

# UPDATER

**Project Acronym:** ODS

**Project Title:** Open Discovery Space: “A socially-powered and multilingual open learning infrastructure to boost the adoption of eLearning resources”

## Open Discovery Space - Updater

**Document revision:** 3.0

**Version:** 14.0.8

**Authors:**

Maria-Cruz Valiente (UAH)

Project co-funded by the European Commission within the ICT Policy Support Programme		
Dissemination Level		
P	Public	
PP	Project Participants	X
C	Confidential, only for members of the consortium and the Commission Services	

WP : WP9

Type <sup>1</sup>: O

**Project Co-ordinator**

*Company name:* INTRASOFT International S.A.

*Name of representative:* Antonis Ramfos

*Address:* Rue Nicolas Bove 2b, L -1253  
Luxembourg

*Phone number:* + 352 44 10 12 2186

*Fax number:* + 352 44 10 12 2359

*E-mail:* Antonis.ramfos@intrasoft-intl.com

*Website:* [www.opendiscoveryspace.eu](http://www.opendiscoveryspace.eu)

---

<sup>1</sup>Deliverable Type: P (Prototype), R (Report), O (Other)

---

## Table of Contents

1	Introduction .....	6
1.1	Definition .....	6
1.2	Requirements.....	6
1.3	The ODS identifiers.....	6
1.4	XML files reading .....	7
1.5	Readable names for Drupal vocabularies .....	12
2	Running the updater script.....	16
	Appendix – Development Issues.....	17

---

## List of Figures

Figure 1: XML files location (harvested data) .....	7
Figure 2: Repositories to import to the ODS portal.....	8
Figure 3: Data files added to the ODS portal.....	8
Figure 4: Invalid data files discarded by the updater .....	9
Figure 5: Log files generated by the updater .....	12

## List of Tables

Table 1: Mapping between LOM fields and ODS Drupal fields .....	10
---	----

## 1 INTRODUCTION

### 1.1 Definition

The updater is an application in PHP which is the responsible to create the educational object nodes for the ODS Drupal using LOM<sup>2</sup>-based XML files generated from harvested content from different repositories. Each XML file represents an educational object node in the Drupal portal.

### 1.2 Requirements

- Drupal 7.22
- Drush 5.8
- MySQL 5.1.6.3
- Apache 2.2.17
- PHP 5.3.5

### 1.3 The ODS identifiers

This version of the updater rests on the premise that the current educational object nodes in the ODS portal could have the new two ODS identifiers that identify a specific educational resource: ODS general identifier (machine name: field\_ods\_general\_identifier) and ODS metadata identifier (machine name: field\_ods\_metadata\_identifier). However, it is mandatory that all the new ones from repositories must include them (files are discarded if they don't contain the ODS identifiers). Therefore, in order to determine if a new educational resource is in the ODS portal yet, the updater follows the next process: For each educational object node stored in the portal, the updater checks if the portal node has the ODS identifiers filled: ODS general identifier (machine name: field\_ods\_general\_identifier) and ODS metadata identifier (machine name: field\_ods\_metadata\_identifier):

- a) If the existing node in the portal has these two identifiers, then the updater checks if they are equal to the ODS identifiers of the new educational resource. If they are equal then the updater updates the information of the portal node with the new content.
- b) If the existing node in the portal doesn't have these two identifiers (i.e. these two fields are empty), then the updater checks if the next ODS educational objects fields are equal to the fields of the new educational resource: LO

---

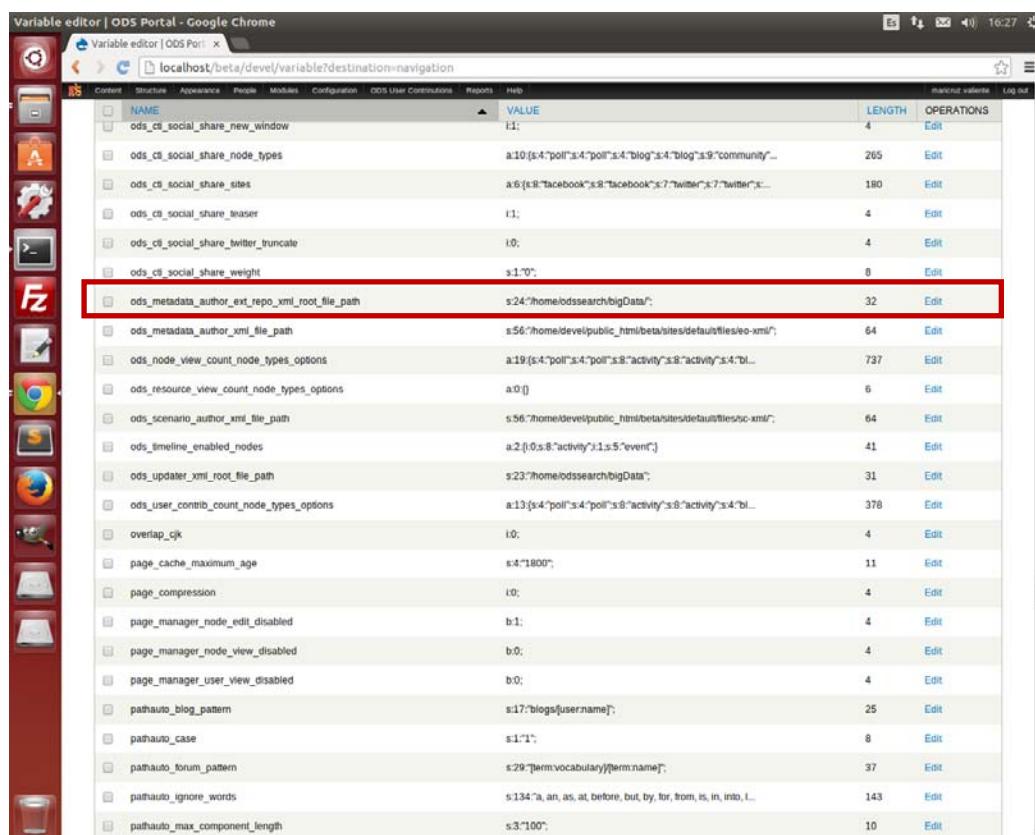
<sup>2</sup> Learning Object Metadata: [http://ltsc.ieee.org/wg12/files/LOM\\_1484\\_12\\_1\\_v1\\_Final\\_Draft.pdf](http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf)

identifier (machine name: field\_lo\_identifier<sup>3</sup>) and Data Provider (machine name: field\_data\_provider), where the Data Provider represents the ID of the repository term included in the *Repository* vocabulary (machine name: repository). If these fields are equal then the updater updates the information of the portal node with the new content.

If the updater doesn't find the educational resource in the portal, then it adds the educational resource as a new node.

## 1.4 XML files reading

The location of the harvested content is defined in the variable 'ods\_updater\_xml\_root\_file\_path' defined in the ODS Drupal portal (*Administration menu > Development > Variable editor*) as shown in Figure 1.



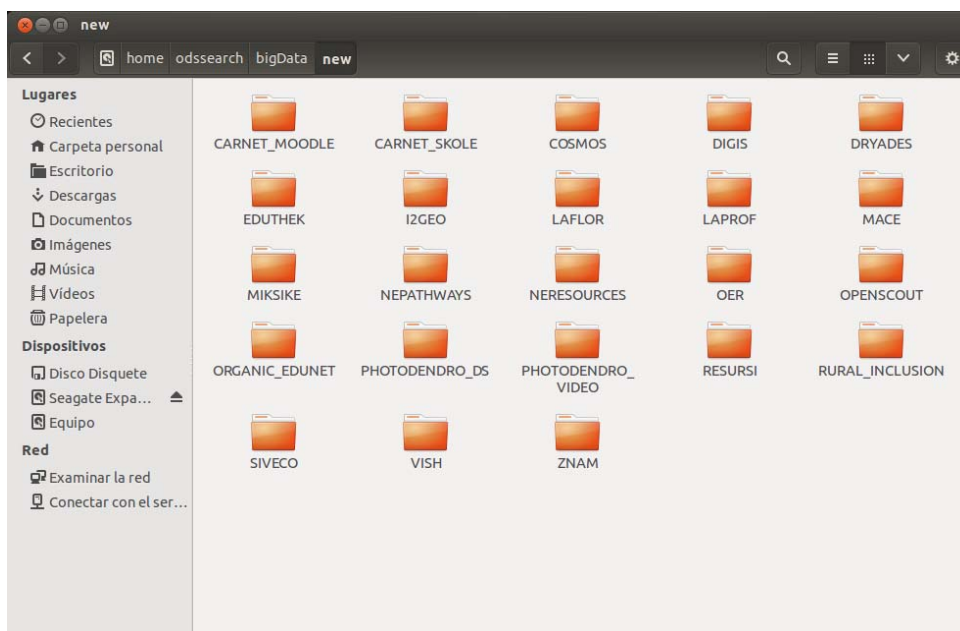
NAME	VALUE	LENGTH	OPERATIONS
ods_cs_social_share_new_window	t1;	4	Edit
ods_cs_social_share_node_types	a:10({s:4:"poll";s:4:"poll";s:4:"blog";s:9:"community..."	265	Edit
ods_cs_social_share_sites	a:6({s:8:"facebook";s:8:"facebook";s:7:"twitter";s:7:"twitter";s:...	180	Edit
ods_cs_social_share_wtaser	t1;	4	Edit
ods_cs_social_share_twitter_truncate	t0;	4	Edit
ods_cs_social_share_weight	s:1:"0";	8	Edit
ods_metadata_author_ext_repo_xml_root_file_path	s:24:"/home/odssearch/bigData/";	32	Edit
ods_metadata_author_xml_file_path	s:56:"/home/dev/public_html/beta/sites/default/files/eco-xml/";	64	Edit
ods_node_view_count_node_types_options	a:19({s:4:"poll";s:4:"poll";s:8:"activity";s:8:"activity";s:4:"bl..."	737	Edit
ods_resource_view_count_node_types_options	a:0[]	6	Edit
ods_scenario_author_xml_file_path	s:56:"/home/dev/public_html/beta/sites/default/files/eco-xml/";	64	Edit
ods_timeline_enabled_nodes	a:2({i:0:s:8:"activity";i:1:s:5:"event"};	41	Edit
ods_updater_xml_root_file_path	s:23:"/home/odssearch/bigData/";	31	Edit
ods_user_contrib_count_node_types_options	a:13({s:4:"poll";s:4:"poll";s:8:"activity";s:8:"activity";s:4:"bl..."	378	Edit
overlap_cjk	t0;	4	Edit
page_cache_maximum_age	s:4:"18000";	11	Edit
page_compression	t0;	4	Edit
page_manager_node_edit_disabled	b:1;	4	Edit
page_manager_node_view_disabled	b:0;	4	Edit
page_manager_user_view_disabled	b:0;	4	Edit
pathauto_blog_pattern	s:17:"blogs[username]";	25	Edit
pathauto_case	s:1:"1";	8	Edit
pathauto_forum_pattern	s:29:"[term:vocabulary]/[term:name]";	37	Edit
pathauto_ignore_words	s:134:"a, an, as, at, before, but, by, for, from, is, in, into, I, ...	143	Edit
pathauto_max_component_length	s:3:"100";	10	Edit

Figure 1: XML files location (harvested data)

For the updater processing, the harvested data will be added in a folder named 'new' (see Figure 2). Each folder included in this path will be managed as a repository and it must contain the different XML files that are part of the repository (e.g. if the location of a specific XML file is /home/odssearch/bigData/new/

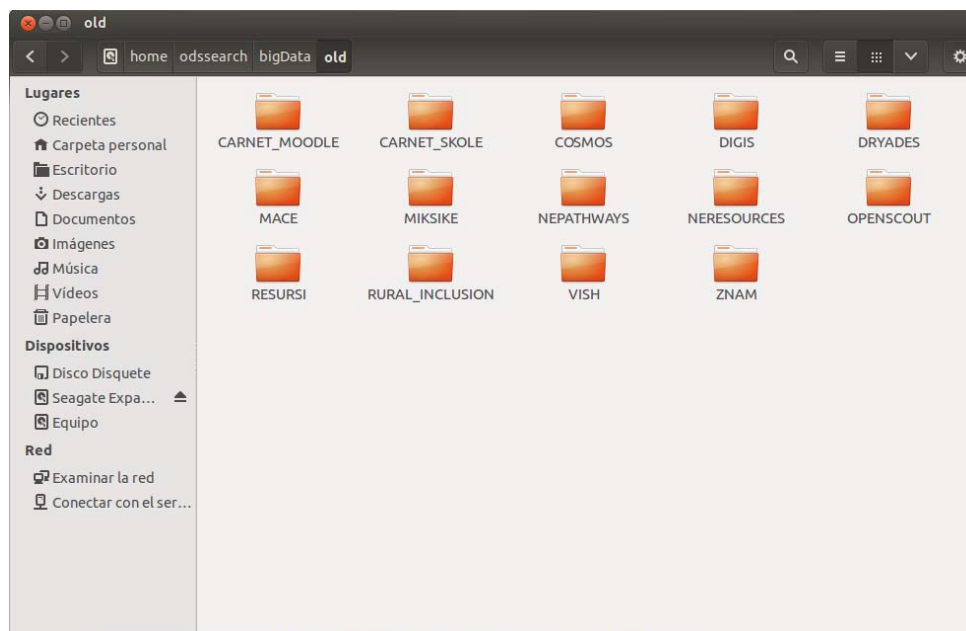
<sup>3</sup> Problem: this field is not mandatory according to the ODS Application Profile, however we need to check this field when we don't have the ODS identifiers. Therefore, this field should be mandatory.

ORGANIC\_EDUNET/ODS\_ORGANIC\_EDUNET\_\_1030.xml, then the repository name that we have to process for this resource will be 'ORGANIC\_EDUNET'<sup>4</sup>).



**Figure 2: Repositories to import to the ODS portal**

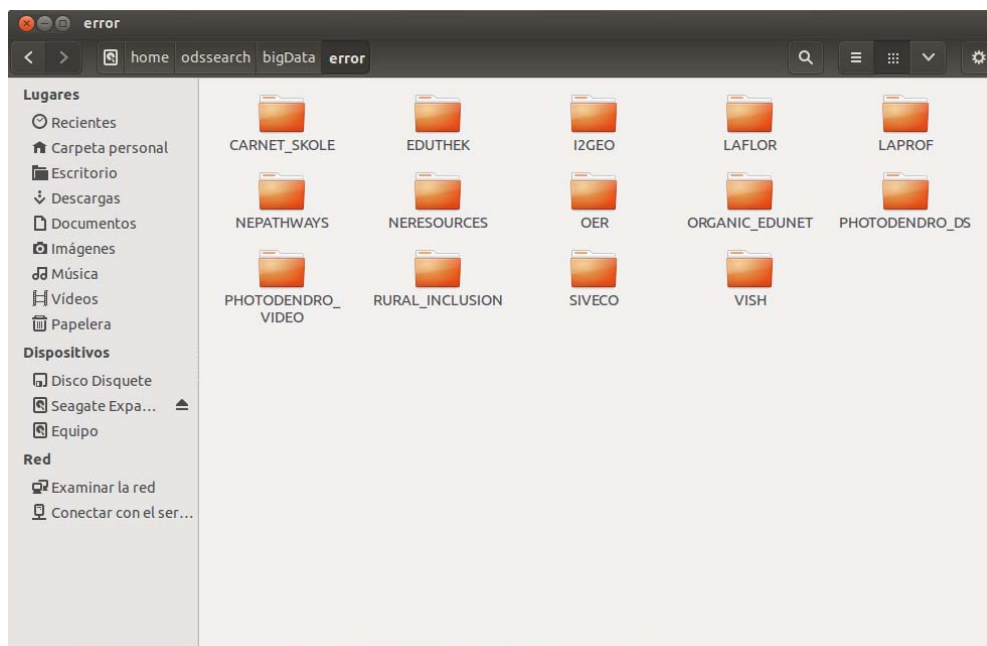
During the execution of the updater each processed file is moved to a folder named 'old' of the ODS updater XML root file path for later data retrieval (see Figure 3), and invalid files are stored in a folder named 'error' of this ODS path (see Figure 4).



**Figure 3: Data files added to the ODS portal**

<sup>4</sup> The repositories.ini file defines the repository names that we have to use in the Drupal portal for the Repository vocabulary (e.g. Organic.Edunet is the drupal vocabulary name for the repository 'ORGANIC\_EDUNET').





**Figure 4: Invalid data files discarded by the updater**

Not all the harvested data are used in the updater for the creation of the educational object nodes. Table 1 illustrates the mapping between the LOM fields that the updater processes from the XML files and the ODS Drupal fields from the 'Educational Object' content type (machine name: educational\_object). It is worth noting that the updater fills the Educational Object's *ODS file location* field (machine name: field\_ods\_file\_location) with the current location of the XML file in a server in order to be able to retrieve this resource from the server for later processing of the whole content.

**Table 1: Mapping between LOM fields and ODS Drupal fields**

LOM field	Educational Object field	Machine Name	Vocabulary
general/identifier/entry	LO Identifier	field_lo_identifier	-
	ODS general identifier (catalog = 'ODS')	field_ods_general_identifier	-
general/title/string (language attribute)	Title	title_field	-
general/language	General Language	field_general_language	ODS AP Languages
	Language	language	-
general/description/string (language attribute)	Educational Object Description	field_eo_description	-
	Body	body	-
general/keyword/string (language attribute)	Edu Tags	field_edu_tags	Edu Tags
general/aggregationLevel/value	Aggregation Level	field_aggregation_level	ODS AP Aggregation Level
lifecycle/contribute/entity	Author Fullname	field_author_fullname	-
lifecycle/contribute/date/dateTime	EO Update Date	field_eo_update_date	-
metametadata/identifier/entry	ODS metadata identifier (catalog = 'ODS')	field_ods_metadata_identifier	-
technical/format	Media Type	field_technical_format	ODS AP Technical.Format
technical/location	Resource Link	field_resource_link	-

LOM field	Educational Object field	Machine Name	Vocabulary
educational/learningResourceType/value	Learning Resource Type	field_learning_resource_type	ODS AP Educational.LearningResourceType
educational/context/value	Educational Context	field_educational_context	ODS AP Educational.Context
educational/typicalAgeRange/string (language attribute)	Educational TypicalAgeRange	field_educational_typicalagerang	-
rights/cost/value	Rights Cost	field_rights_cost	ODS AP Rights.Cost
rights/copyrightAndOtherRestrictions/value	Rights Copyright	field_rights_copyright	ODS AP Rights.Copyright
classification/taxonpath/taxon/entry/string (language attribute)	Classification TaxonPath	field_classification_taxonpath	-
	Classification Discipline	field_classification_discipline	ODS AP Classification.Discipline

As it has been already mentioned, it is important to notice that the repository name of each XML file is the name of the folder in which the file is part of, and the educational object field Data Provider (machine name: field\_data\_provider) will store the ID for the corresponding term in the *Repository* vocabulary (Drupal taxonomy). Besides, it is important to remark that when a vocabulary term included in a field of a XML file is not found in the corresponding ODS AP vocabulary, the file will be rejected since it is non-conformant to the ODS Application Profile. However, in order to avoid the rejection of many XML files, there are some exceptions:

- XML files are not discarded when they have a value not included in the *ODS AP Technical.Format* vocabulary. Besides, the new value is added to the *ODS AP Technical.Format* vocabulary and the *Media Type* field is filled with this value in the Drupal node.
- XML files are not discarded when they have a value not present in the ODS AP Classification.Discipline taxonomy. However, in this case, the value is not added to the ODS AP Classification.Discipline vocabulary and the value is not stored in the *Classification Discipline* field of the Drupal node.

Finally, the updater generates a set of log files according to the pattern *udpater-[repository-name]-[date].log*. Each log file includes information about the files that has been processed for the repository that it is representing. Besides, the updater generates a summary log file (*udpater-summary-[date].log*) that includes information about the results of the import process. The set of log files are located in a folder named 'log' of the ODS updater XML root file path (see Figure 5).

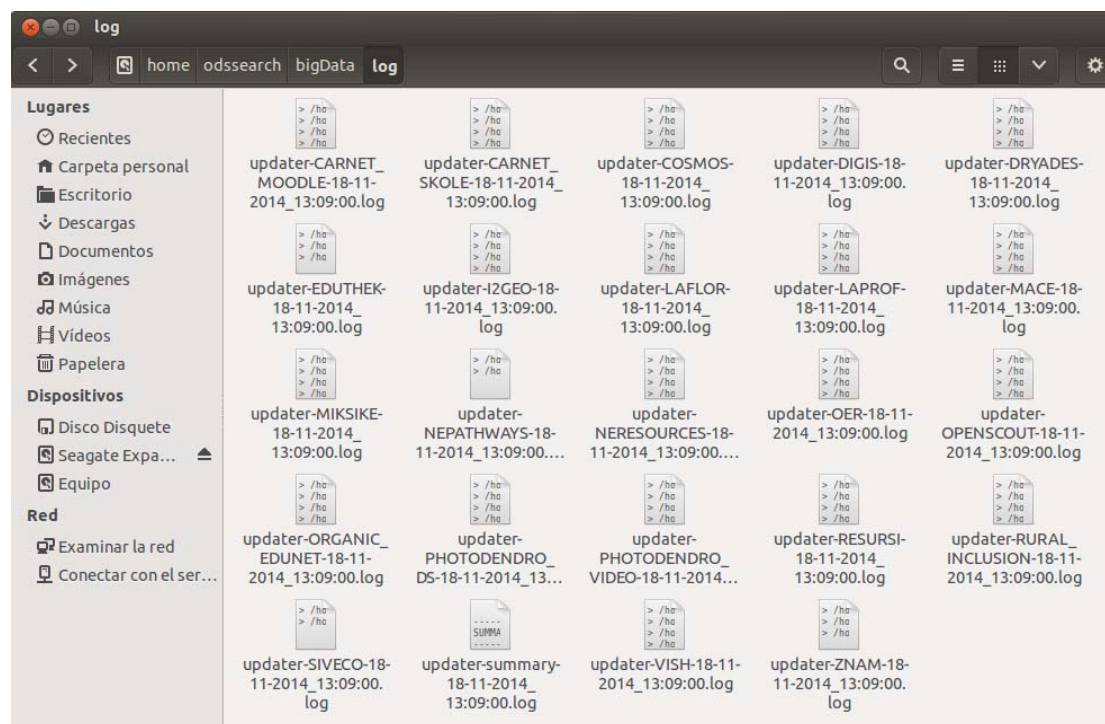


Figure 5: Log files generated by the updater

## 1.5 Readable names for Drupal vocabularies

Aggregation levels, language codes, languages and repository names from the harvested data must be replaced by an existing taxonomy term in the corresponding ODS Drupal vocabulary (*Administration menu > Structure > Taxonomy*). To do this, the updater uses different '.ini' files that contain the valid terms to use in the ODS Drupal vocabularies: *aggregation\_level.ini*, *language\_codes.ini*<sup>5</sup>, *languages.ini* and *repositories.ini*. The '.ini' files must be located in the next path: */path\_to\_drupal\_installation/odsUpdater*. For each .ini file, it is important to check that the corresponding Drupal vocabulary has the same terms than the ini file. Besides, for the languages, if new languages are added the Drupal portal languages configuration has to be updated through the option *Administration menu > Configuration > Regional and language > Languages*.

<sup>5</sup> In order to avoid the rejection of many XML files, the resources are not discarded when they contain an invalid language code which is part of an attribute (language attribute) in the LOM field (e.g. the educational typical age range field).

The current content of the .ini files are the following:

**aggregation\_level.ini:**

```
; Drupal Taxonomy: ODS AP Aggregation Level
; Machine name: ods_ap_aggregation_level
1 = "Educational Object"
2 = "Lesson Plan"
3 = "Educational Scenario"
```

**language\_codes.ini:**

```
bg          = "bg"
cs          = "cs"
da          = "da"
de          = "de"
de-DE       = "de"
Deutsch     = "de"
el          = "el"
en-US       = "en"
eng         = "en"
english     = "en"
en          = "en"
es          = "es"
et          = "et"
fi          = "fi"
fr          = "fr"
ga          = "ga"
hr          = "hr"
hu          = "hu"
it          = "it"
lt          = "lt"
lv          = "lv"
nl          = "nl"
pl          = "pl"
pt          = "pt"
ro          = "ro"
ru          = "ru"
sr          = "sr"
sv          = "sv"
und         = "en"
```

**languages.ini:**

```
; Drupal Taxonomy: ODS AP Languages
; Machine name: ods_ap_languages
bg          = "Bulgarian"
cs          = "Czech"
da          = "Danish"
de          = "German"
el          = "Greek"
en          = "English"
es          = "Spanish"
et          = "Estonian"
```

```
fi          = "Finnish"
fr          = "French"
ga          = "Irish"
hr          = "Croatian"
hu          = "Hungarian"
it          = "Italian"
lt          = "Lithuanian"
lv          = "Latvian"
nl          = "Dutch"
pl          = "Polish"
pt          = "Portuguese"
ro          = "Romanian"
ru          = "Russian"
sr          = "Serbian"
sv          = "Slovenian"
```

### repositories.ini:

```
; Drupal Taxonomy: Repository
; Machine name: repository
CARNET_MOODLE          = "Carnet "
CARNET_SKOLE           = "Carnet "
COSMOS                 = "Cosmos"
COSMOS_HEALTHY         = "Cosmos"
DIGIS                  = "DIGIS"
DIGIS_HEALTHY          = "DIGIS"
DRYADES                = "Dryades"
DRYADES_HEALTHY        = "Dryades"
EDUTHEK                = "Eduthek"
GREEN_OER              = "Green OER Comments"
GREEN_OER_HEALTHY      = "Green OER Comments"
I2GEO                  = "i2geo"
I2GEO_HEALTHY          = "i2geo"
ILDE                    = "ILDE"
LAFLOR                 = "La Flor"
LAFLOR_HEALTHY         = "La Flor"
LAPROF                 = "LaProf"
LAPROF_HEALTHY         = "LaProf"
MACE                   = "MACE"
MIKSIKE                = "Miksike"
MIKSIKE_HEALTHY        = "Miksike"
MOODLE                 = "Moodle Carnet"
MOODLE_HEALTHY         = "Moodle Carnet"
NEPATHWAYS             = "Nepathways"
NERESOURCES            = "Neresources"
OER                    = "OER"
OPENSCOUT              = "OpenScout"
ORGANIC_EDUNET         = "Organic.Edunet"
ORGANIC_EDUNET_HEALTHY = "Organic.Edunet"
PHOTODENDRO_DS         = "Photodendro"
PHOTODENDRO_DS_HEALTHY = "Photodendro"
PHOTODENDRO_VIDEO      = "Photodendro"
```

PHOTODENDRO\_VIDEO\_HEALTHY = "Photodendro"  
RESURSI = "Resursi"  
RURAL\_INCLUSION = "Rural Observatory"  
RURAL\_INCLUSION\_HEALTHY = "Rural Observatory"  
SIVECO = "Siveco LRE"  
SIVECO\_HEALTHY = "Siveco LRE"  
SKOLE = "Skole"  
SKOLE\_HEALTHY = "Skole"  
VISH = "Vish"  
ZNAM = "Znam.bg"  
ZNAM\_HEALTHY = "Znam.bg"

## 2 RUNNING THE UPDATER SCRIPT

As it has been already mentioned, all the harvested LOM-based XML files must be located in the 'new' folder of the ODS updater xml root file path (e.g. /home/odssearch/bigData/new), and the '.ini' and updater.php files must be located in the 'odsUpdater' folder of the Drupal installation path (e.g. /home/devel/public\_html/bet/odsUpdater) for a successful execution of the updater script. To run the updater script, open a terminal, navigate to the odsUpdater directory and type the next command:

```
sudo drush php-script updater.php
```



## Appendix – Development Issues

This section provides a set of remarks about some implementation issues that should be improved according to the LOM-based fields of an Educational Object node in the Drupal portal.

- The updater considers the first general language label of the XML file to be the language of the drupal Educational Object node. If the XML file doesn't contain a general language label, then the updater assigns the default language code "en" (English) to the *language* field of the Educational Object node.
- The updater considers the first string title with the same language of the Educational Object node to be the title of the Educational Object node. If the XML file doesn't contain a title with the same language of the node, in order to avoid empty titles, the updater takes the first title as the title of the Educational Object node (also, in order to avoid inconsistencies, we don't change the language of the drupal node by the language of that title). We assume that at least we will have a valid title in the XML file.
- According to the LOM standard an educational resource may have several entities. However, an Educational Object node defines the *field\_author\_fullname* field as a field with only one value. Therefore, the updater considers the first contribute entity label of the XML file to be the author (contributor) of the Educational Object node.
- According to the LOM standard an educational resource may have several general identifiers. However, an Educational Object node defines the field *field\_lo\_identifier* as a field with only one value. Therefore, the updater considers the first general identifier label of the XML file (with catalog different from "ODS") to be the value of the *LO identifier* field in the Educational Object node.
- According to the LOM standard an educational resource may have several description labels. However, an Educational Object node defines the *field\_eo\_description* field as a field with only one value. Therefore, the updater considers the first description label of the XML file to be the description of the Educational Object node. Also, this description is stored in the *body* field of the Educational Object node.
- According to the LOM standard an educational resource may have several technical location labels. However, an Educational Object node defines the *field\_resource\_link* field as a field with only one value. Therefore, the updater considers the first technical location label of the XML file to be the resource link of the Educational Object node.
- According to the LOM standard an educational resource may have several typical age range labels. However, an Educational Object node defines the *field\_educational\_typicalagerang* field as a field with only one value. Therefore, the updater considers the first typical age range label of the XML file to be the age range of the Educational Object node. On the other hand, the typical age range

field is processed as a string and perhaps it should be better categorized into a real age range for further analysis in the ODS portal (especially for the search functionality: the age range filter).

- The updater considers the entry label of the classification/taxonpath/taxon element to be classifications of the Educational Object node (*field\_classification\_taxonpath*). However, sometimes the value of the taxon entry is in the form “term::term::term...” or “term>term>term...” and this is how the classification category is displayed in the summary page of the ODS portal. Besides, although according to the LOM standard a learning object may have several taxonpath labels, we store in the drupal node only the information (taxons and discipline) of the first taxonpath label.
- The updater considers the last term after the separators “::” in the entry label of the classification/taxonpath/taxon element to be the discipline field of the Educational Object node (*field\_classification\_discipline*). If the taxon entry doesn’t include the separators, the whole text is taken as the discipline.
- According to the LOM standard an educational resource may have several dates regarding different contribute labels. However, an Educational Object node defines the *field\_eo\_update\_date* field as a field with only one value. Therefore, the updater considers the first valid contribute update date of the XML file to be the value of the *EO Update Date* field in the Educational Object node.