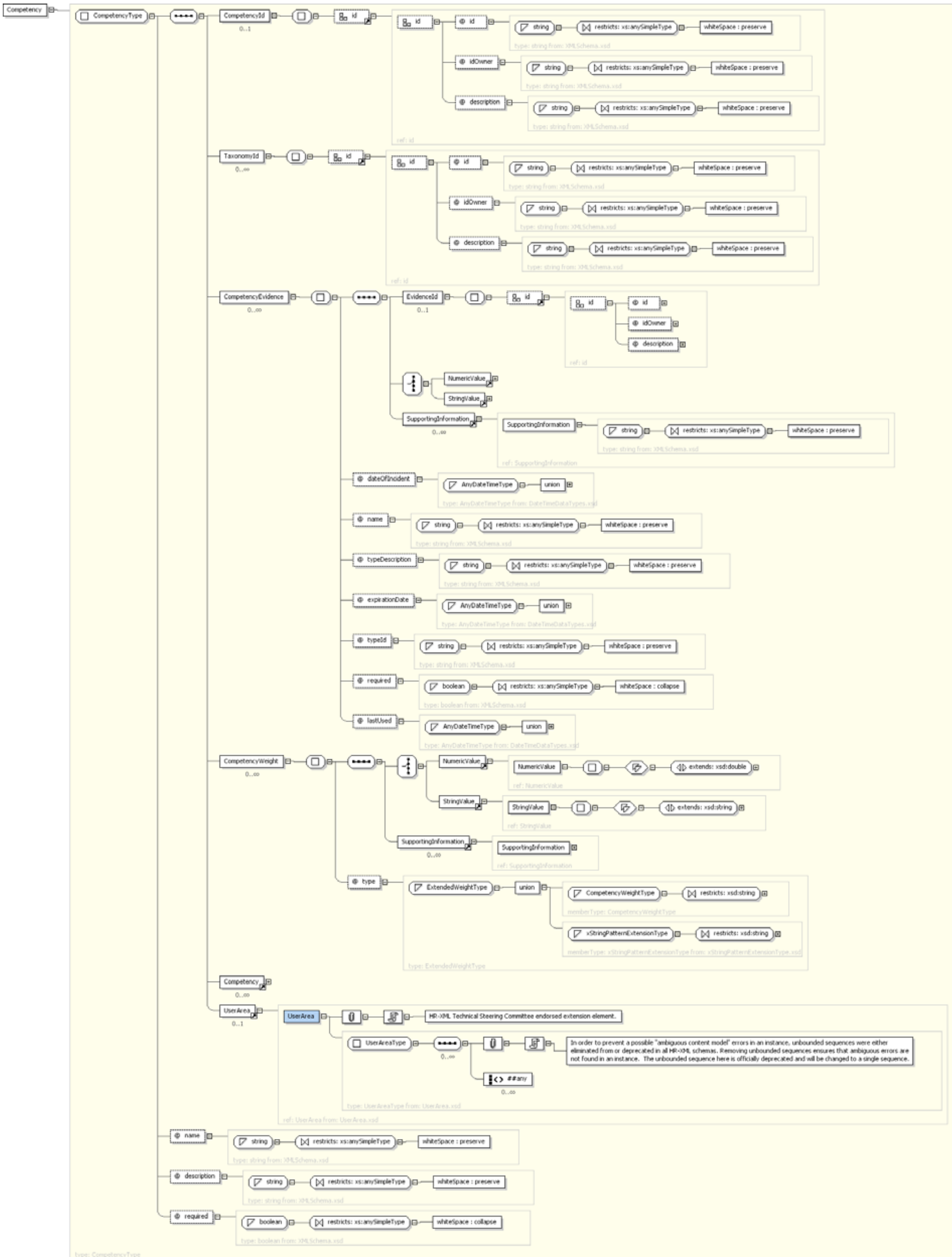

Figure 11: HRXML Schema Visualization

No.	Name	Explanation
0	Competency	
0.1	Name	The name for the related component.
0.2	Description	This optional attribute is available to provide additional information about the Id.
0.3	Required	A Boolean used to indicate whether the CompetencyEvidence is mandatory for a particular position or given context.
1	CompetencyId	An identification code assigned to identify or classify the Competency. A taxonomy might include an identification code for each Competency or identification codes might be agreed upon by trading partners.
2	TaxonomyId	A code that identifies the taxonomy.
3	CompetencyEvidence	CompetencyEvidence is used to capture information to substantiate the existence, sufficiency, or level of a Competency. CompetencyEvidence might include test results, reports, performance appraisals, evaluations, certificates, licenses, or a record of direct observation, such as a report given by a former supervisor or other employment reference.
3.1	EvidenceId	A code that identifies the CompetencyEvidence.
3.2	NumericValue	NumericValue is the required or measured level for the competency. The content of NumericValue is a rating value. The minValue, maxValue, interval, and description attributes of NumericValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
3.2.1	minValue	The minimum value of the rating scale.
3.2.2	maxValue	The maximum value of the rating scale.
3.2.3	interval	The increment or step for the relevant scale.
3.2.4	description	This optional attribute is available to provide additional information about the Id.
3.3	StringValue	StringValue is the required or measured level for the competency. The content of StringValue is a rating value. The minValue, maxValue, and name attributes of StringValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
3.3.1	minValue	The minimum value of the rating scale.

No.	Name	Explanation
3.3.2	maxValue	The maximum value of the rating scale.
3.3.3	description	This optional attribute is available to provide additional information about the Id.
3.4	SupportingInformation	Contains additional descriptive information to substantiate or clarify a rating, measure, value, etc.
3.5	dateOfIncident	The date on which the CompetenceEvidence first establishes the existence of the Competency.
3.6	name	The name for the related component.
3.7	typeDescription	A description of the type of CompetencyEvidence.
3.8	expirationDate	The identification of any applicable expiration date, such as the date that a license or certification expires.
3.9	typeId	A code identifying the type of CompetencyEvidence.
3.10	required	A boolean used to indicate whether the CompetencyEvidence is mandatory for a particular position or given context.
3.11	lastUsed	A requirement or assertion for the date on which the Competency was last used.
4	CompetencyWeight	CompetencyWeight allows the capture of information on the relative importance of the Competency or the sufficiency required or other type of dimension. An extensible "type" attribute is available so that custom weights or dimensions may be specified. Multiple CompetencyWeights are permitted since more than one type might apply to the competency.
4.1	NumericValue	NumericValue is the required or desired level for the competency. The content of NumericValue is a rating value. The minValue, maxValue, interval, and description attributes of NumericValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
4.1.1	minValue	The minimum value of the rating scale.
4.1.2	maxValue	The maximum value of the rating scale.
4.1.3	interval	The increment or step for the relevant scale.
4.1.4	description	This optional attribute is available to provide additional information about the Id.
4.2	StringValue	StringValue is the required or measured level for the competency. The content of StringValue is a rating value.

No.	Name	Explanation
		The minValue, maxValue, and name attributes of StringValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
4.2.1	minValue	The minimum value of the rating scale.
4.2.2	maxValue	The maximum value of the rating scale.
4.2.3	description	This optional attribute is available to provide additional information about the Id.
4.3	SupportingInformation	Contains additional descriptive information to substantiate or clarify a rating, measure, value, etc.
4.4	type	Identifies the type of CompetencyWeight. Enumerated values are: levelOfInterest (A level of interest asserted or required for the competency) and skillLevel (A level of skill asserted or required for the competency).
5	Competency	recursion
6	UserArea	extensible area

As a key issue to enable a cost effective and enduring enrichment process in MACE the main requirements for the schema design were on the one hand to be compliant to standards and on the other hand to enable an easy and understandable communication about competences. For the enrichment process we will use a simplified competence schema, which is compliant and extensible towards the above-described standards, but which enables us to use competence cards as the main mean to communicate with the user about competencies and allow the end users to do easy metatagging and annotation of competencies. For a definition of the complete schema and the use cases developed for the competence card application, see the competence card specification.

The Competence Cards provide a flexible way of describing competences and connecting the various metadata to the competence. This allows us to display information to the user about the relation between the learning objects and the architectural competencies.

The Competence Card Schema contains the following core elements:

1. Domain: a domain in which the competence belongs
2. Competence: the information about the competence itself (title, description, etc.)
3. Competence maps: the relation diagram

4. Proficiency scales: describes which proficiency scales are used to rate relations to the competence
5. Related resources: a list of resources related to the competence
6. Related persons: a list of persons related to the competence
7. Related evidences: a list of evidences related to the competence

The elements in the Competence Card Schema allow us to export the competence information to the standards IMS/IEEE RDCEO or HR-XML.

The Competence Card definition is an extension to the currently available competence definitions. We have to write a specification ourselves or adapt to a specification and expand it with the missing elements. The most extensive specification and well fitted for our purposes is the HR-XML specification. It already has a competencyEvidence so only the competencyExperts and competencyResources have to be added to that schema. In that way we will also be compatible with the HR-XML schema.

This would result in a schema like:

No.	Name	Explanation
0	Competency	
0.1	Name	The name for the related component.
0.2	Description	This optional attribute is available to provide additional information about the Id.
0.3	Required	A boolean used to indicate whether the CompetencyEvidence is mandatory for a particular position or given context.
1	CompetencyId	An identification code assigned to identify or classify the Competency. A taxonomy might include an identification code for each Competency or identification codes might be agreed upon by trading partners.
2	TaxonomyId	A code that identifies the taxonomy.
3	CompetencyEvidence	CompetencyEvidence is used to capture information to substantiate the existance, sufficiency, or level of a Competency. CompetencyEvidence might include test results, reports, performance appraisals, evaluations, certificates, licenses, or a record of direct observation, such as a report given by a former supervisor or other employment reference.

No.	Name	Explanation
3.1	EvidenceId	A code that identifies the CompetencyEvidence.
3.2	NumericValue	NumericValue is the required or measured level for the competency. The content of NumericValue is a rating value. The minValue, maxValue, interval, and description attributes of NumericValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
3.3	StringValue	StringValue is the required or measured level for the competency. The content of StringValue is a rating value. The minValue, maxValue, and name attributes of StringValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
3.4	SupportingInformation	Contains additional descriptive information to substantiate or clarify a rating, measure, value, etc.
3.5	dateOfIncident	The date on which the CompetenceEvidence first establishes the existence of the Competency.
3.6	name	The name for the related component.
3.7	typeDescription	A description of the type of CompetencyEvidence.
3.8	expirationDate	The identification of any applicable expiration date, such as the date that a license or certification expires.
3.9	typeId	A code identifying the type of CompetencyEvidence.
3.10	required	A boolean used to indicate whether the CompetencyEvidence is mandatory for a particular position or given context.
3.11	lastUsed	A requirement or assertion for the date on which the Competency was last used.
4	CompetencyWeight	CompetencyWeight allows the capture of information on the relative importance of the Competency or the sufficiency required or other type of dimension. An extensible “type” attribute is available so that custom weights or dimensions may be specified. Multiple CompetencyWeights are permitted since more than one type might apply to the competency.
4.1	NumericValue	NumericValue is the required or desired level for the competency. The content of NumericValue is a rating value. The minValue, maxValue, interval, and

No.	Name	Explanation
		description attributes of NumericValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
4.2	StringValue	StringValue is the required or measured level for the competency. The content of StringValue is a rating value. The minValue, maxValue, and name attributes of StringValue provide information about the rating scale that is being used, so that the rating value can be interpreted.
4.3	SupportingInformation	Contains additional descriptive information to substantiate or clarify a rating, measure, value, etc.
4.4	type	Identifies the type of CompetencyWeight. Enumerated values are: levelOfInterest (A level of interest asserted or required for the competency) and skillLevel (A level of skill asserted or required for the competency).
5	Competency	recursion
6	CompetencyExperts	collection of experts related to this competence
6.1	Expert	the expert related to this competence
6.1.1	expertId	ID of the expert
6.1.2	firstName	first name of the expert
6.1.3	lastName	last name of the expert
7	CompetencyResources	collection of resources related to this competence
7.1	Resources	a resource
7.1.1	resourceId	ID of the resource
7.1.2	Name	name of the resource
7.1.3	URI	URI of the resource

4.2 Definitions and modelling of learning processes

For modelling learning processes and best practices the project follows an incremental approach. As first workshops with end users have shown it is difficult to elicit instructional schemas and classify the involved activities according to a given vocabulary of activities. The use case analysis described in work package 2 lists several activities relevant in different activities including instructional design, learning and assessment.

- Instructional Design Use Cases: Evaluation of market requirements in terms of professional profiles, Review of regulations and laws concerning education and professions, Definition of educational objectives in terms of competencies, Decomposition of competencies in terms of knowledge and skills, Planning of didactic modules, Evaluation of credits, Collecting documents for lesson authoring, Adding case studies to a task class (4C/ID)

- Learning Use Cases: Learning theoretical concepts, Applying theoretical concepts, Learning conceptual and visual associations, Re-using best practices and/or common mistakes, Case Based Design
- Assessment Use Cases: Self-assessment, Social Activities, Collaborative design, Field trip

Describing the learning processes in detail requires a formalism and method to collect the instructional designs, best a standardized vocabulary for learning activities performed in the educational process and examples and best practices actually applied and instantiated in the different educational contexts.

As first steps in this document introduces the schema to represent the instructional designs and the methods for the elicitation. Furthermore a first version of the vocabulary found will be described.

4.2.1 Background for instructional/learning design for architecture

Instructional design theories offer guidance about what methods to use in what situations. In short they identify the most effective instruction suitable to the educational objectives and the context of learning. Instructional design theories are goal-oriented, and, unlike most theories that are descriptive, they are prescriptive in nature as they offer guidelines as to what method to use to attain a given goal. Their main aspects are:

- Methods for facilitating learning
- Situations in which to use those methods
- Instructional conditions (nature of content, learner, learning environment, constraints)
- Desired instructional outcomes (effectiveness, efficiency, appeal).

Instructional design theories should not be mistaken with Learning theories such as Behaviorism, Cognitivism or Constructivism. Instructional design theories do in fact mirror those paradigms and turn them into educational practice. Therefore they follow the demands of our society for a variety of attitudes towards learning and its provision. The new paradigm of Instructional design theories should thus incorporate the previous one, but it also needs to be restructured following the actual needs of learners and teachers:

- Learning based
- Development of initiative, teamwork, thinking skills and diversity
- Customization of learning process

In this perspective, the new paradigm should consider and support the full range of types of learning with all its varieties and forms. It should focus on soft skills and life long abilities. Finally, it should provide for individual differences, being flexible and adaptive. (Reigeluth, C.M., 1999). Many Instructional design theories and discourses have tried to answer these needs: constructivist design theories for problem solving (Jonassen, 1999; Reigeluth, 1999; Schwarz, Lin, Brophy, & Bransford, 1999); calls for new directions on technology-based design (Kozma, 2000); design methodologies for acquiring complex skills (van Merrienboer, 1997); and a ‘first principles of instruction’ approach to instructional design (Merrill, 2002).

Proposals for models of instructional design for teaching Architecture design generally focus on particular aspects of the design process. So according to the attention paid more to one or to another we can make a brief review of the methods that can be used in instructional design for teaching Architecture Design. However, we basically assume a Socratic idea of Pedagogy, which claims that self-learning and self-discovery are the only ways to learn. These foundations have obviously been developed and expanded by studies and research of several cognitive theorists such as Bloom (1956), Gardner (1983), Gagnè (1985), psychologists as Dewey (1910) Rogers (1969) Vigotsky (1978), philosophers as Schön (1983) Goleman (1995) and instructional design professionals like Merril (1983), Reigeluth (1983) Jonassen (1999).

The general model that has been developed from these theories is basically a constructivist model of a learning environment in which students are engaged and kept active, constructive, collaborative, intentional, complex, conversational and reflective (Jonassen, website) using instructional processes that are:

- Modelling (demonstrate how and why to perform the activities necessary for the completion of the task or objective)

- Coaching (provide students with encouragement, diagnosis, direction and feedback)
- Scaffolding (adjust the task to match the students' level of performance, activate a support system with conceptual, meta-cognitive, procedural and strategic aims.)

Modelling provides learners with an example of the desired performance, worked examples of problems solved by an expert. Not only the overt performance, but also the covert performance can be modelled, that is the reasoning behind each step of the solution process. A good coach motivates the learners, analyses their performances, gives feedback and advice, and provokes reflection and elaboration of what was learned. Scaffolding may be provided according to the purposes served, for example conceptual scaffolding guides the learner in what to consider, meta-cognitive scaffolding suggests ways to think about the problem, procedural scaffolding gives hints on how to use the system, strategic scaffolding helps in analysing and approaching learning tasks.

"Authentic" learning environments in the constructivist tradition are situations that allow learners to create their own personal knowledge in a particular task environment, a surrogate to the actual problem-solving environment. Learning design methods for creating "authentic" learning environments in the constructivist tradition, which are used in the teaching of Architecture, are the following:

- Problem Based Learning
- Case Based Instruction
- Discourse Based Learning

4.2.2 Problem Based Learning (PBL) and related computer aided systems

If we focus on the problem solving aspect of the Design Process, the instructional method that is applied is Problem-Based Learning, which is a pedagogical strategy for posing significant, contextualized, real world situations and providing resources, guidance, and instruction to learners as they develop content, knowledge and problem-solving skills. A "learning by doing" instrument to allow students to:

- Engage the challenge and become self-directed (management skills). In PBL students learn because they are interested in solving a concrete problem and not because they are told to do so. They have to manage the steps to solve the problem.
- Recognise what is needed and how to go about it (learning skills).

- Apply all appropriate knowledge bases to solve the problem (application)
- Collaborate with peers (cooperative skills)
- Reflect on the process in order to improve their work (self-awareness)
- Reason effectively and creatively (critical thinking)

Although this approach is definitely student-centered, teachers have still a crucial role in the learning process. In PBL teachers are requested not to give answers but to provide students with more room for active self-learning, thus they need a new set of teaching roles and teaching skills. If we take Bibace et al. 's Teaching style Model, the teachers' verbal behaviours which has been shown to be more effective are the collaborative and facilitative styles, where students take an active role in learning and teachers facilitate and monitor students' progress.

Most of the difficulties that arise in PBL implementation are due to the changes in students and teachers' roles, especially in an environment where collaborative work is not the norm. Students get the perception that they are not learning enough content since the knowledge base is not explicitly developed, whereas teachers find it difficult to modify their teaching style and tend to give mini-lectures, instead. A further obstacle may be the time needed to implement a really effective PBL process, which does often not coincide with the curriculum time requirements. Finally assessment may present problems connected with the assessment of the group performance.

On the other hand, PBL and technology can be easily coupled to design an effective learning environment, for example problems can be presented in realistic and motivating ways. Software can be used to create simulations and make it easy to see and understand certain processes, or to make representational mental models explicitly visualised, and to turn hypothesis into formal models. Besides web-based collaborative tools such as forums, chats or wikis make group work possible and fruitful.

In the teaching of Architecture Design PBL is an invaluable instrument to develop students' problem solving and self-management skills, as well as their ability to integrate various disciplinary content/knowledge.

4.2.3 Case Based Instruction (CBI)

Architecture Design instruction is different from other academic disciplines because students are directed to a corpus of desirable outcomes rather than principles or theories. Therefore knowledge disseminated in the design studio is often packaged in the form of precedents or generalizations drawn from a number of instances, rather than from first principles. Precedents are specific designs or buildings, which are exemplary, often past solutions to specific design problems; in some case they are negative ones or failures. They should contain three indispensable aspects:

- A description of the context that surrounds each case
- Descriptions of the various stages of progression that case has gone through prior to its solution, as well as its solution
- A description of the processes or methods that are relevant for these states.

Learning takes place through examination, analysis, and abstraction of the information contained in the case representation by the students with the help of the instructor. The abstractions derived by this analysis are useful to bridge the gap between conceptual and physical variables that are the basis of spatial design. The ability to use those abstract concepts to give rise and later to justify and explain explicit description of design is exactly the core of the competence the students should acquire at the end of the course.

In CBI there is a constant interplay of skill and knowledge, of theory and practice. Discovering how principles affect solutions and vice versa is the way through the solution of a design problem. This is made by a generation-test experiential procedure, which by and large simulates the real environment of the architectural design office where a process of “reflection in action” takes place. The instructor, who acts as a facilitator, guiding the students through exploration and discovery, but also structuring their findings, also plays an important role.

The Case-Based Reasoning cycle (Riesbeck & Schank, 1989; Kolodner,1993, Jon A. Elorriaga and I. Fernández-Castro, 2000) involves four processes: retrieve, reuse, revise and retain which are described below.

- Retrieve: The current problem is analysed and the relevant features of the situation are used to search the CM through the indices in order to find a set of similar cases. The retrieved cases are tested with the aim of selecting the most appropriate one.

- Reuse: Once a case is chosen the case-based reasoner must adapt it to fit the characteristics of the current situation. There exist two kinds of adaptation in CBR: structured adaptation, in which adaptation rules are applied directly to the retrieved solution, and derivational adaptation, which produces the new solution by re-using the method that generated the solution for the retrieved case. The adapted solution is applied to the problem in the real domain. However, the application is considered outside of the CBR process, as it is performed by an external system.
- Revise: After applying the solution, the next CBR phase consists of the revision of the obtained outcome to check whether the proposed solution really solves the problem or not. If the solution was unsuccessful, the failure is explained and the solution is repaired.
- Retain: In this phase the current problem, the proposed solution and the outcome are stored in the CM. A new case is created extracting some data from the new experience, i.e. the relevant features, the applied solution and the outcome. If the solution was successful, the new case is stored to help in the future problem solving activity. If the solution failed the case is stored in the CM together with the failure and its repair, in order to avoid similar failures. The new case must be integrated into the CM and the indices updated.

4.2.4 Discourse based learning and tutorial dialogue

Both PBL and CBI involve extensive tutorial intervention, which play an important part on learning outcomes. The discursive nature of learning has become the focus of a lot of research across different fields: education, cognitive studies, linguistic and pragmatics since when educational researchers transferred the focus from the individual to the social process of learning (Vygotsky, 1978). The socio-cognitive views of thinking and learning argue that most knowledge is an interpretation of personal experiences and also social in nature: in other words knowledge is jointly constructed in interaction. Discourse based learning approach is based on such assumption that knowledge is a social product and mediated through language (Gee and Green, 1998). Knowing and thinking are situated discourses that students exchange with the text, the teacher and with the other students (Bruner, 1990).

Students learn by talking, as well. Several analysis of classroom interactions (Brown, 1993; Lapadat, 2003; Keefer et al, 2000) have pointed out that through reasoning and arguing collaboratively on different ideas, students learn to think critically. As a result, they also improve their understanding by discussion on specific knowledge objects. Moreover, by encouraging narrative skills and eliciting self-explanation (Chi et al., 1994), students are also stimulated to express their implicit theories. Therefore, discourse in educational context supports knowledge revision process and conceptual change (Mason, 2001; Lapadat, 2000). These theories point out the importance of speech acts, patterns of the exchange, turn-taking models and roles that teachers and students take during the educational dialogue. There is evidence of teacher discourse style affecting the kind of discourse created in learning situations, and also the distribution of turns and the illocutionary power of speech acts may have a consequence in the development of learning.

Conversational analysis has been crucial to examine the effects that speech acts produced by the teachers may have on students' learning outcomes. Teachers' speech acts and classroom talk have been investigated and classified by several linguists such as Barnes (1969) Flanders (1970) and Sinclair and Brazil (1982). However, a rigid classification often does not completely account for what happens in student-tutor interaction, since speech acts do not always perform one type of function at a time. So we found it more feasible to use a variety of categories, which come both from Sinclair and Brazil (1982) and from other analyses for example Porayska-Pomsta, Mellish and Pain (2000). In addition, we also should take into consideration further studies about the peculiar aspects of the tutoring dialogue. According to Graesser, Person and Magliano (1995), a typical tutoring dialogue is described through a "tutoring frame", which consists of five stages:

1. Tutor asks an initiating question;
2. Student provides a preliminary answer;
3. Tutor gives (confirmatory or negative) feedback on whether the answer is correct or not;
4. Tutor scaffolds to improve or elaborate the student's answer in a successive series of exchanges (taking 5–10 turns, according to Graesser et al. 1995);
5. Tutor gauges student's understanding of the answer.

The tutoring frame suggests that tutors basically dominate the dialogue, dictate the agenda, craft the appropriate next question, select the next example and problem to be solved, pose the analogy and counterexample, give the feedback, and so forth.

Nevertheless, the effectiveness of the tutoring dialogue in the promotion of learning depends also on other two crucial components:

- The active involvement of the students in the construction of knowledge; forming hypotheses and justifications (Chan, Burtis, Scardamalia & Bereiter, 1992); forming analogies (Chi et al., 1994); reflecting, summarizing, and predicting (Palinscar & Brown, 1984); justifying, criticizing, and exploring (Collins et al., 1989); or revising one's existing knowledge (Chi, 2000).
- The value of interaction in the construction of knowledge. "Interactive" acts are those comments that elicit a response from the students, such as asking the students either content or comprehension gauging questions, or scaffolding them. "Non-interactive" are instead comments those in which the tutors give lengthy explanations without giving the students a turn to respond, or tutors giving feedback followed immediately by an explanation, without interjecting a comprehension-gauging question.

By focusing on the kinds of interaction emerging as particularly important for learning, and by abstracting formal and computational definitions of them, dialogue analysis can help bridge the gap between the empirical evaluation of interaction and the design of Intelligent Educational Systems (IESs) capable of interacting with their users. Furthermore, since the issue of tutorial skills and tutorial discourse is crucial in the implementation of learning through PBL and other related methods, we have observed and described the development of tutorial discourse in the interaction between students and teacher/tutor in our research. It aimed at modelling tutorial dialogue in homework revision sessions in Architectural Design courses.

4.2.5 Representation of learning designs

IMS Learning Design (IMS-LD) is a specification to model learning strategies and represent learning processes of users at runtime. Furthermore, IMS-LD is focused on the design of pedagogical models to manage learning activities which are linked to learning objects within a learning flow. The structure of an IMS-LD design consists of a learning flow with plays, acts, activities, activity structures and environments and is flexible enough to provide several personalized itineraries depending on the role assigned.

The structure of IMS-LD is detailed in three different and complementary steps:

1. Level A is the main part of the specification as it provides the bottom-line to build any Unit of Learning with the elements Method, Play, Act, Role, Role-part, Learning activity, Support activity and Environment; Level A provides the basic structure of activities and processes along with the roles´ definition;
2. Level B incorporates some other elements to create more expressive lesson plans (for instance, based on adaptation, collaboration or monitoring) using Properties, Conditions, Calculations, Monitoring services and Global elements. Level B is the actual key for adaptation; and
3. Level C adds Notifications, and enables a mean to trigger some dependant activities. Every layer adds new possibilities for representing more complex educational scenarios and it is built on the previous one. The combination of some of these elements is the key for modelling several instructional methods.

Technically, the specification defines a XML document, called imsmanifest.xml, describing a pedagogic scenario in detail, and linking the actual resources in a variety of formats with it and it is embedded in an IMS Content Packaging information package.

4.2.6 Methodology to elicit learning designs in architecture

Based on the approach of the Australian AUTC project a methodology has been developed to describe and elicit instructional design in a simple and understandable way, which can later be formalized in a standardized approach following an IMS-LD Notification. Therefore the protocol from the AUTC project was taken to illustrate a learning design in a temporal format to provide a standard form of communication to describe different kinds of learning designs and highlight the key features of each design.

The basis for this construct is informed by the work of Oliver (1999, 2001) and Oliver and Herrington (2001) that identifies the critical elements required in a learning design, particularly when ICT mediated. The critical elements comprise the tasks or activities learners are required to perform, the content or resources learners interact with, and the human support mechanisms provided to assist learners to engage with the tasks and resources.

The sequence of a learning design outlines these three components (tasks, resources and supports) as they are used over a period of time. Thus, the sequence illustrates the learning activities, the resources and supports, plus the artefacts the students produce to arrive at the final component of the sequence which is the learning outcomes. An example is illustrated in Figure 12.

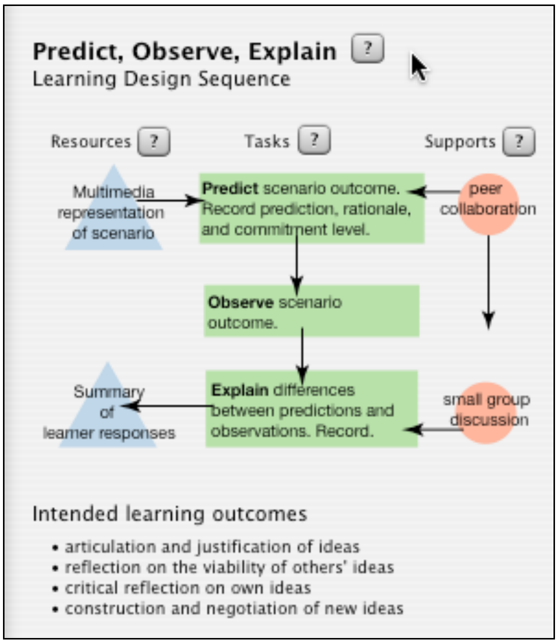

Figure 12: One of the designs from the Learning Designs project

Representing tasks: The tasks are represented by a series of rectangles, arranged vertically. These activities represent the learner's "journey". Each rectangle has a description of what the learners are required to do or produce. Activities that are assessable are distinguished with an asterisk (*).

Representing resources: Learning resources are represented by triangles to the left of the activity sequence. An arrow from a resource (triangle) to an activity (square) indicates that resources are available to the student when doing the activity. An arrow from an activity (square) to a resource (triangle) indicates that a resource is produced during the activity and becomes a resource for others to use later.

Representing supports: The learning supports are represented by circles to the right of the activity sequence. An arrow from a "circle" to a "square" indicates that support strategies are being used to assist the students in their learning.

Representing different combinations of activities, resources and supports: Resources and supports can be specific to an activity, they can be introduced before beginning an activity or when an activity is complete, or they may be available for the entire duration of the learning experience. To represent this graphically, the following convention is suggested:

- If learning resources or supports are limited to particular activities, their availability is represented with a horizontal arrow to the specific activity for which they are available. If a learning resource or support is available for multiple activities then the resource triangle and/or support circle is drawn once (where it is firstly introduced to the students) and a vertical arrow indicates the resource and/or support is available for period of time.
- If students produce artefacts from a learning activity, which are then used as resources for subsequent activities, an arrow is drawn from the activity to the resource.

This approach will be used also in further workshops with architects and educators to elicit some base templates and a list of activities used for metatagging learning objects from the MACE repositories.

4.2.7 Best practices in the field of architecture education

Related to the different types of activities and learning resources used in the field of architecture education following instructional methods, techniques, and assessment methods have been identified from first user workshops.

In order to understand how the actual reality of Design Instruction can be referred to reference tutoring discourse frames and cognitive models of Design, an extensive protocol analysis has been taken as basis on a set of design revision session in regular courses of Architecture-Engineering Degree at Università Politecnica delle Marche (Italy). It involved two teachers and a number of student groups. The research has been conducted adopting the following methodology:

- Observation and videotaping of students-teacher exchanges while reviewing design home-works.
- Protocol analysis of videotapes according to the representation models of design and interaction and negotiation of meaning (De Grassi, 2005).
- Semi-structured interviews of students and tutor in order to triangulate the observation and analyze data from more than one perspective and increase their reliability.

Protocol Analysis is defined as a type of qualitative investigation, which consists of submitting transcripts and similar written records to systematic examination. It is certainly a time consuming method but given the field we are investigating it is also the most reliable way to find out processes and critical areas, since it also gives the possibility to analyze discourse in action. The main focus of this technique is to get insights into the processes rather than the products. Its biases have been acknowledged by the scientific community but it still remains as somewhere in the middle ground between “hard” empirical methods and “weak” purely observational methods. We have thus video-recorded two subsequent revision sessions for 3 groups of students, lasting approx 20min each, transcribed the script and segmented it. The coding scheme has been developed and the segments categorized accordingly. The coding scheme obviously corresponds to the framework of research that we projected on to the data (figure 1). The categories that have been coded are:

Design cognition process

- Macro Strategies:
- Multimodal perceptual design reasoning (conceptual level, perceptual level and external world)
- Micro Strategies:
- Proposing a Solution (proposing, clarifying, retracting, making a design decision, consulting external information, postponing a design action)
- Analyzing Solution (analyzing, justifying, evaluating)
- Reflection-in Action
- Gestures & Sketches

Tutorial discourse

- Turns (time for students and tutor)
- Acts (according to the tutoring frame)
- Discourse Strategies (empathy, pauses, silence, fillers, etc.)

In that sense a set of first activities in instructional practices have been identified which will be used as a first set for metetagging learning objects from the MACE object repositories.

Instructional methods:

- Lecture/presentation
- Demonstration/Modeling
- Problem Solving Lab.
- Case Based Instruction
- Discourse Based Learning (Tutorial)
- Cooperative Group Learning
- Field Trip
- Autonomous learning

Techniques/Activities/Tasks

- Presentation
- Case study
- Practice exercises
- Sketches/drawings
- Homework Revision Sessions
- Recurrent Mistakes
- Simulations/Role Plays
- Tutorials
- Forums /Faqs

5 Taxonomy of usage related / social metadata

5.1 Introduction

In a digital world with a noticeably increase in the information overload, personalized services are getting more and more attention to provide a user with the appropriate material and services. That can only be achieved by building a detailed user profile about the user. For example, Google builds a detailed profile for each user from her previous interactions with the system. In the context of learning, providing the user with the appropriate material in the appropriate time and format requires collecting and mining usage information about a user's previous activities in different systems she may have used. This enables building a profile that is based on a user's actual behaviour when working with learning material and associated systems, not on predefined stereotypes. Such data can later be used by information systems to conclude on user aims and goals. In this context, detailed observations of user activities with learning objects and related systems is defined as Attention Metadata, because it is information about the attention a user can give to material she works with, for how long she works with it, when the action took place and what was her opinion about it (extracted from annotations, tags and rating information), etc.

The collected observations can be generalized into behavioural patterns. Behavioural patterns describe in general how a user handles information, e.g. which activities she carries out with them. The comparison of behavioural patterns from various users allows clustering similar users. Based on such clusters, we expect to be able to precisely predict future steps and goals of a user.

Our approach aims to utilize contextualized attention metadata to capture behavioural information of users in learning contexts that can be used to deal with the information overload in user centric ways. We introduce a schema for collecting Contextualized Attention Metadata (CAM). The Contextualized Attention Metadata schema (CAMs) is designed to enable collecting observations about the attention users give to content and about their contexts. CAMs enables the correlation of the observations, thus reflects the relationships that exists between the user, her context and the content she works with.

However, capturing observations about the attention of users requires solving several problems: Applications need to be developed that capture the observations of the user attention. Such applications need to be integrated into the user daily working environment without disturbing or interrupting her. Moreover, the applications need to capture the observations from existing applications of daily use, e.g. the Microsoft Office Suite, Web Browsers, server logs. Each of these tools continuously provides observations, thus generating a stream of information on the interactions of the user with the respective application.

Furthermore, user observations need to be captured in a generalized format that allows merging and processing of the various streams of observations. By merging the observations, it is possible to contextualize each observation, e.g. by identifying which activities the user carried out simultaneously or within a short time-span. For example, with which keywords the user found relevant documents which she really wanted. We therefore speak of contextualized observations of the user attention. We broaden our notion of context by describing the context through all additional information (e.g. information about parallel activities) available at the time the user activity is taking place.

The work described here is published in

> Martin Wolpers, Jehad Najjar, Katrien Verbert and Erik Duval, Tracking Actual Usage: the Attention Metadata Approach. International Journal Educational Technology and Society (ISSN: 1436-4522 (online) and 1176-3647 (print)), Special Issue on "Advanced Technologies for Life-Long Learning", 2007

5.2 Components of the Contextualized Attention Metadata schema (CAMs)

Figure 13 shows the CAM schema developed to allow tracking user activities across different systems.

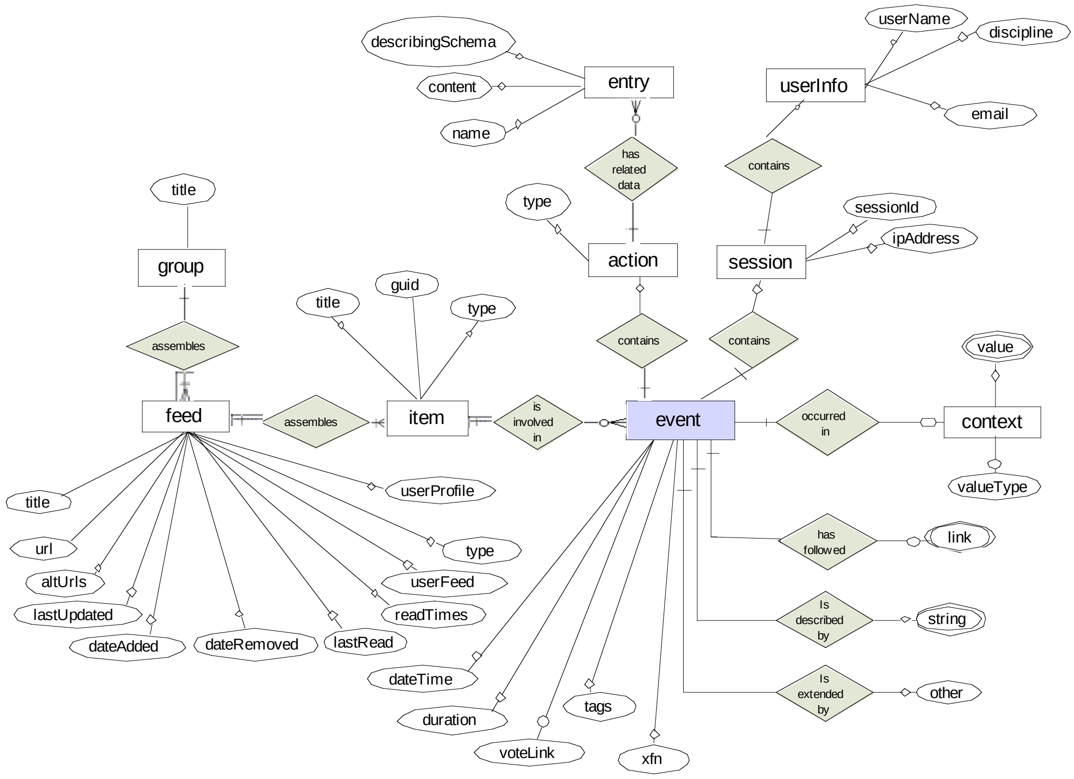

Figure 13: The CAM schema elements (in ER notation)

The CAM schema is designed to allow for tracking user activities in all systems she may interact with while working with documents. As shown in Figure 13, CAMs collects the attention of one user in all systems within one *group* element. The purpose of the *feed* element is to group the attention of the user in one specific system. The *item* element collects the attention given to one specific digital document. Because of the fact that every digital document may be accessed on different occasions and be involved in different tasks (reading, editing, updating, listening to, etc.), it requires capturing information related to every event in which the document was involved in. Each *item* (e.g., document) may be involved in one or more event that has different relevant information. For example, in one event, the document can be edited in one system and afterwards, in another event, the same document can be read or updated. Note that the same document may also be accessed in another system. Each time a document is accessed, more attention metadata is collected, like access date/time, context, duration spent on working with the document and data actively created by the user (ranking, annotations and tags). A teacher may use the same document in her online courses for two different groups of students and in two different contexts (time, application, and topic). In each course where the document is used, different metadata information can be collected, like the time spent on learning it, the topic (computer, management, etc.) of the course and the teacher evaluation of the usefulness of the document to each group of students (it might be well perceived by computer science students but not by physics students).

Making the *event* element central in CAM schema allows for the identification and relation of information about every event the document was involved in across different systems. As also shown in Figure 13, in the CAM schema the elements *duration*, *voteLink*, *xfn* and *tags* are moved from the item element into the event element. This reordering enables the identification of the duration spent with the document, tags given to the document by the user, social relationship and user experience (*voteLink*; like-dislike information) per event.

We will now describe the *item* element and its sub-elements including the *event* element, The *group* element groups all attention metadata of one user in all applications she may work with (all in one group instance) while the *feed* element groups the attention of one user in one specific tool (each tool in a separate feed instance).

More descriptive information on CAMs' elements is given at: http://ariadne.cs.kuleuven.ac.be/empirical/attention/CAM%20schema_Document_v1.5.pdf

Item: the *item* element groups attention metadata of one document. Each document can be involved in different actions (read, listen to, edit, etc.), in different dates, for different periods, and in different contexts.

The *item* element has three sub-elements that do not change over the different actions and events the document may be involved in; those elements record the properties of the document itself. The data is collected here to recognize and identify the document across different systems and contexts:

- **Title:** the *title* captures a human readable name given to the document, when the document is created or edited. This element is necessary to enable users to easily recognize the document. For example, "Global warming - Wikipedia, the free encyclopedia" is a title of a document about global warming on Wikipedia.
- **GUID:** the *guid* element represents a global unique identifier to the document within a given context. E.g. http://en.wikipedia.org/wiki/Global_warming is the unique identifier that is used to locate the document with the title mentioned above on the internet.

- **Type:** the *type* element holds the MIME technical type of the document. For example, “html” is the correct MIME type of the above document on global warming.

On the other hand, all data about the different events the document may be involved in are grouped in the event element. This data describes the attention given to the document, like the time spent with it, tags attached to the document after reading it, and the context where it was used. Each event related data is grouped in one *event* instance.

- **Event:** the *event* element groups attention metadata of each event the document was involved in. For every event instance the following attention metadata is collected:
  - **Action:** the *action* element provides information on the action that the document was involved in (e.g. if it was inserted into local file system or digital repository, opened in a viewer application, bookmarked or rated in a search in a digital repository, etc.):
    - **Action Type:** the *actionType* element holds the type of action (task) the document was involved in. Its value is normally a reference to a value in the action value space. For example, the URL http://.../Actiontype/insert can be a reference to the insert value if the action was inserting a document into a repository.
    - **Entry:** the *entry* element records data related to the action performed. About the same action one or more entries can be recorded. For example, if the document was found using a query, one instance of the *entry* element can store the query terms used by the user to form her query. Another entry can store the results list of that query. In case of insertion, it also records the name of metadata schema (for example IEEE LOM or Dublin Core) used to index the document. If the action was a chat conversation with another person, this element can store the name of the chat partner in one entry instance and the text of the conversation in another entry instance.

- **Date Time:** the *dateTime* element holds the date and time at which the event took place. This element keeps all timestamps of events where the document was involved.
- **Duration:** the *duration* element records the time spent with the document (in seconds).
- **Session:** the *session* element holds the information that is needed to identify the working session.
  - **Session ID**: the *sessionId* element holds a unique identifier for the session.
  - **IP Address**: the *ipAddress* element holds the IP address of the user computer.
  - **User Info**: the *userInfo* element collects information about the user name, email address and scientific discipline of the user performing the action.

The session information (*sessionId* and *ipAddress*) are used to identify the user throughout the different events and tasks she may interact in with the document. The data about the user is collected per event because the same user may have, for example, different user name, IP address every time she works with the same document. Working with a document from the computer at work or at home may result in different IP addresses for the same user.

- **Context:** the *context* element captures information that describes the environments the user may interact with. For example, information about a course (discipline and description) where a user has uploaded a document. The title and description of a course about Human Computer Interaction in the Blackboard (2006) or Moodle (2006) systems are contextual information about the usage of a document. Data captured here can be extracted from the properties of the courses where the documents is used; each course in Moodle, for example, has a title and description, this data can be used to extract information about the context. This data is essential to identify the different contexts where digital content is used. This element has two sub-elements:

- **Value Type:** the *valueType* element holds a reference to an element of an ontology or taxonomy that describes the discipline as derivable from the value element above. The topics might serve as search terms to identify the appropriate discipline with online services like Swoogle (Ding et al., 2004).
- **Value:** the *value* element holds a free text that is extracted, for example, from the title of the course in applications like Moodle or Blackboard. It describes the topics of other documents involved in working with the recent document, e.g. the topics of all documents involved in a course. This element takes multi string value description entries. Those string entries can be used to express the same value in more than one language.

In addition to the information captured in the above two elements, more contextual information can be extracted from other elements described earlier. For example, *event.dateTime*, event.*action.actionType* and *event.session.userInfo.discipline* are rich contextual information. Such data enable identifying interesting patterns about user attention given to documents. For instance, using the element *event.action.actionType* we know if the user is browsing a webpage, working with PowerPoint slides or listening to music, or may do all at the same time. This data can, for example, enable identifying the songs that the user listens to when working with MS PowerPoint and when browsing the web. Using the *event.dateTime* element it is possible to identify the music a user listens to in the morning from the music she listens to in the evening. In addition, it is also possible to identify the web pages that a user consults when working with PowerPoint slides.

- **Followed Links:** the *followedLinks* element groups the set of URIs included in the document and followed by the user. This can be a link to a relevant webpage of a document that is currently read by the user.
- **XFN:** the *xfn* element tracks the social relationship of the author of the document to the reader consulting the document, if the value of *event.action.actionType* element is read in a web browser application.

- **Vote Link:** the *voteLink* element records the user interest (likes and dislikes). The element can take one of the following values:
  - vote-for: means “I like the document.”
  - vote-abstain: means “I have neutral opinion.”
  - vote-against: means “I did not like the document.”

  The data can be used, for example, to recommend a user with documents that are similar (of similar author for instance) to the documents that a user voted positively. Documents that are similar to the *vote-against* documents can be hidden from the user. More interestingly, this data can be used to rank the document based on the votes of a set of users. If many users voted for one specific document, this means that the document is interesting.
- **Tags:** the *tags* element holds a free text label or keywords that is used to describe the document. For example, attention metadata, user data and user tracking are valid tags for this paper.
- **Description:** the *description* element covers descriptive annotations that might be provided by users to express their experience with the document. It uses the value space of the IEEE LOM (2002) “Description” element which is a multi string value, to allow providing the same information in different languages. This element is useful to collect reviews or descriptive data about user experiences about the read item. Some users are interested in annotations other users provide to digital content.
- **Other:** The *other* element is used to allow providing customised elements that are not covered by this schema.

5.3 Infrastructure

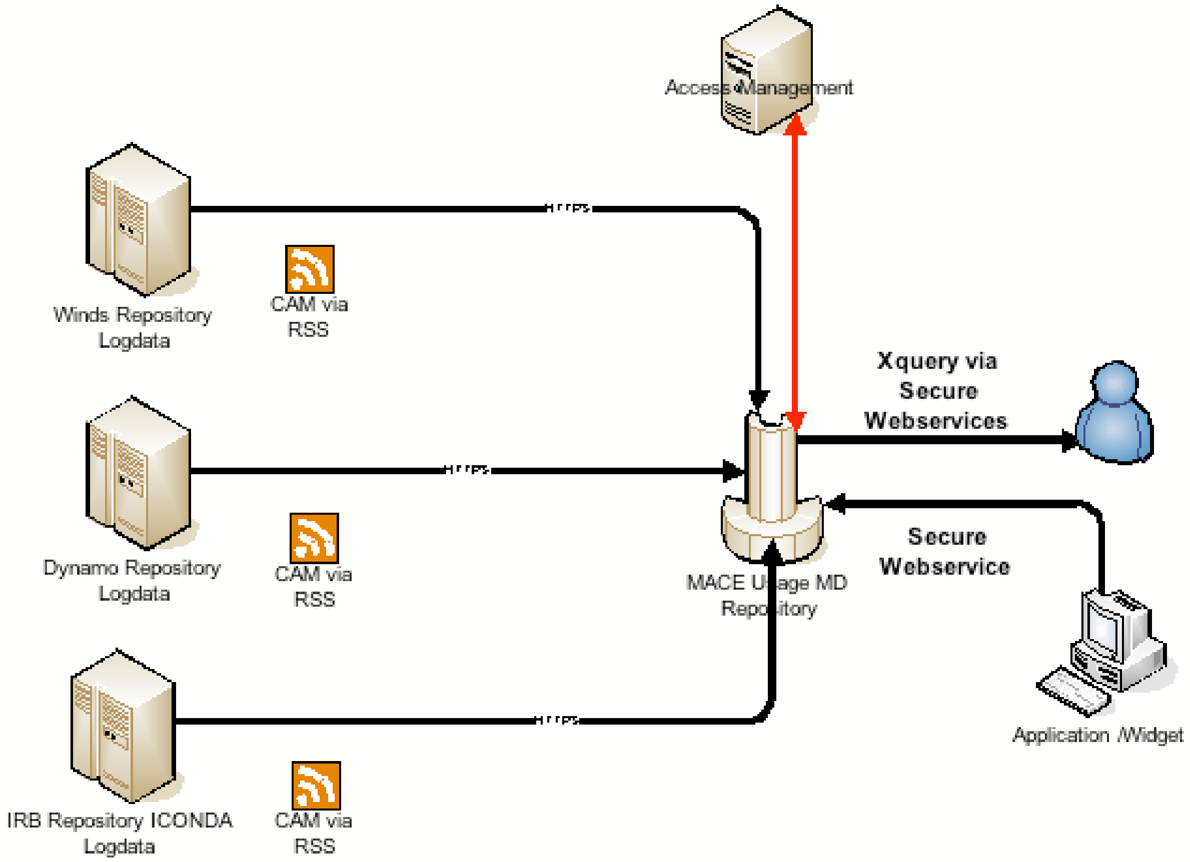

Figure 14: The framework for collecting usage metadata.

Figure 14 above presents the framework that is used to collect and manage usage metadata from applications. Each repository (like WINDS, DYNAMO, ICONDA) publishes CAM instances of its usage metadata via RSS syndication, for example every 24 hours. A CAM stream of every event is represented in one Group element that extends the RSS elements (see the figure below.)

```
<?xml version="1.0"?>
<rss version="2.0" xmlns="http://backend.userland.com/rss2"
    xmlns:cam="http://ariadne.cs.kuleuven.ac.be/empirical/attention#">
  <channel>
    <title>Dynamo Attention</title>
    <link>http://...../Dynamo/</link>
    <description>User attention in Dynamo server.</description>
    <language>en-us</language>
    <pubDate>Wed, 25 Apr 2007 04:00:00 GMT</pubDate>
    <lastBuildDate>Wed, 25 Apr 2007 09:41:01 GMT</lastBuildDate>
    <generator>Dynamo Attention Generator</generator>
    <managingEditor>editor@example.com</managingEditor>
    <item>
      <title>User1 Attention</title>
      <link>http://..........</link>
      <description>..................</description>
```

```
        <pubDate>Wed, 25 Apr 2007 09:39:21 GMT</pubDate>
        <guid>http://liftoff.msfc.nasa.gov/2003/06/03.html#item573</guid>
        <cam:group>
        <cam:feed>
          <cam:title>Dynamo-attention</cam:title>
           <cam:item>
              <cam:title>House</cam:title>
              <cam:guid>http://....<cam:guid>
              <cam:events>
                 <cam:event>
                   <cam:action>
                            <cam:actionType>search</cam:actionType>
                            <cam:relatedData>
                                        <cam:entry>
                                          <cam:name>querytext</cam:name>
                                          <cam:content>java overering</cam:content>
                                        </cam:entry>
                            </cam:relatedData>
                   </cam:action>
                   <cam:dateTime>2007-04-22T09:54:34</cam:dateTime>
                   <cam:session>
                            <cam:sessionId>-1745540467</cam:sessionId>
                                  <cam:ipAddress>134.58…..</cam:ipAddress>
                                  <cam:userInfo>
                                        <cam:userName>Najjar</cam:userName>
                                  </cam:userInfo>
                            </cam:session>
                      <cam:event>
                 </cam:events>
                </cam:item>
          </cam:feed>
    </cam:group>
    </item>
       <item>
        <title>User2 Attention</title>
        <link>http://..........</link>
        <description>..................</description>
        <pubDate>Wed, 25 Apr 2007 09:39:21 GMT</pubDate>
        <guid>............</guid>
        <cam:group>attention of another event</cam:group>
       </item>
       <item>….
      </channel>
</rss>
```


The RSS CAM instances of attention events from all repositories and tools are first stored in a temporarily database as outlined in Figure 15. The attention merger that is build above the central attention repository classifies the events and stores them under the appropriate user feed. In this way, all user attention streams from all applications are grouped into one CAM attention stream.

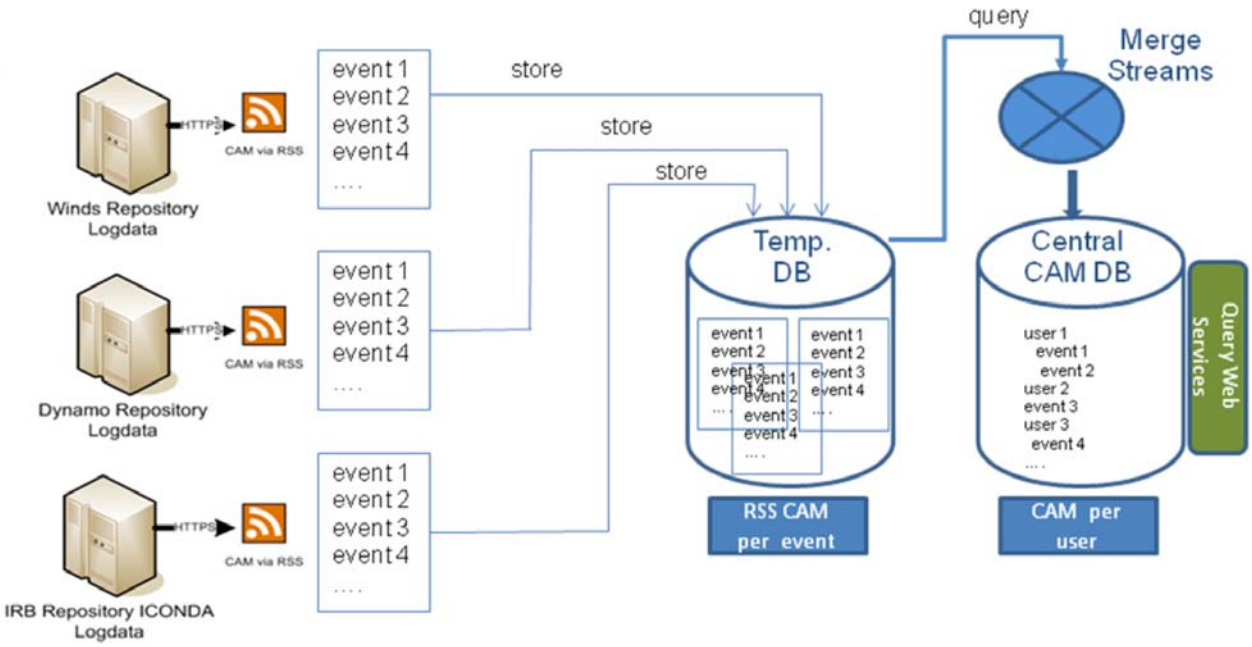

Figure 15: RSS CAM instances and user attention streams.

Applications like recommender systems, contextualized search engines and user modelling tools can access the attention repository, using the Query Web Services, to get relevant usage information about learning objects or users. For example, the following queries can be sent to the attention repository to get usage data about one learning object:

- When was the last time an object used by users?
- Number of times the object is already used?
- In what context the object is used?
- What is the ranking assigned to the object by different users?
- What tags have users attached to the object?
- What annotations users attached to the object?
- Etc.,

Other usage related information about users can also be retrieved:

- What are the common objects used by the user?
- What are the last 10 objects used by the user (see XQuery code below)?
- What are the common search terms and used by the user?
- What are the common keywords that represent the user?

- What is the average time spent by a user in specific application?
- Etc.,

5.4 Summary

This report discloses a schema and framework that will be used to collect and manage usage metadata from different applications of MACE project. The presented schema allows capturing of contextual information about the usage of learning objects. Detailed usage data about every event for the object involved in it can be tracked and then sent to a central attention metadata repository.

The collected usage metadata can be used for several purposes like,

- Build user profiles that are based on previous activities of a user
- Update metadata records of learning objects with descriptive information about the real usage made of the object.
- Social filtering and recommender systems can access the usage data to collect information that allows providing the user with relevant objects.
- Build user communities based on the objects they previously used.
- Etc.,
- Provide stack holders of the project with statistical information about user activities AND the real usage made of learning objects and tools.

Working with confidential data about user behaviour requires developing services that guarantee user privacy. It is understood that MACE will only collect and store usage data necessary for the queries and will not provide access to confidential data for third parties. Within MACE only anonymised data will be used. MACE will take precautionary measures to ensure that no personal data will be revealed to unauthorised persons. Wherever personal data will be collected, users will be informed what kind of data is being collected and how it is ensured that this data is not assignable to a person. Users will (on request) gain access to the data. Personal profiles will not be stored on a central repository, but instead in local repositories with full user control.

6 Conclusion

This deliverable presents for each enrichment work package metadata attributes that will be used for content enrichment in MACE. As can be seen from the deliverable, the work packages overlap at several points, for example usage and context metadata sometimes use the same attributes. The same holds true for context and domain metadata. In parallel, end users have described the way architects work and learn in their field, the results of which can be seen in work package 2. Ideas and suggestions from their side have been included in this deliverable as well.

The basis for metadata enrichment forms the content and domain metadata, which is provided in several formats using various metadata schemas. MACE has developed an application profile in order to harmonize the metadata descriptions and unlock the repositories in MACE. The work presented in this section results from the integration of available metadata provided from the MACE providers: DYNAMO, WINDS and ICONDA agreed to map their contents into LOM v1.0

We then introduced the general concept of context and introduced our triangular concept of the real world, virtual representations of it and metadata of it both, with special regard to context metadata. This kind of metadata is a wide and diffuse field, which can hardly ever be caught in its full spectrum. But we were able to identify some subtypes of context, which are especially of interest for MACE, like: Architectural context, Physical context, Social, Usage and Role context and Technical context.

Additionally we presented MACE's three primary entities (objects/subjects, users and contents) and discussed relations between the various context types and the primary entities. Accordingly we analyzed how incomplete metadata of each kind can be completed and enriched. In succession to that we presented our plans for integrating foreign databases of vastly different nature and described how all data can be brought into relation with each other.

In the next section "Competences and Competence Metadata" were defined. The special focus was on the competences of the building engineer or architect. Heavily based on this view we presented the MACE Competence Metadata Schema, which consists of Mandatory, Core and Optional competencies. Both, Core and Optional competencies can be weighted differently in an educational context. We thus extended our view to Generic, Extended and Specific competencies and defined and modelled the learning process.

"Taxonomy of Usage related/social metadata" is metadata gathered from the former activities of users. By gathering such information a system is able to adapt to different users and to predict future user needs, thus further enhancing information retrieval and learning processes.

The technical problems of collecting CAM include the development of tools that unobtrusively integrate into existing applications, the abstraction generated usage metadata and making the data available. Therefore we presented a schema for CAM and how we plan to capture such data. We also explained the required RSS syndication infrastructure.

The consortium will now concentrate its efforts on harmonizing the overall metadata structure and take into consideration the abilities, needs and requirements from all partners and future end users when creating the integrated MACE infrastructure.

7 Publications

Duval, E. (Ed.): 2005, *IEEE Standard for Learning Object Metadata 1484*.12.1-2002, IEEE 2002

Najjar Jehad, Wolpers Martin and Duval Erik. 2006a. *Attention Metadata: Collection and Management*, WWW2006 workshop on Logging Traces of Web Activity: The Mechanics of Data Collection, Edinburgh, Scotland, May 23, 2006

Najjar Jehad, Wolpers Martin and Duval Erik. 2006b *Towards Effective Usage-Based Learning Applications: Track and Learn from User Experience(s)*, IEEE International Conference on Advanced Learning Technologies (ICALT), Kerkrade, The Netherlands, July 5-7, 2006

OAI: 2002, *Open Archives Initiative Protocol for Metadata Harvesting*, Protocol Version 2.0 of 2002-06-14

8 References

Akman, V.1999. *Strawson on intended meaning and context*", in: [Bouquet et al. 1999].

AttentionXML, AttentionXML specifications, http://developers.technorati.com/wiki/attentionxml

Braun, S. and A. Schmidt (2007). Expert Finding as Informal Learning Support: Competency-Aware and Socially-Aware. First International ExpertFinder Workshop (EFW'07).

Bunt, H. 1994. *Context and Dialogue Control*", in: Think Quarterly, 3(1): 19-31, 1994.

Cheetham, G., & Chivers, G. (2005). Professions, Competence and informal Learning. Northampton: Edward Elgar Publishing.

Collins 1999. Collins English Dictionary, Millennium Edition, HarperCollins Publishers, Glasgow, 1999.

Contextualized Attention Metadata Management http://ariadne.cs.kuleuven.ac.be/emirical/

Croon. 1998 *Reframing the Notion of Context in Information Systems Research*, in: N.J. Buch et al. (eds.), Proc. of IRIS 21, Dep. Of Computer Science, Aalborg University, 1998.

Dey, A. K. (2001). Understanding and Using Context. Personal Ubiquitous Computing, Springer-Verlag. 5: 4--7.

Ericsson, K. A., Charness, N. 1997 *Cognitive and Developmental Factors in Expert Performance*, in [Feltovich et al. 1997].

Feltovich, P. J., Ford, K. M. & Hoffman, R. R. (Eds.). 1997.*Expertise in Context - Human and Machine*, AAAI Press, Menlo Park, California, 1997.

Heidegger, M. 1962. *Being and Time*, Harper & Row, New York, 1962.

HR-XML Consortium. (2006). Completed Specifications: Competencies. from http://ns.hr-xml.org/2_4/HR-XML-2_4/CPO/Competencies.htmlIEEE-LTSC (2002). IEEE Standard for Learning Object Metadata (Draft). IEEE Standard 1484.12.1.

IMS Global Learning Consortium. (2005). IMS Reusable Definition of Competency or Educational Objective Specification. 2006, from http://www.imsglobal.org/competencies/index.html

Klemke. R. 1999. *The Notion of Context in Organisational Memories*, in: CONTEXT-99 - 2nd International and Interdisciplinary Conference on Modeling and Using Context, Trento (Italy), September 9-11, 1999.

Klemke. R. 2000. *Context Framework - an Open Approach to Enhance Organisational Memory Systems with Context Modelling Techniques*, in: PAKM2000: Third International Conference on Practical Aspects of Knowledge Management, 30.-31. October 2000, Basel, Switzerland.

Nardi, B. A. (ed.). 1996a. *Context and Consciousness – Activity Theory and Human-Computer Interaction*, MIT Press, Cambridge, Massachusetts, 1996.

Nardi, B. A. 1996b. *Activity Theory and Human-Computer Interaction*, in [Nardi 1996a].

Oliver, R. (1999). Exploring strategies for on-line teaching and learning. Distance Education, 20(2), 240-254.

Oliver, R. (2001). Seeking best practice in online learning: Flexible Learning Toolboxes in the Australian VET sector. Australian Journal of Educational Technology, 17(2), 204-222.

Oliver, R. & Herrington, J. (2001). Teaching and learning online: A beginner's guide to e-learning and e-teaching in higher education. Edith Cowan University: Western Australia.

Penco, C. 1999. *Objective and cognitive context*, in: [Bouquet et al. 1999].

Petrelli, D., E. Not, et al. (2001). Modelling and Adapting to Context. Personal and Ubiquitous Computing. 5: 20-24.

Schilit, B., N. Adams, et al. (1994). Context-Aware Computing Applications. {IEEE} Workshop on Mobile Computing Systems and Applications. Santa Cruz, CA, US.

Schilit, B. N. and M. M. Theimer (1994). Disseminating Active Map Information to Mobile Hosts. IEEE Network. 8: 22--32.

Simple Publishing Interface, V.0.1: http://ariadne.cs.kuleuven.be/lomi/index.php/SimplePublishingInterface

The Common SQI WSDL Binding (CSWB): http://sourceforge.net/projects/sqi-wsdl/

Webster. 1996. Webster's New Encyclopedic Dictionary, 1996.

Ziemke, T. 1997. *Embodiment and Context*, in: proceedings of hte 2nd European conference on cognitive science, workshop on context, ECCS'97, Manchester, UK, 1997.

Zimmermann, A. and A. Lorenz (2007). An Operational Definition of Context.