COLDEX

Collaborative Learning and Distributed Experimentation

Information Society Technologies Programme

Project number: IST-2001-32327

UNED contribution to Functional Documentation

Metadata Definitions

Deliverable Number:	D.7.2.2
Contractual Date of Delivery:	
Delivery Date:	
Version:	
Lead Partner:	
Contributing partners:	UNED
Authors:	Verdejo, M.F.; Barros, B; Mayorga, J.I.; Velez, J.
Contact:	nmayorga@lsi.uned.es

1 Coldex System Functionality. Metadata definition

The goal of this proposal is to find a minimum consensus for a Learning Object metadata set within the scope of the Coldex Project. This set is aimed to constitute a common ground for annotation and so, making it possible to search for learning objects within the Learning Object Repository (LOR).

Before arriving to this draft set, we have received the partners proposals, which we have had to adapt in order to reach a collection of metadata suitable for all partners. We have used the Learning Object Metadata v1.0 [1] as a reference to guide this effort.

Furthermore, each partner could find necessary to extend this set to fit their own purposes. This is, of course, possible and, for allowing it, the LOR will distinguish different Learning Object types. Each of these types is characterised by a metadata collection, which includes the common set but adds some more to fit the object needs. So, every Coldex Learning Object will incorporate the whole common metadata set as well as the type-specific metadata. Therefore, every tool producing Coldex LOs is supposed to comply with this metadata specification.

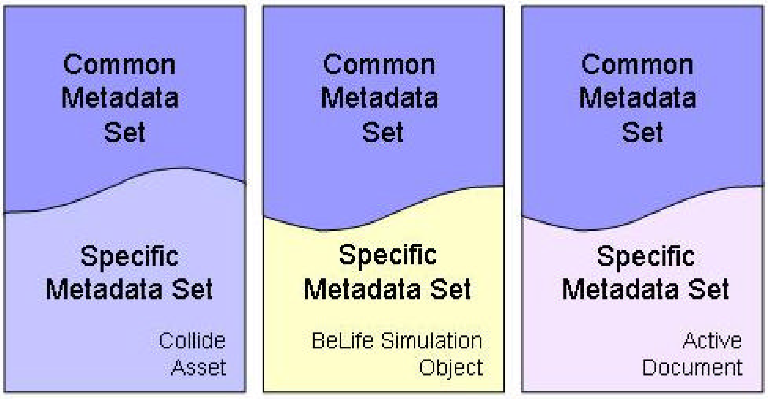


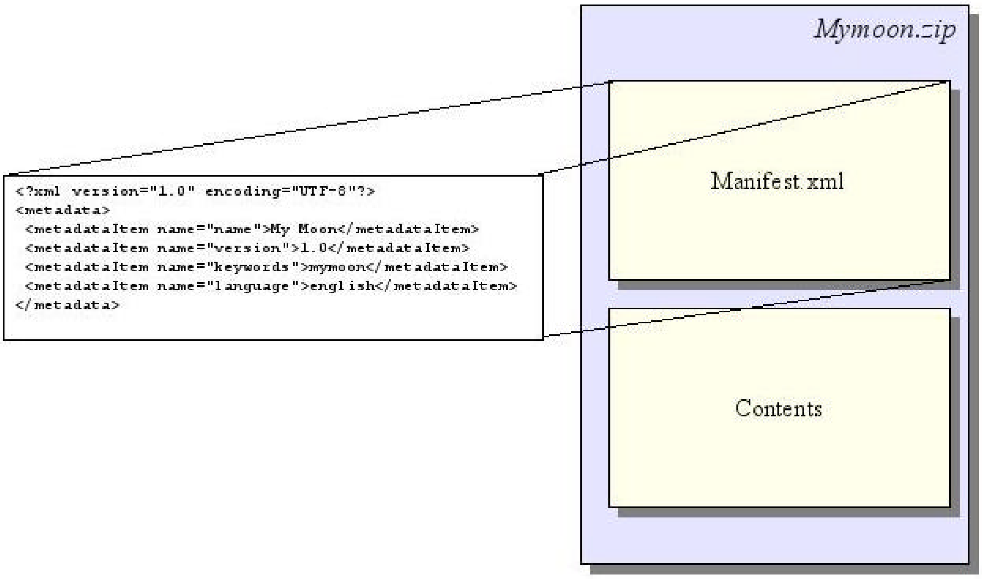
Figure 1 Assembling Metadata Sets

Figure 1 shows how the common and specific metadata sets join to build a Learning Object annotation record for a particular Learning Object Type.

1.1 Learning Object Packaging Model

Most of the knowledge which allows annotating Learning Objects (LOs, for short) comes from the tools. Therefore, a part of the metadata is to be supplied along with the LO so that the LOR is able to extract this information and add it, automatically, to the LO metadata record.

This requirement motivated the definition of a packaging model for LOs. This model describes accurately the way and structure for storing LOs. According to this definition, a Learning Object is packaged into a zip file, which includes both the LO contents and a file called manifest.xml. This file contains the LO meta-information.



```
<?xml version="1.0" encoding="UTF-8"?>
<metadata>
 <metadataItem name="name">My Moon</metadataItem>
 <metadataItem name="version">1.0</metadataItem>
 <metadataItem name="keywords">mymoon</metadataItem>
 <metadataItem name="language">english</metadataItem>
</metadata>
```

Figure 2 LO Packaging model

This solution is not compulsory for Coldex LOs. Nevertheless, complying with this model, the LO author could take advantage of the LOR's auto-filling capabilities, which would result in an easier annotation process. Otherwise, a LO can be represented by any file structure or type (PDF, DOC, TEX, DVI, or any MIME type). Nevertheless, is highly advisable for all Coldex partners to adapt or develop their tools to comply with this packaging model.

1.1.1 Manifest structure

The manifest.xml file is an XML descriptor containing all the tool-supplied meta-information for the described LO. It is made of a collection of metadata items, any of which, in turn is composed of a name and a value (corresponding to the metadata represented by this item). The manifest structure is quite simple (as can be seen in its DTD in Figure 3).

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT metadata (metadataItem+)>
<!ELEMENT metadataItem (#PCDATA)>
<!ATTLIST metadataItem name CDATA #REQUIRED>
```

Figure 3 manifest.xml DTD

Whenever a metadata item allow multiple values, the manifest.xml file should have an entry for each of these values having the same name (see, for instance, the *Keywords* and *Palettes* entries in figure 3).

Metadata items are not mandatory. Missing items won't cause malfunction: only the auto-filling process will not take advantage of them. Figure 4 shows an example of manifest.xml file. Metadata items in Common set have been typed in boldface.

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata>
      <metadataItem name="Name">My Moon</metadataItem>
      <metadataItem name="Version">1.0</metadataItem>
      <metadataItem name="Keyword">coolmodes</metadataItem>
      <metadataItem name="Keyword">mymoon</metadataItem>
      <metadataItem name="Date">2004/04/16 14:26:53</metadataItem>
      <metadataItem name="Language">english</metadataItem>
      <metadataItem name="Type">Collide</metadataItem>
      <metadataItem name="Palettes">Moon</metadataItem>
      <metadataItem name="Palettes">Graphical Calculator</metadataItem>
      <metadataItem name="Palettes">DrawPalette</metadataItem>
</metadata>
```

Figure 4 Example manifest.xml file (for the object mymoon.zip)

1.1.2 Auto-filling

The LOR is able to extract automatically some information from the context to fill in the LO's final metadata record. For doing so, the LOR metadata architecture incorporates a number of software *spy agents* witch in change of inspecting the context and gather some useful meta-information. We distinguish different context types depending on the nature of the information that can be extracted from them. Next we present them in order of precedence:

- Default values, which are obtained from the metadata items defaults
- System environment, which supplies all the system information, such as file size, current date or file name
- Social Context which provides the social information, such as user, group or community data
- Collaborative Context, which contains the information related to the tasks, activities and projects in which the user is involved
- Tool Context, which corresponds to the manifest information

These spy agents retrieve the metadata values in a concrete order. This means there is a relative priority of application among them. Thus, some metadata fields that are filled by a spy can be overwritten by the following ones. For instance, Tool Context metadata values always overwrite the other previously auto-filled metadata record values. Figure 5 show the order of application of these spies.

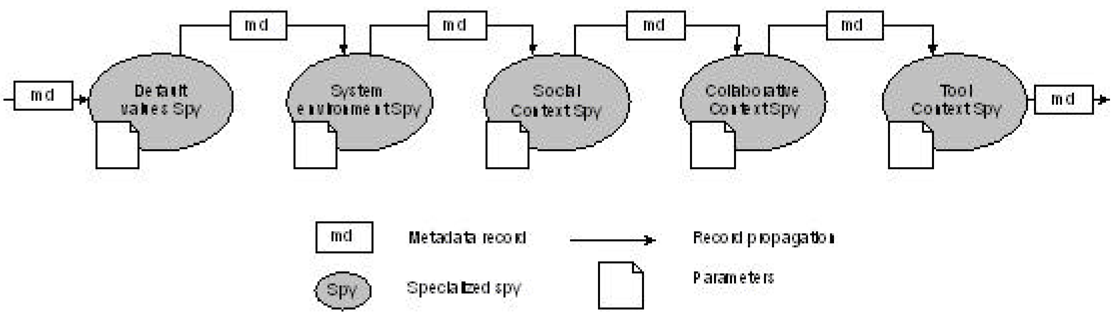
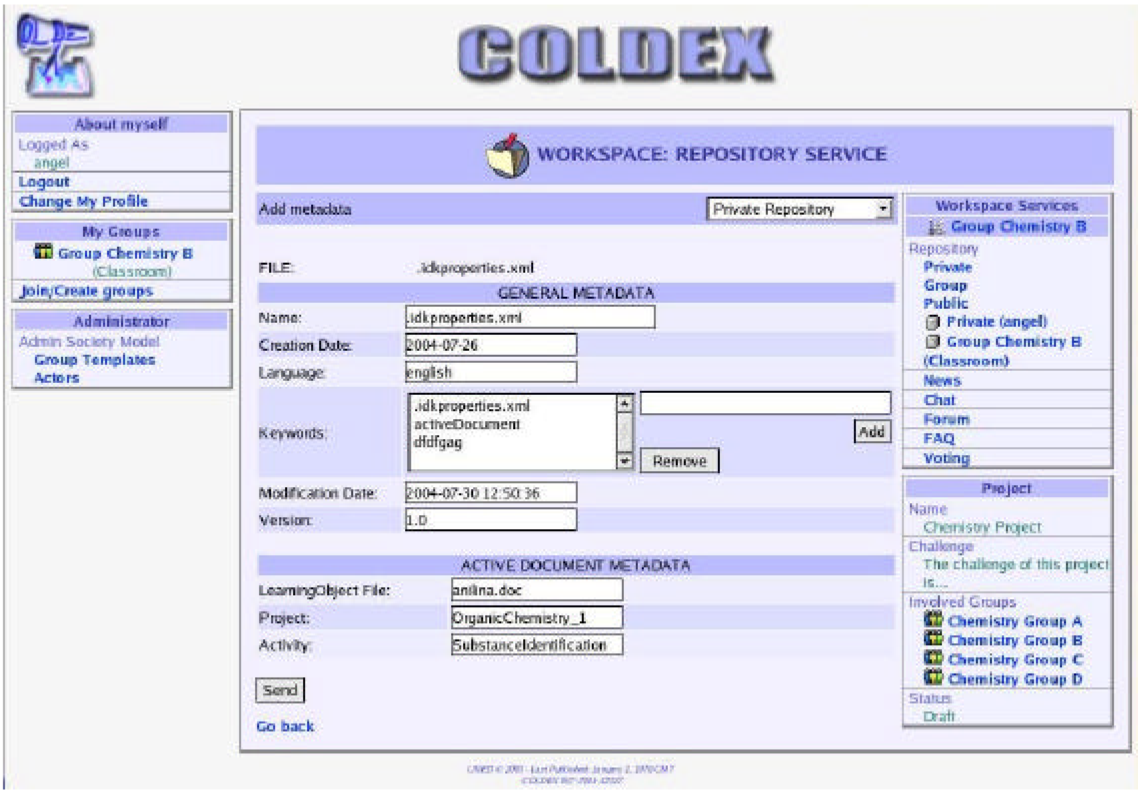


Figure 5 Metadata Spy Agents order of application

1.1.3 Meta-documentation Process

The meta-documentation process starts after the auto-filling task, which supplies initial data for partially populating the metadata record that the user is to validate and complete.

In addition, users can define their own Learning Object Types. For doing so, they must provide the LO type specific metadata schema, which are to extend the common metadata set. When a LO is to be annotated, the system will present the common as well as the type-specific metadata. Figure 6 shows a metadata annotation screen.



Add metadata — Private Repository

FILE: .idkproperties.xml

GENERAL METADATA

Name:	.idkproperties.xml
Creation Date:	2004-07-26
Language:	english
Keywords:	.idkproperties.xml activeDocument dfdfgag
Modification Date:	2004-07-30 12:50:36
Version:	1.0

ACTIVE DOCUMENT METADATA

LearningObject File:	anilina.doc
Project:	OrganicChemistry_1
Activity:	SubstanceIdentification

Project

Name
Chemistry Project

Challenge
The challenge of this project is...

Involved Groups
Chemistry Group A
Chemistry Group B
Chemistry Group C
Chemistry Group D

Status
Draft

Figure 6 Sample Metadata Annotation Screen

1.2 COLDEX Metadata Set

In the following section we present the metadata schema definition for common metadata set (see section 1.2.1) and partner specific ones (see sections 1.2.2, 1.2.3, 1.2.4, 1.2.5 and 1.2.6)

1.2.1 Metadata Set Description

The following elements are given for each metadata item bellow:

- *Metadata*: which is the metadata item identifier (or label)
- *Multiple valued*: it specifies whether the metadata item can accept more than one value

- *Type*: the expected data type for filling in this item. Allowed types are either strings or numbers
- *Vocabulary*: it collects the allowed values for this field (either by listing them or by referring to a standard definition)
- *Required*: it informs of the compulsory character of this metadata item for the described LO
- *Description*: a text defining the item semantics
- *Auto-filling*: the context from which the information for filling in this metadata item can be collected (see auto-filling section)
- *LOM reference*: a reference to the corresponding IEEE LTSC LOM v1.0 metadata field when applicable

1.2.2 Common Metadata Set Definition

Metadata	**1.2.2.1 Id**
Multiple Valued	No
Type	String
Vocabulary	LO Identifier : 'LOR_PREFIX.number'
Required	Yes
Description	This field uniquely identifies the LO within the COLDEX System. This means that this identifier must be unique among all the different LORs. In order to accomplish this, identifiers are auto-generated by COLDEX server as a combination of a prefix that references the LOR plus a local generated number. The Id is a read only metadata field that is to be used to reference a LO from other ones. For instance, valid identifiers would be 'VXU.4368', 'UDUI.324', 'UNED.9867'.
Auto filling	System
LOM reference	1.1.2 Entry

Metadata	**1.2.2.2 Name**
Multiple Valued	No
Type	String
Vocabulary	Open
Required	Yes
Description	The Name of the LO within the repository.
Auto filling	Tool / System
LOM reference	1.1.2 Entry

Metadata	**1.2.2.3 Language**
Multiple Valued	No
Type	String
Vocabulary	ISO 639-2 [4]
Required	No
Description	The LO language (i.e., the language in which the LO contents are written)
Auto filling	System
LOM reference	1.3 Language

Metadata	**1.2.2.4 Description**
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	LO Description (an informative text about the LO and its purposes)
Auto filling	Tool
LOM reference	1.4 Description

Metadata	**1.2.2.5 Keyword**
Multiple Valued	Yes
Type	String
Vocabulary	-
Required	No
Description	These keywords (which are not constrained by any given vocabulary) would allow looking for the LO
Auto filling	Tool / System
LOM reference	1.5 Keyword

Metadata	**1.2.2.6 Version**
Multiple Valued	No
Type	String
Vocabulary	Version Number
Required	No
Description	Version Number (its semantics are given by the author)
Auto filling	Tool / System
LOM reference	2.1 Version

Metadata	**1.2.2.7 Date**
Multiple Valued	No
Type	String
Vocabulary	yyyy/mm/dd hh:mm:ss
Required	No
Description	The LO creation date
Auto filling	Tool / System
LOM reference	2.3.3 Date

Metadata	**1.2.2.8 Type**
Multiple Valued	No
Type	String
Vocabulary	CollideAsset ActiveDocument BeLifeSimulationObject …
Required	Yes
Description	It represents a LO type
Auto filling	Tool
LOM reference	-

Metadata	**1.2.2.9 Mode**
Multiple Valued	No
Type	String
Vocabulary	Collaborative Individual
Required	Yes
Description	The LO development mode (i.e., whether it's been done collaboratively or not)
Auto filling	Tool / Default (Collaborative)
LOM reference	-

Metadata	**1.2.2.10 Format**
Multiple Valued	No
Type	String
Vocabulary	Open. MIME is recommended (see RFC2045 [1], RFC2046 [2])
Required	No
Description	MIME type of the LO contents
Auto filling	Tool / System
LOM reference	4.1 Format

Metadata	**1.2.2.11 Size**
Multiple Valued	No
Type	Number
Vocabulary	-
Required	No
Description	The size, in bytes, of the LO contents
Auto filling	System
LOM reference	4.2 Size

Metadata	**1.2.2.12 Parent**
Multiple Valued	No
Type	String
Vocabulary	LO Identifier (see Id metadata field)
Required	No
Description	It identifies the LO from which this one derives
Auto filling	System
LOM reference	-

Metadata	**1.2.2.13 Relation**
Multiple Valued	Yes
Type	String
Vocabulary	LO Identifier
Required	No
Description	By now, it is the name of another LO within the same repository, which holds a relationship with this one (further work: inter-repository relations)
Auto filling	Tool
LOM reference	7.2.1.2 Relation

Metadata	**1.2.2.14 Group**
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	The group to which the Actor responsible for the LO creation belongs
Auto filling	Tool / Social Context
LOM reference	-

Metadata	**1.2.2.15 Actor**
Multiple Valued	No
Type	String
Vocabulary	The actors declared in the system (Student, Teacher, …)
Required	No
Description	The user stereotype played by the user within the group when adding the LO
Auto filling	Tool / Social Context
LOM reference	-

Metadata	**1.2.2.16 Author**
Multiple Valued	No
Type	String
Vocabulary	User Identifier
Required	No
Description	The LO’s author’s name. This user is responsible for uploads the LO.
Auto filling	Tool / Social Context
LOM reference	-

Metadata	**1.2.2.17 Contributor**
Multiple Valued	Yes
Type	String
Vocabulary	User Identifier
Required	No
Description	Collaborative co-author
Auto filling	Tool / Social Context
LOM reference	-

Metadata	**1.2.2.18 Modifier**
Multiple Valued	Yes
Type	String
Vocabulary	User Identifier
Required	No
Description	The name of each of the LO’s modifiers, i.e., persons who change the LO after its creation
Auto filling	Social Context
LOM reference	-

Metadata	**1.2.2.19 Reviewer**
Multiple Valued	Yes
Type	String
Vocabulary	User Identifier
Required	No
Description	The name of member assessing or evaluating the LO
Auto filling	Tool / Social Context
LOM reference	-

Metadata	**1.2.2.20 Project**
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	The project where this LO is used
Auto filling	Tool / Workspace
LOM reference	-

Metadata	**1.2.2.21 Activity**
Multiple Valued	Yes
Type	String
Vocabulary	-
Required	No
Description	The activity where this LO is used
Auto filling	Tool / Workspace
LOM reference	-

Metadata	**1.2.2.22 Task**
Multiple Valued	Yes
Type	String
Vocabulary	-
Required	No
Description	The task where this LO is used
Auto filling	Tool / Workspace
LOM reference	-

Metadata	**1.2.2.23 Goal**
Multiple Valued	Yes
Type	String
Vocabulary	-
Required	No
Description	A goal which is reached within the instructional process when creating this LO
Auto filling	Tool / Workspace
LOM reference	-

Metadata	**1.2.2.24 Topic**
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	The domain topic or concept to which this LO is related
Auto filling	Tool / Workspace
LOM reference	-

Metadata	**1.2.2.25 Repository**
Multiple Valued	No
Type	(UNED, UDUI, VXU, INESC)
Vocabulary	LOR local unique Identifier.
Required	No
Description	This metadata identifies the LO repository uniquely
Auto filling	Tool / Workspace
LOM reference	-

1.2.3 Specific Metadata Set Definition: Collide Asset

Metadata	**1.2.3.1 Palettes**
Multiple Valued	Yes
Type	String
Vocabulary	-
Required	No
Description	The palette list used by the Collide Asset
Auto filling	Tool
LOM reference	-

Metadata	**1.2.3.2 City**
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	-
Auto filling	Tool
LOM reference	-

Metadata	**1.2.3.3 Country**
Multiple Valued	NO
Type	String
Vocabulary	-
Required	No
Description	-
Auto filling	Tool
LOM reference	-

Metadata	1.2.3.4 ReadOnly
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	-
Auto filling	Tool
LOM reference	-

1.2.4 Specific Metadata Set Definition: BeLife Simulation Object

Metadata	1.2.4.1 ObjectType
Multiple Valued	No
Type	String
Vocabulary	(status, result)
Required	Yes
Description	A discriminator field for distinguishing between Belife-status and results LO
Auto filling	Tool
LOM reference	-

Metadata	1.2.4.2 SimulationSpace
Multiple Valued	No
Type	String
Vocabulary	-
Required	Yes
Description	-
Auto filling	Tool
LOM reference	-

Metadata	1.2.4.3 GreenHouse
Multiple Valued	No
Type	String
Vocabulary	-
Required	Yes
Description	-
Auto filling	Tool
LOM reference	-

Metadata	**1.2.4.4 SimulationTime**
Multiple Valued	No
Type	String
Vocabulary	-
Required	Yes
Description	-
Auto filling	Tool
LOM reference	-

1.2.5 Specific Metadata Set Definition: TREEphoto

Metadata	**1.2.5.1 PartOfTheTree**
Multiple Valued	Yes
Type	String
Vocabulary	-
Required	No
Description	-
Auto filling	Tool
LOM reference	-

Metadata	**1.2.5.2 Spicie**
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	-
Auto filling	Tool
LOM reference	-

1.2.6 Specific Metadata Set Definition: Active Document

Metadata	**1.2.6.1 Resource**
Multiple Valued	No
Type	String
Vocabulary	-
Required	No
Description	-
Auto filling	Tool
LOM reference	-

Annex A. An Annotation Example

This section describes an annotation example. The object, named MyMoon is a Collide asset, so it entails both common metadata and Collide-specific metadata.

Metadata	*Value*
Name	MyMoon
Language	En
Description	A moon photo…
Keywords	Moon
Keywords	Astronomy
Keywords	Space
Keywords	crater
Keywords	geometry
Version	1.1
Date	2004/07/16 10:19:59
Type	Collide
Mode	Individual
Format	application/coolmodes
Size	101376
Parent	-
Relation	-
Group	Astronomy Group
Actor	Teacher
Author	Maria
Version	1.1
Contributor	Bamaria, Niels
Modifier	-
Reviewer	-
Size	99458
Goal	Calculate the crater’s depth
Activity	Astronomy 101
Project	Astronomy Basics
Task	Measuring shadows
Topic	Astronomy
Repository	UDUI
Palette	QocReferenceFrame
Palette	Logic
Palette	DrawPalette
Read-only	False
City	Duisburg
Country	Germany

1.3 References

[1] IEEE Standard for Learning Object Metadata *v6.1 to Final Draft Standard IEEE 1484.12.1-2002 1484.12.1* Learning Technology Standards Committee of the IEEE (http://ltsc.ieee.org)

[2] *Freed, N.; Borenstein, N.* Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies *RFC 2045, November 1996*

[3] *Freed, N.; Borenstein, N.* Multipurpose Internet Mail Extensions (MIME) Part One: Media Types *RFC 2046, November 1996*

[4] *International Standard Organization* Codes for the Representation of Names of Languages *ISO 639-2 http://www.loc.gov/standards/iso639-2/langcodes.html*