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: 2.0

Version: 14.0.7

Authors:

   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 [1]:   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*

---

[1]Deliverable Type: P (Prototype), R (Report), O (Other)

Table of Contents

List of Figures

List of Tables

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[2]-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 udpater 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 is 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, then the updater checks if the next ODS educational objects fields are equal to the fields of the new educational resource: LO identifier (machine name:
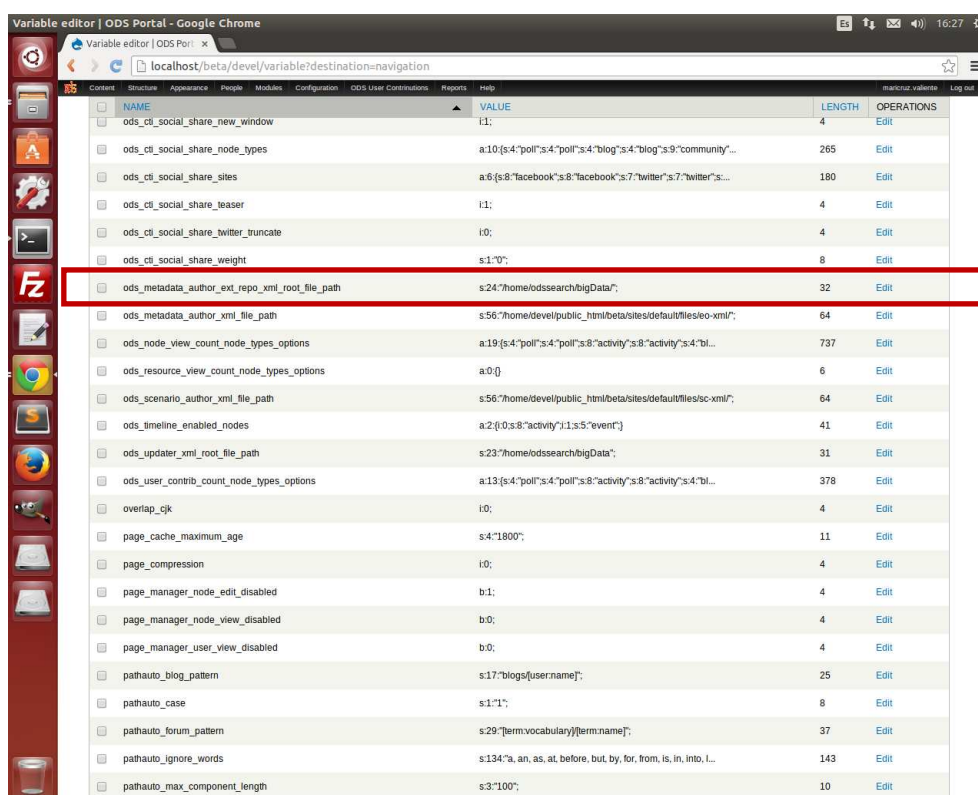
---

[2] Learning Object Metadata: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

field_lo_identifier[3]) 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.

**Figure 1: XML files location**

The harvested data will be added in the 'new' folder of this location for the updater processing (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/ ORGANIC_EDUNET/ODS_ORGANIC _EDUNET__1030.xml, then the repository name assigned to this resource will be 'ORGANIC_EDUNET').

---

[3] 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.
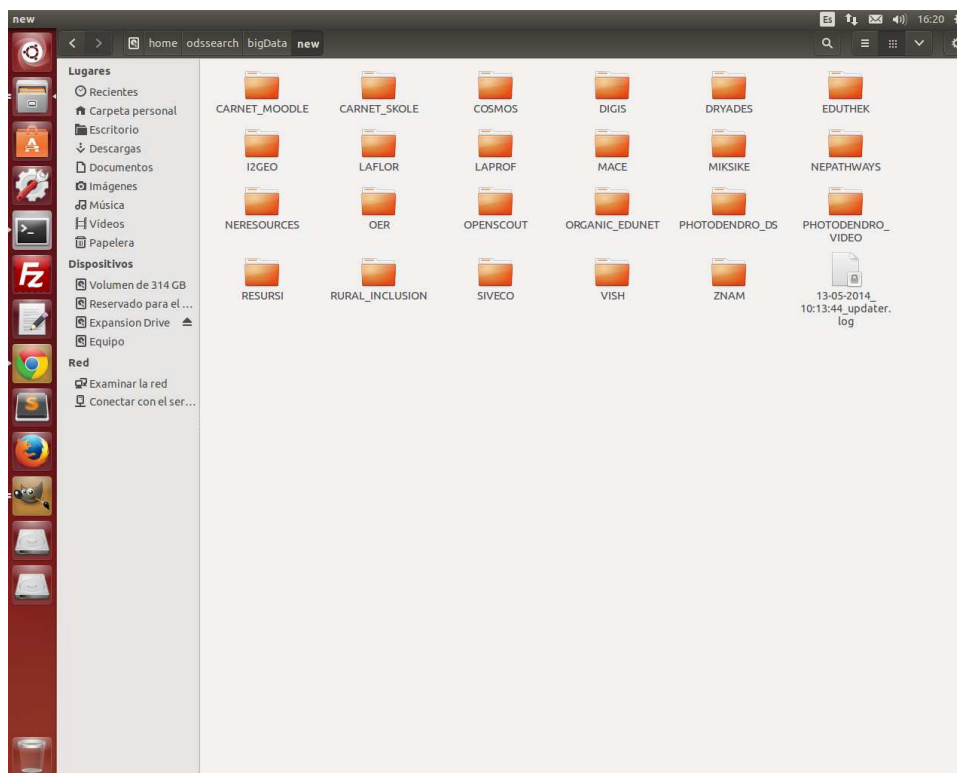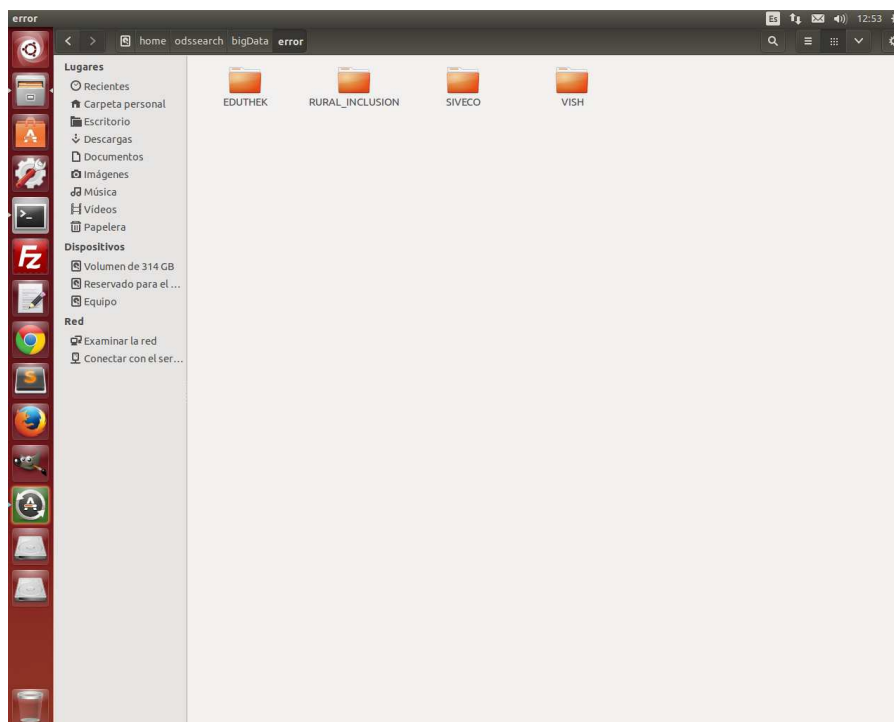
Figure 2: Harvested data to be added to the ODS portal

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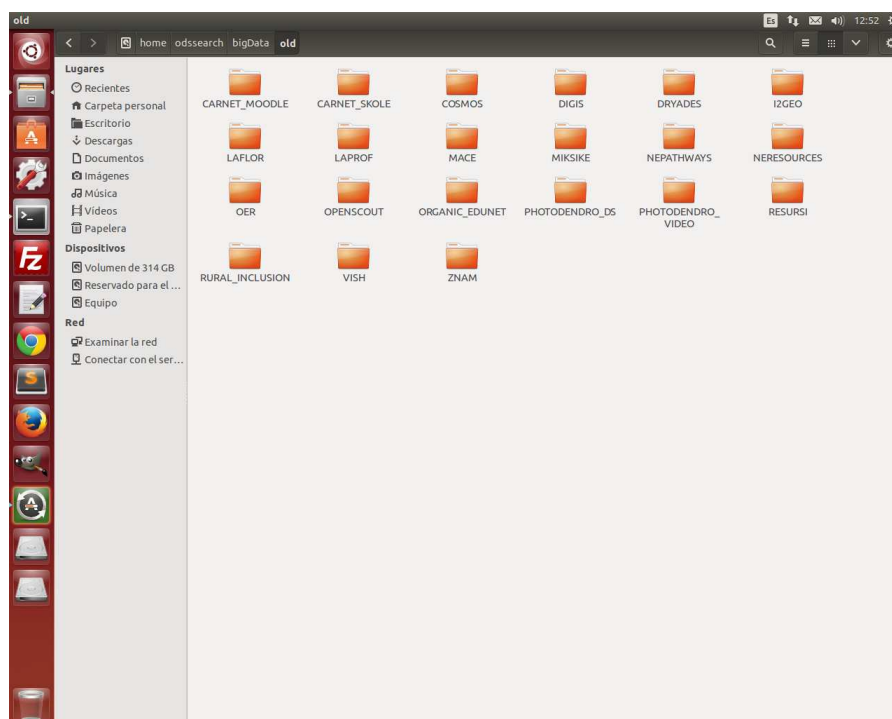
Figure 3: Processed data files added to the ODS portal

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 next ODS Drupal field with the current location of the XML file ('old' folder) in order to be able to retrieve this resource from the server for later processing of the whole content: ODS file location (machine name: field_ods_file_location).

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	-
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.

Besides, it is important to remark that when a vocabulary term included in a field of a XML file is not not found in the corresponding ODS AP vocabulary, the file will be rejected since it is non-conformant to the ODS Application Profile.

Figure 5: Updater log sample

September 2014, v2.0

Finally, once the updater has finished the processing of the XML files it generates a log file which is located in the same path of the harvested data (i.e. '/home/odssearch/bigData/new') with information about the processed XML files. Figure 5 illustrates the content of a log file generated by the updater script.

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, languages.ini, language_codes.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*.

aggregation_level.ini:

```
; Drupal Taxonomy: ODS AP Aggregration 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"
de-DE           = "German"
Deutsch         = "German"
el              = "Greek"
eng             = "English"
english         = "English"
en              = "English"
en-US           = "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"
und             = "Other"
```

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"
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"
PHET                           = "PHET"
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"
SMART                          = "SMART"
TES                            = "TES"
VISH                           = "Vish"
ZNAM                           = "Znam.bg"
ZNAM_HEALTHY                   = "Znam.bg"
```

Open Discovery - Updater

September 2014, v2.0

2 DEVELOPMENT ISSUES

This chapter provides a set of remarks about some implementation issues that should be improved:

- The updater considers the first general language label of the XML file to be the language of the Educational Object node. If the XML file doesn't contain a general language label, then the updater assigns the "en" (English) language code 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 language node, ir order to avoid empty titles, the updater takes the first title as the title of the Educational Object node and it changes the language of the Educational Object node with the language of this first title (when a title doesn't contain the language attribute, the updater assigns the value "und" to the language of this title).

- 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: age filter).

- The updater considers the entry label of the classification/taxonpath/taxon element to be classifications of the Educational Object node. 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, here the problem that we have is that we can have diferente taxons in different taxonpath labels, if they have the same language, we store in the drupal node the taxon of the last taxon label.

- The updater considers the last term after the separators "::" in the entry label of the classification/taxonpath/taxon element to be disciplines of the Educational Object node. If the taxon entry doesn't include the separators, the whole text is taken as the discipline.

3   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