



EUROPEAN COLLECTED LIBRARY OF ARTISTIC PERFORMANCE

www.ECLAP.eu

Grant Agreement No 250481

DE3.3.1

Infrastructure: Content and Metadata processing and semantification

Version: 1.6

Date: 31/10/2012

Project Title: ECLAP
Project Number: ICT-PSP-250481
Deliverable Number: DE3.3.1
Deliverable Type: public
Work-Package contributing to the Deliverable: WP3
Nature of the Deliverable: document
Status: final, closed

Contractual Date of Delivery: 31/12/2011
Approve for quality control by: 31/10/2012
Finally approved by coordinator: 31/10/2012
Actual Date of Delivery: 31/10/2012

Document responsible : Ivan Bruno
Email address: ivanb@dsi.unifi.it

Affiliation acronym: DSI
Authors: <ul style="list-style-type: none">• Ivan Bruno, Michela Paolucci, Pierfrancesco Bellini, Alice Fuzier (DSI)• Nicola Mitolo• Jaap, (BnG)

Revision History: (only for versions approved by the document coordinator or if this action is delegated to someone else)

Revision	Date	Author	Organization	Description
V 0.1	17/11/11	Ivan Bruno	DSI	DE Set up
V 1.0	28/11/11	Ivan Bruno	DSI	Integration
V 1.1	12/12/11	Ivan Bruno	DSI	Integration
V 1.2	23/12/11	Ivan Bruno	DSI	Integration
1.6	31/10/2012	P. Bellini	DSI	Final version

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Catalogue:

Title	Infrastructure: Content and Metadata processing and semantification
Identifier.de	DE3.3.1
Identifier.ISBN	
Creators	Ivan Bruno, Nicola Mitolo, Jaap
Subject	Content and Metadata processing and semantification
Description	Update version of content and metadata processing and semantification, functionalities and procedures
Keywords	Metadata, Working group, Best Practice Network, Performing Arts, Education, Training, Digital Libraries, Tools, IPR
Publisher	ECLAP
Date	31-12-2011
Format	Document
Type	DOC
Language	EN

Citation Guidelines

Author(s) name Surname, Deliverable number, Deliverable title, ECLAP Project, DD/MM/YY, URL: univocally determined on <http://bpnet.eclap.eu>

ECLAP Copyright Notice

Depending on the document's declaration of accessibility on the title page, the following notices apply:

- the document is Private or Restricted to the consortium, the access to the document is regulated by the consortium agreement. Ask ECLAP project coordinator Paolo Nesi for more information via info@eclap.eu.

Please note that:

- You can become affiliated with ECLAP. This will give you access to a great amount of knowledge, information related to ECLAP services, content and tools. If you are interested please contact ECLAP coordinator Paolo Nesi at info@eclap.eu. Once affiliated with ECLAP you will have the possibility of using the ECLAP for your organisation.
- You can contribute to the improvement of ECLAP by sending your contribution to ECLAP coordinator Paolo Nesi at info@ECLAP.eu
- You can attend ECLAP meetings that are open to public, for additional information see www.eclap.eu or contact ECLAP coordinator Paolo Nesi at info@eclap.eu

Table of Contents

1	LIST OF THE ACRONYMS	6
2	EXECUTIVE SUMMARY AND REPORT SCOPE	7
2.1	CONTENT AND METADATA PROCESSING AND SEMANTIFICATION	8
2.2	ECLAP PORTAL USERS AND GROUPS	9
2.3	MULTILINGUAL TAXONOMY	9
2.4	EUROPEANA MODULE	9
2.5	CONCLUSIONS	9
3	CONTENT AND METADATA PROCESSING AND SEMANTIFICATION	11
3.1	ECLAP PORTAL INSTALLATION PARAMETERS/SETTINGS (IVAN BRUNO)	11
3.1.1	Content & metadata web upload Configuration	12
3.1.2	Configuring the Form Submit User Content	13
3.1.3	Configuring the User Content for the inclusion	16
3.1.4	Configuration options	17
3.2	CONTENT & METADATA WEB UPLOAD PAGE	18
3.2.1	Metadata Section	19
3.2.2	Resource Section	20
3.2.3	Workflow type identification	22
3.2.4	Taxonomy Classification	22
3.2.5	Groups Section	22
3.2.6	Terms of Use	23
3.2.7	Metadata checking	23
3.2.8	Production progress	25
3.3	CONTENT & METADATA EDIT PAGE	25
3.3.1	Update Content	25
3.3.2	Change Icon	26
3.3.3	Edit workflow	26
3.3.4	Enrich	27
3.3.5	Validate	27
3.3.6	Ipr Wizard	27
3.3.7	Workflow	27
3.4	METADATA EDITING, METADATA EDITOR (DSI)	28
3.4.1	Metadata Editor Web Page	28
3.4.2	Metadata Editor Enrichment mode	34
3.4.3	Metadata Editor Validation mode	35
3.4.4	Leaving Metadata Editor	36
3.4.5	Metadata Editor Read-Only mode	37
3.5	FUNCTIONALITIES OF THE ECLAP AUTOMATED BACK OFFICE (BRUNO - DSI)	37
3.5.1	Configuration of AXCP Grid Node	37
3.6	CONTENT AND METADATA PRODUCTION (BRUNO - DSI)	38
3.6.1	UploadAXCP	39
3.6.2	UploadAXCP4Device Rule	42
3.6.3	Extract Resource AXCP Rule	44
3.6.4	LD/MD/HD video production AXCP Rule	44
3.6.5	Automatic translation	45
3.7	CONTENT & METADATA INGESTION (BELLINI, BRUNO - DSI)	46
3.7.1	Ingestion Database	47
3.7.2	Content and Metadata Retrieval (Ingestion)	47
3.7.3	Content Generation from Metadata (Harvesting)	47
3.7.4	Content Retrieval	48
3.7.5	Resource Injection	48
3.7.6	Content management (Bruno - DSI)	49

*DE3.3.1– Infrastructure: Content and Metadata processing and semantification
Best Practice Network*

3.7.7	Update Metadata	49
3.7.8	DeleteObject	49
3.7.9	ProduceIcon	50
3.8	CONTENT AND METADATA ANALYSIS (FUZIER – DSI)	50
3.8.1	User to user	50
3.8.2	Content to content	51
3.8.3	Content to user	53
4	ECLAP PORTAL USERS AND GROUPS (MICHELA PAOLUCCI - DSI)	55
4.1	CONTACT INGESTION	55
4.1.1	Import contacts from e-mail providers and Social Networks.....	55
4.1.2	Import contacts from file	55
4.1.3	Import contacts view	56
4.1.4	Dependencies	56
4.2	CONTENT MANAGEMENT IN GROUPS	57
4.2.1	View the contents.....	58
4.2.2	Filter the contents.....	58
4.2.3	Make actions on the content	59
4.2.4	Dependencies	59
5	ECLAP USER MANUAL	61
6	MULTILINGUAL TAXONOMY	62
6.1	PURPOSE & SCOPE	62
6.2	SOFTWARE ARCHITECTURE.....	62
6.3	PROCEDURES FOR CREATING & UPDATING TAXONOMIES.....	62
6.4	FUNCTIONALITIES.....	63
7	EUROPEANA MODULE	67
7.1	FUTURE PLANS	68
8	BIBLIOGRAPHY	69

1 List of the Acronyms

The following acronyms will be used in this deliverable:

Acronym name	Explanation
BPN (Best Practice Network)	ECLAP is a BPN. The ICT Policy Support Programme Competitiveness and Innovation Framework Programme 2010 of the European Commission Best Practice describes a BPN as an instrument that aims to “promote the adoption of standards and specifications for making European digital libraries more accessible and usable by combining the "consensus building and awareness raising" function of a thematic network with the large-scale implementation in real-life context of one or more concrete specifications or standards by its members.”
DE (Deliverable)	The acronym DE stands for deliverable - the reports and objects that will be produced during the ECLAP project.
DoW	Description of Work of ECLAP, in which the work packages, deliverables and the scope of the project are described.
M (Project month)	The ECLAP project started in July 2010 (M1) and will finish in June 2013 (M36). In this deliverable, the project months are specified in which certain tasks and deliverables will be completed. For instance, if a deliverable is due in December 2011, this is M18 of the ECLAP project.
NMIUG	Minimum target value of members that have to be involved in the UG (minimally 250 in year 3 of the project, ideally 350, as defined in DE1.1. ¹).
Target Users	Classification of the users that can be interested in ECLAP, based on socio-demographical variables of a segment of. Starting from the definition provided in DE2.1.1. ² the identification of ECLAP target users is extended in this deliverable.
UG	A group of experts that will function as a source for testing and validating the results of the ECLAP portal development. Also, the members of the UG will receive updated information about the project development. The UG will consist of experts representing the different users of ECLAP, including people to assess accessibility issues.
UGC	User generated content: Catch-all term for any comment, tag or other description as well as Content, Previews or Metadata; or comments, tags and descriptions thereof that is provided by End Users to ECLAP. ³
UGM	User Group Member
WP (Work Package)	ECLAP consists of seven Work Packages, in which the various partners carry out specific tasks.
WG (Working Group)	A WG is an interdisciplinary collaboration of researchers working on new research activities. ECLAP consists of three Working Groups: - Working group (A) on Performing Arts Education and Training - Working group (B) on Intellectual Property and business models for content - Working group (C) on Best Practices Tools for performing art digital libraries
AXCP (AXMEDIS Content Processing)	Backoffice platform based on AMXEDIS content processing grid infrastructure

¹ Tzouvaras, V. et al., 6 October 2010. DE1.1 Assessment and Evaluation Manual. ECLAP project.

² Baltussen, L.B. et al., 1 October 2010. DE2.1.1. User Requirements and Use Cases. ECLAP project. p. 17.

³ Same definition of UGC as described in the Europeana Data Provider & Aggregator Agreements, p. 6 (<http://version1.europeana.eu/web/europeana-project/documents>).

2 Executive Summary and Report Scope

The aim of the ECLAP (European Collected Library of Artistic Performance) project is to create a considerable online archive for performing arts in Europe, which will also become searchable in Europeana. ECLAP is creating a best practice network (BPN) of important European performing arts institutions and archives. Within this BPN ECLAP is going to develop best practise guidelines covering key areas of making digitised performing arts accessible, such as metadata and content modelling, mapping of metadata standards, semantic enrichment, IPR issues and management, business models, ingestion and integration of end-user contributions, education and leisure tools, and digital libraries tools. This will result in cultural enrichment and promotion of European culture, and in improvements in learning and research in the field of performing arts.

This deliverable is the result of the work done in the WP3. This WP is dedicated to the detailed definition and set up of the technical infrastructure which provides tools and services for automated content and metadata collection, processing and posting on Europeana.

Moreover, the ECLAP Social Service Portal and tools provide access to the catalogue for searching and accessing value-added ECLAP services, integrated as a section for performing arts in Europeana. The WP is organised in the following sub-WPs.

WP3.1: Content/Metadata Ingestion infrastructure (M4-M7)

The task is dedicated to definition of:

- Metadata model. ECLAP metadata profile for Performing Arts Contents will be modeled and defined, based on Europeana recommendations and standards and semantics, focusing on its interoperability based on the usage of metadata standards and specifications. This task includes the review of main standards.
- Metadata acquisition tools, and of enrichment procedure and service, web based;
- IPR models and forms
- Setting up AXCP Ingestion tools for digital library management that would be located by many content partner locations. These local tools will provide facilities for digital library organization, integration with OAI-PMH, support ECLAP metadata model, metadata and content acquisition posting them on ECLAP.
- Installations and maintenance, content harvesting, workflow and management, reuse of metadata mapping tools from other project.
- Procedures and protocols for material preparation and posting.
- Content ingestion models and tools for collecting content from users

As a result of this task, ECLAP will have the tools installed at every content partner location and adapted to ECLAP needs, as well as available guidelines and training facilities for its maintenance and installation, which will be used by new members who wish join ECLAP network.

WP3.2: Content/Metadata Processing and Semantification tools (M7-M36)

The purpose of this task is setting up ECLAP Social Service Portal and related AXMEDIS Content Processing Tool in connection with the AXCP Ingestion tools. The setup of the ECLAP Social Service Portal will be realized customizing the DSI XMF social network which provides support for:

- Access to the content, make queries via PC and Mobile/PDA,
- Create communities and groups for the ECLAP Networking,
- Upload content on ECLAP providing metadata and classifications, similar to UGC models
- Socialize with other ECLAP users
- User registration and networking: single user and groups;
- Search and retrieval of content on the basis of semantic information associated with digital content, i.e., web 2.0 technologies

DE3.3.1– Infrastructure: Content and Metadata processing and semantification Best Practice Network

- Play content: rendering visually content as video, audio, web pages, images, document, etc. With multiple player and languages.
- Voting and ranking content;
- Comments and annotations of Content:
- Search for similar users;
- Search for similar objects (digital resources);
- Digital resource upload for professional and User Generated Content, UGC:

Thus the deliverable describes the current state of: ECLAP Social Service Portal, ECLAP automated back office, installation parameters/settings, configuration settings, user manual, scripts, and activities, procedures for creating users, groups, procedures for updating taxonomies, etc.; content and metadata, processing and semantification, AXCP Ingestion tool, content format, model for metadata and content, procedure for content enrichment, etc.

This deliverable improves and extends what reported in:

DE3.1: *Infrastructure: ingestion and processing content and metadata*

DE3.2: *Accessibility and Multilingual support for the ECLAP solution*

The deliverable is structured in sections:

Section 3 reports on the status of Content and Metadata processing and semantification. This section provides the work done for: ECLAP portal installation parameters/settings, content & metadata web upload page, content & metadata edit page, metadata editing, metadata, functionalities of the ECLAP automated back office, content and metadata, content & metadata ingestion, content and metadata analysis.

In Section 4 the status of the users and groups is reported. Section 4.1 reports on the current status on contact ingestion to import contacts that each user has previously registered in other modalities, Section 4.2 content management in groups to allow the group managers to manage a set of contents (and not one by one).

Section 5 reports the reference for the ECLAP User Manual.

Section 6 reports on the current status of the multilingual taxonomy editor.

Section 7 reports on the Europeana module, a widget showing relevant results from the Europeana portal when users search for content on the ECLAP portal.

2.1 Content and Metadata processing and semantification

On ECLAP, professional and user generated content are provided as digital resource & metadata or using a manual web upload, providing a content file together with the metadata entered via a web form, or when they are a great number of content items using an automated content and metadata collection, ingestion and repurposing.

When contents are uploaded through the Upload page, the user is called to fill a form organized to cope with: i) Dublin Core metadata necessary for indexing and retrieval, ii) taxonomy terms for classifying contents, iii) Groups association, iv) single or multiple digital resources locations and v) workflow type related to the life-cycle of content.

Content and Metadata Ingestion is the alternative to the upload page when user has to upload a great number of content items (i.e. importing metadata and content coming from ECLAP partners and Digital Archives).

The Ingestion process allows:

- Harvesting and collecting any kind of content and metadata accessing to a large set of formats and databases.
- Transcoding and adaptation.

- multilingual enrichment of metadata,
- validation of metadata,
- IPR definition.

Users decide to use upload page or ingestion channel, content and metadata are processed in the ECLAP BackOffice. It is on the back-side of the ECLAP Portal and manages all the “dirty” activities in an automated manner. It is established and derived from AXMEDIS AXCP (content processing grid platform) plus additional tools. The ECLAP Back Office is a scalable solution for: Content ingestion and integration, database management, processing, formatting, adaptation, transcoding, metadata mapping and processing, connection with other social networks to propagate queries and get content or for posting content; Content and users similarity analysis and clustering, for users and content recommendations; Content aggregation and integration for educational and entertainment productions, etc..

2.2 ECLAP Portal users and groups

New services have been developed both for single users and for groups, the **Contact Ingestion** and the **Content Management**,

To attract new users on the portal is a very important issue. As it is done in other Social Networks the possibility to import contact information from other sources (email providers or other social networks) is an important feature. The *Contact Ingestion* service allows to import contacts from e-mail providers and Social Networks, import contacts from file and view imported contacts showing the status.

Given the large amount of contents uploaded on the portal, the development of a service that would allow the group managers to manage a set of contents (and not one by one) was deemed essential.

The *Content Management* service developed, offers three main functionalities:

- View the contents associated with the group,
- Filter content on the basis of the workflow type, workflow state, IPR model, IPR model type (public/private), if published for Europeana, if hidden, if have content associated.
- Make actions on all the content selected as set hidden/not hidden, set the workflow type, publish for Europeana, apply the same IPR model, delete objects, etc. The actions that can be done depend on the user role (group manager, group manager & publisher, site administrator).

2.3 Multilingual Taxonomy

The purpose of the multilingual taxonomy editor is to offer an interface for users to translate/validate terms in the ECLAP taxonomy into one of the 13 defined languages for the portal. These 13 languages are: Danish, Polish, Slovenian, Greek, English, Italian, French, Dutch, Spanish, Hungarian, German, Portuguese and Catalan. The functionalities present are to add/edit terms and to add translations to each term, Moreover have been implemented the Multilingual Taxonomy Export allowing to export the taxonomy as XLM or as RDF/XML.

2.4 Europeana module

The purpose of the module is to offer users of the portal the possibility to discover the richness of the content on the portal of Europeana <http://www.europeana.eu>, to which ECLAP content will also be aggregated. This widget works in the way that whenever a user searches for content on the ECLAP portal, the Europeana module, or widget, shows relevant results from the Europeana portal. The module will be enhanced allowing to search for ‘near by’ content, use an improved search (using semantically enriched fields), filter out ECLAP content as they will be shown from the ECLAP portal search results.

2.5 Conclusions

The following table briefly reports the work done and the work to be done or to be completed for the various sections.

DE3.3.1– Infrastructure: Content and Metadata processing and semantification
Best Practice Network

Description	Work done	Work to be done/completed
Web Content Upload/Update on ECLAP	<ul style="list-style-type: none"> Implemented the Web Content Upload Page Implemented Single or Multiple uploads Implemented Upload by providing URLs Implemented Update media content page (Edit Object service) Implemented Administrative pages for Configurations and Back-office settings 	<ul style="list-style-type: none"> None
Web Content Upload/Update Back-Office	<ul style="list-style-type: none"> Implemented the set of Back-office AXCP rules (upload, update, content format, metadata updating) 	<ul style="list-style-type: none"> Updating external tools for possible media format not yet managed Metadata analysis and decision for publication
ECLAP Workflow	<ul style="list-style-type: none"> Implemented the Workflow for ECLAP Workflow Services for Back-Office 	<ul style="list-style-type: none"> None
Metadata Editor	<ul style="list-style-type: none"> Implemented the metadata editing web page Implemented Enrich mode Implemented Validation mode 	<ul style="list-style-type: none"> Editing of Performing Arts metadata
Automatic metadata translation	<ul style="list-style-type: none"> A draft implementation of automatic metadata translation using Google Translate 1.0 API is available 	<ul style="list-style-type: none"> To be finalized the metadata translation rule considering using Google Translate 2.0 API or another translation service
Metadata validation	<ul style="list-style-type: none"> A manual metadata validation workflow is available 	<ul style="list-style-type: none"> The rule for automatic metadata checking for publication needs to be finalized as well as the relative metrics need to be defined
AXCP Ingestion tool	<ul style="list-style-type: none"> Implemented the set of Back-office AXCP rules for massive content and metadata harvesting and uploading 	<ul style="list-style-type: none"> Refining Ingestion process
Contact Ingestion	<ul style="list-style-type: none"> Import contacts view Import contacts from file Import contacts from e-mail providers and Social Networks 	<ul style="list-style-type: none"> None
Content Management in Groups	<ul style="list-style-type: none"> View contents Filter contents Make actions on the content 	<ul style="list-style-type: none"> None
Metadata analysis and similarity	<ul style="list-style-type: none"> User to user Content to user Content to content 	<ul style="list-style-type: none"> None
Europeana Module	<ul style="list-style-type: none"> A search widget block to search on Europeana is available on ECLAP portal 	<ul style="list-style-type: none"> Show nearby content Improve search Filter ECLAP results
Multilingual taxonomy	<ul style="list-style-type: none"> Taxonomy Editor 	<ul style="list-style-type: none">

	• Taxonomy Export	
--	-------------------	--

This document will be also produced in a publishable manner, as a guideline for public access and spreading out.

3 Content and Metadata processing and semantification

On ECLAP, professional and user generated content are provided as digital resource & metadata or using a manual web upload, providing a content file together with the metadata entered via a web form, or when they are a great number of content items using an automated content and metadata collection, ingestion and repurposing.

When contents are uploaded through the Upload page, the user is called to fill a form organized to cope with: i) Dublin Core metadata necessary for indexing and retrieval, ii) taxonomy terms for classifying contents, iii) Groups association, iv) single or multiple digital resources locations and v) workflow type related to the life-cycle of content.

Content and Metadata Ingestion is the alternative to the upload page when user has to upload a great number of content items (i.e. importing metadata and content coming from ECLAP partners and Digital Archives). The Ingestion process allows:

- Harvesting and collecting any kind of content and metadata accessing to a large set of formats and databases.
- Transcoding and adaptation.
- multilingual enrichment of metadata,
- validation of metadata,
- IPR definition.

Both users decide to use upload page or ingestion channel, content and metadata are processed in the ECLAP BackOffice. It is on the back-side of the ECLAP Portal and manages all the “dirty” activities in an automated manner. It is established and derived from AXMEDIS AXCP (content processing grid platform) plus additional tools. The ECLAP Back Office is a scalable solution for: Content ingestion and integration, database management, processing, formatting, adaptation, transcoding, metadata mapping and processing, connection with other social networks to propagate queries and get content or for posting content; Content and users similarity analysis and clustering, for users and content recommendations; Content aggregation and integration for educational and entertainment productions, etc..

3.1 ECLAP Portal installation parameters/settings (Ivan Bruno)

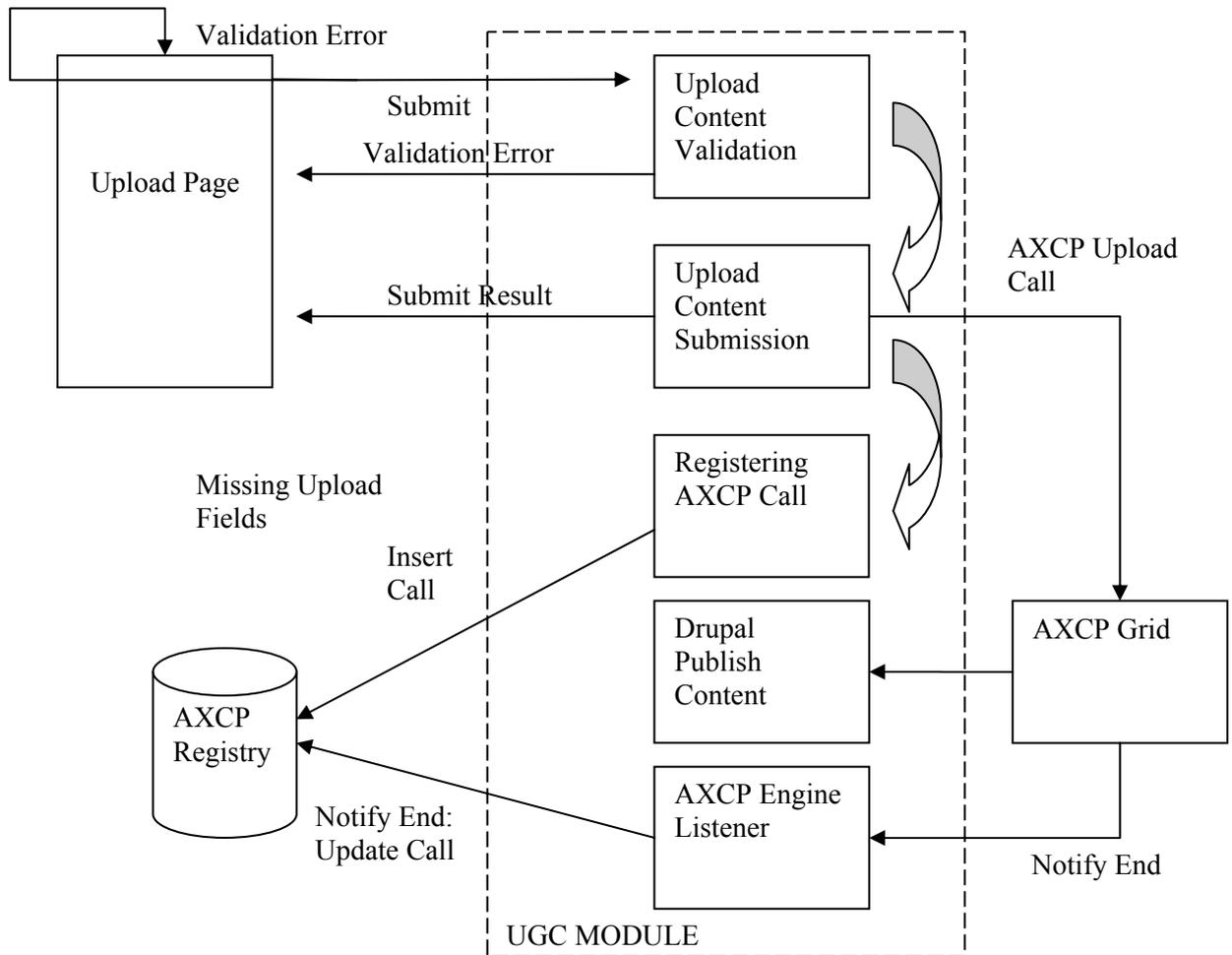
As the ECLAP portal is implemented using the Drupal framework version 6, the modules used for the Content & Metadata creation and management are all Drupal 6 compliant. In the table below an overview of the implemented Drupal ECLAP modules is reported:

Drupal module	Drupal Description	Drupal admin	Drupal ECLAP dependencies
UGC (Axmedis Upload Content)	Allow users to upload own content	Axmedis Submit User Content Settings loaded module User-generated content 1) Configuring the Form Submit User Content 2) Configuring the	Depends on: Taxonomy, og_groups

		User Content for the inclusion	
MetadataEditor (Metadata Editor)	Allows users to edit metadata of contents		Depends on: Axmedis, UGC, EclapWorkflow, Taxonomy, og_groups
Edit_axmedis_obj (Edit Axmedis Object)	Allows users to re-upload previously uploaded content, change icon, access to metadata editing and validating, IPR and workflow		Depends on: Axmedis, UGC, EclapWorkflow, IPR Wizard, MetadataEditor
EclapWorkflow (Eclap Workflow)	Workflow module for ECLAP contents		Depends on: Workflow

3.1.1 Content & metadata web upload Configuration

The ECLAP users can upload contents on the portal through the Upload web page. The upload web page has been developed as part of a more complex module called UGC (User Generated Content). This module provides all functionalities to build and manage the Upload page and the administration issues for customizing and configuring the upload page and the interaction with the AXCP AXMEDIS grid used as back-office platform for content production.



The web upload administration is accessible as AXMEDIS Submit User Content in the Administration panel and it provides two sections:

1. **Configuring the Form Submit User Content:** Configure the form for entering user-generated content
2. **Configuring the User Content for the inclusion:** Configure the environment for the insertion of User Generated Content

3.1.2 Configuring the Form Submit User Content

The Administration panel is accessible only by root/administrator in the Administration section as “Axmedis Submit User Content” and then in “Configuring the Form Submit User Content”.

It allows configuring and customizing the content and metadata upload form according to:

1. Metadata Settings
2. Taxonomy Settings
3. Group Settings
4. Production Settings
5. Publication Settings
6. DRM Settings
7. Validation Settings
8. Publication Workflow Settings
9. Drupal Settings

Metadata Settings

It provides a panel (see the picture below) where the administrator selects the metadata fields to be displayed in the Upload form. Metadata fields are Dublin Core metadata according to the ECLAP requirements for content description. The checkbox close to the metadata name (i.e. dc.Title) allows selecting metadata to use in the upload Form

Metadata Fields	Required	Multiple	Language	Off
<input checked="" type="checkbox"/> dc.Title	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Each metadata can be marked as:

- *Required*, the field is controlled by a JavaScript in the upload page to check if it is not empty.
- *Multiple*, the user can add more fields in the form to cope with multiple instances of metadata
- *Language*, the user has to specify the language of the metadata.
- *Off*, the metadata is disabled

The following picture shows a view of Metadata configuration

DE3.3.1– Infrastructure: Content and Metadata processing and semantification
Best Practice Network

▼ Metadata Settings

Metadata Fields	Required	Multiple	Language	Off
<input checked="" type="checkbox"/> dc.Title	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Creator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Contributor	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Publisher	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Description	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Subject	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Language	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Coverage	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Relation	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Rights	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Source	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Type	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Format	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Identifier	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Date	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> dc.Abstract	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> dc.AccessRights	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> dc.AccrualMethod	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> dc.AccrualPeriodicity	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> dc.AccrualPolicy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> dc.Alternative	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Audience	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> dc.Available	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> dc.BibliographicCitation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> dc.ConformsTo	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.Created	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> dc.DateAccepted	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> dc.DateCopyrighted	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> dc.DateSubmitted	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> dc.EducationLevel	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Taxonomy Settings

It provides a panel (see the picture below) where the administrator selects if a taxonomy has to be associated or not with the content. The taxonomy terms set is the current enabled taxonomy on the portal.

▼ [Taxonomy Settings](#)

Enable/Disable Taxonomy:

Don't use taxonomy

Allow users to assign *Classification* terms to the the content.

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Groups settings

It displays a panel (see the picture below) where the administrator gives to users the possibility to assign own content to groups where he is registered.

▼ [Group Settings](#)

Enable/Disable Groups:

Allow users to assign the content to Groups

Don't use Groups

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Production settings

It displays a panel (see the picture below) where the administrator selects further devices to manage in content production (i.e. smart phone, pda, mobile, set of box). More, the selection can be made visible to users in order to let them the choice of possible devices or hidden to define the default behaviour of the system.

▼ [Production Settings](#)

Select Device(s) Target:

pda

mobile

stb

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Show/Hide Production Options:

Show all allowed users to select values

Hide all options to users. They will be managed as default value by the system

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Publication settings

It displays a panel (see the picture below) where the administrator selects distribution channel (i.e. http or/and p2p). More, the selection can be made visible to users in order to let them the choice of distribution channel or hidden to define the default behaviour of the system.

▼ [Publication Settings](#)

Select Download Channel:

http (Allow download content via http)

p2p (Allow get content via P2P)

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Show/Hide Publication Options:

Show all allowed users to select values

Hide all options to users. They will be managed as default value by the system

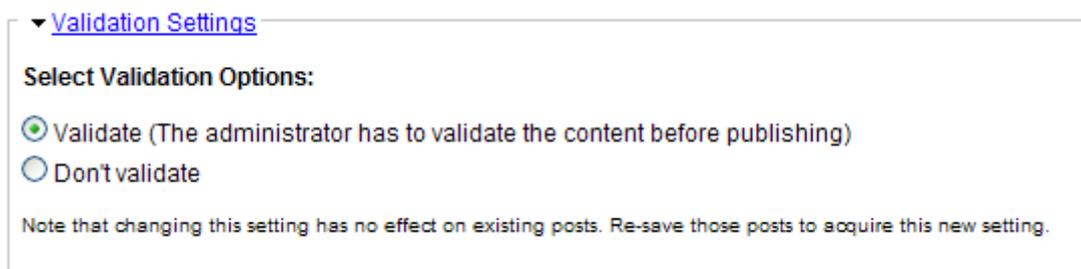
Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Validation settings

It displays a panel (see the picture below) where the administrator defines if the content upload has to be validated or not. The Upload/Publication procedure can be:

- direct/without any moderating device: when this modality is on, all uploaded files via the upload form are published without any further control. This direct modality is available only for some privileged users (i.e a group coordinator)
- with moderating device/controlled, which means validation is needed: when this modality is on, all uploaded files via the upload form are gathered in a database called Administration. Such content are published no sooner than the content manager will have given his assent. This modality is set by default for all new registered users.

The direct/moderate mode is possible by defining a drupal permission called 'trusted user'. The administrator could assign this privilege to such users considered as trusted uploader.



▼ [Validation Settings](#)

Select Validation Options:

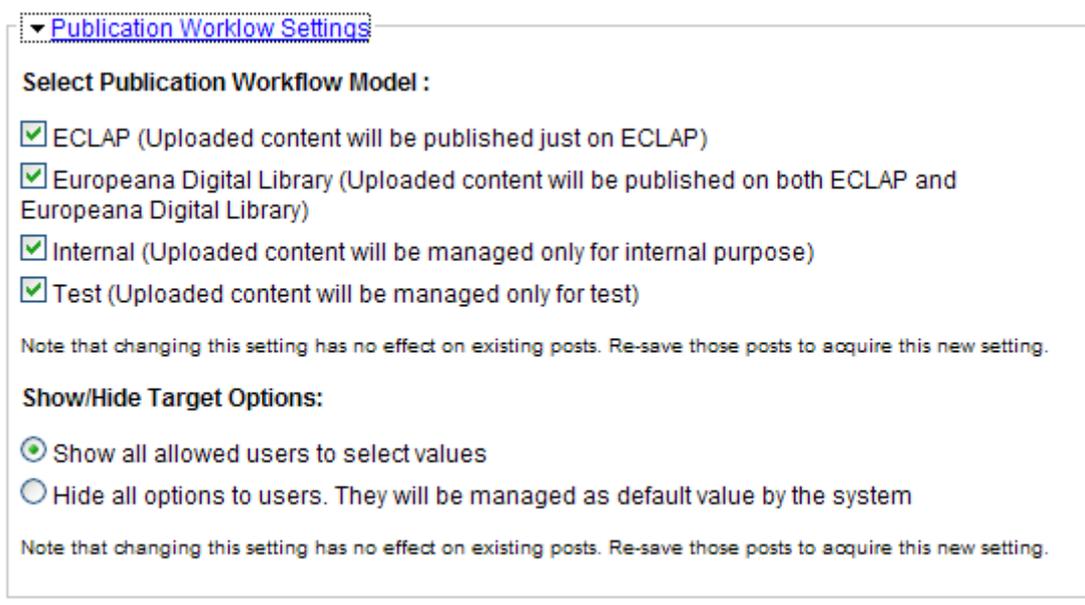
Validate (The administrator has to validate the content before publishing)

Don't validate

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Publication Workflow Settings

It displays a panel (see the picture below) where the administrator selects workflows type for content publication. More, the selection can be made visible to users in order to let them the choice of workflow type or hidden to define the default behaviour of the system.



▼ [Publication Workflow Settings](#)

Select Publication Workflow Model :

ECLAP (Uploaded content will be published just on ECLAP)

Europeana Digital Library (Uploaded content will be published on both ECLAP and Europeana Digital Library)

Internal (Uploaded content will be managed only for internal purpose)

Test (Uploaded content will be managed only for test)

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

Show/Hide Target Options:

Show all allowed users to select values

Hide all options to users. They will be managed as default value by the system

Note that changing this setting has no effect on existing posts. Re-save those posts to acquire this new setting.

3.1.3 Configuring the User Content for the inclusion

This section allows administrator configuring the production environment and the interaction/communication with the AXCP AXMEDIS Grid back-office.

CONFIGURING THE USER CONTENT FOR THE INCLUSION

▼ Environment Settings

UGC AXCP Credentials:

 The user credentials for AXCP Scheduler dedicated to manage UGC rules

UGC AXCP Net Address:

 The IP address for AXCP Scheduler dedicated to manage UGC rules

AXCP Engine WSDL:

 The Web Service WSDL file location for AXCP

AXCP Engine Notification Request:
 Disable
 Enable
 Enable/Disable the AXCP Engine Notification Request

UGC Upload file folder:

 The upload folder path where digital content will be stored

UGC Base Network Address:

 The network address where uploaded contents are available

UGC Owner Name:

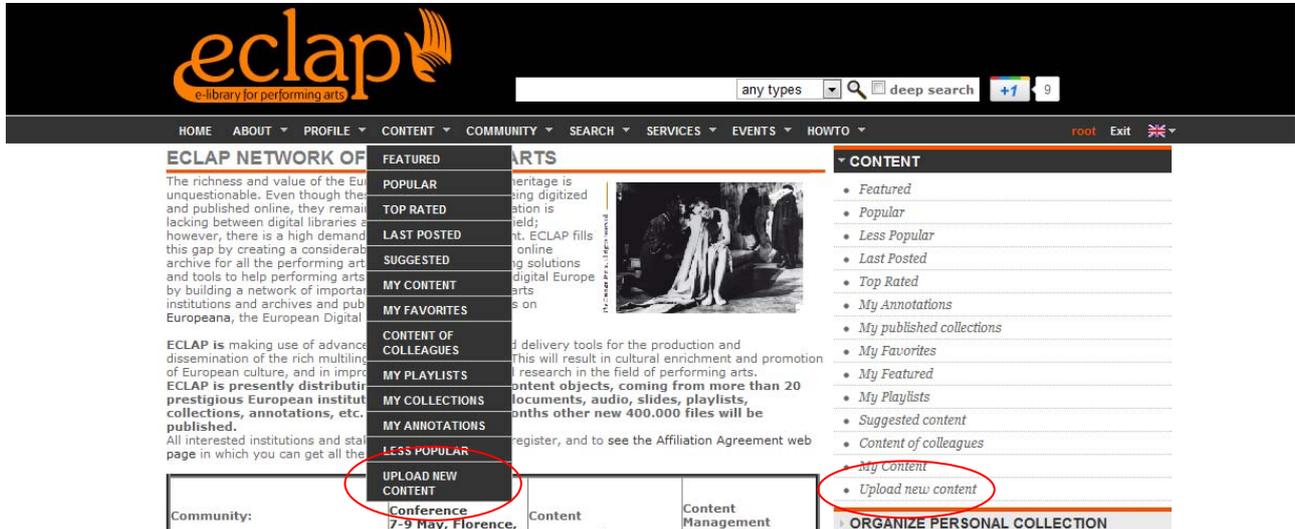
 The owner of the pushed content

3.1.4 Configuration options

Name	Context (drupal, DB,...)	Notes
AXCP Credentials	Drupal/AXCP	The user credentials for AXCP Scheduler dedicated to manage UGC rules
AXCP Net Address	Drupal/AXCP	The IP address for AXCP Scheduler dedicated to manage UGC rules
AXCP Engine WSDL	Drupal/AXCP	The location of wsdl file for creating a WSDL Client in php and communicating with AXCP Scheduler
AXCP Notification Request	Drupal/AXCP	Enable/Disable the request of a notification coming from AXCP (Success or failure)
UGC Upload file folder	Drupal/AXCP	The upload folder path where digital content will be stored
UGC Base Network Address	Drupal/AXCP	The base URL to be used in conjunction with the upload folder to build the url for getting data by the AXCP in content production
UGC Owner Name	Drupal/AXCP	The default value for the owner of content

3.2 Content & metadata web upload page

The upload page is accessible at the Drupal path 'ugc/upload' and by hovering on Content->Upload New Content on top menu of front-end or clicking on Upload New Content in "Content" clock on the left column. The upload page is available only for authenticated users.



The upload page consists in a form divided into sections:

INSERT CONTENT

- Metadata Section
- Workflow type identification for the uploaded content
- Taxonomy Classification
- Groups Section
- Resource Section

dc.Language:

Select the language of the following digital resource if it has a specific one; otherwise, leave it empty.

Select one or more files for upload.

Decide from where and how you can provide the file(s). Multiple files can be provided via massive upload (first option) or providing them with an hard disk or FTP or database. This last option is available only for partners please content the ECLAP coordinator info@eclap.eu. Single files can be directly provided with this upload page according to the next first option or providing an FTP, HTTP access, the next second option.

One or multiple files on your disk.:

SELECT FILES

With this button you can select multiple files in a single directory with a batch upload. Those files can be of different kind/type extension. Multiple selection of files has to be performed by keeping pressed the control key. For each file, you will have the possibility of specifying a different title. All the others, metadata will be taken from above. And, you have the possibility of editing, changing the metadata later, for each single content, by using the Edit Object action on your content.

File URL:

Examples are: ftp://ftp.yourdomain.com/filename.doc,
ftp://username:password@ftp.yourdomain.com/filename.doc,
http://www.adomain.com/myfile.gif,
http://username:password@www.mydomain.com/thisfile.pdf

- Terms of Use

Upload **Reset**

3.2.1 Metadata Section

The user has to fill the Dublin Core fields necessary for indexing and retrieval. Some of those metadata are required according to the metadata settings. Metadata marked as multiple provides a couple of iconized

button like  that allows adding/deleting fields (i.e. another title, more authors, etc...). To cope with multilingual metadata, the user has to associate the metadata definition language with the metadata set, this field is mandatory.

INSERT CONTENT

▼ Metadata Section

Metadata Definition Language

Select language for the following metadata: *

▼
Language of the following metadata

dc.Title: * ? + 

dc.Creator: * ? + 

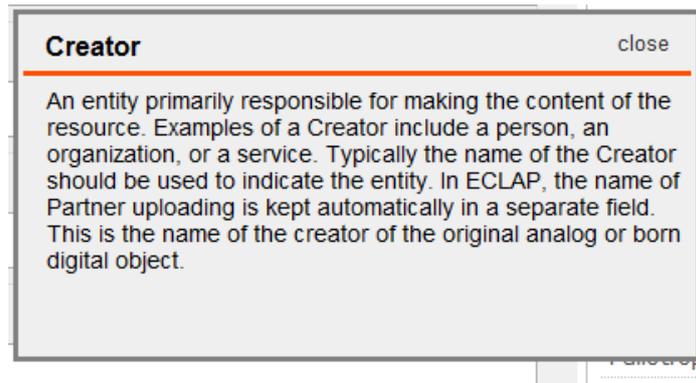
dc.Contributor: ? + 

dc.Publisher: ? + 

dc.Description: * ? + 

dc.Subject: * ? + 

Each metadata input provides the  icon that allows displaying the definition/help dialog related to the metadata. In the following, an example of help dialog is shown:



3.2.2 Resource Section

Two mutually exclusive modalities are available. The former allows selecting single or a multiple selection of local files from user's computer (File on your disk), the latter to edit a valid URL of a remote digital resource (File URL). If the resource has a language, the language selector allows performing the choice.

Resource Section

dc.Language:
 ▼
Select the language of the following digital resource if it has a specific one; otherwise, leave it empty.

Select one or more files for upload.

Decide from where and how you can provide the file(s). Multiple files can be provided via massive upload (first option) or providing them with an hard disk or FTP or database. This last option is available only for partners please content the ECLAP coordinator info@eclap.eu. Single files can be directly provided with this upload page according to the next first option or providing an FTP, HTTP access, the next second option.

One or multiple files on your disk.:
SELECT FILES
With this button you can select multiple files in a single directory with a batch upload. Those files can be of different kind/type extension. Multiple selection of files has to be performed by keeping pressed the control key. For each file, you will have the possibility of specifying a different title. All the others, metadata will be taken from above. And, you have the possibility of editing, changing the metadata later, for each single content, by using the Edit Object action on your content.

File URL:

Examples are: ftp://ftp.yourdomain.com/filename.doc,
ftp://username:password@ftp.yourdomain.com/filename.doc,
http://www.adomain.com/myfile.gif,
http://username:password@www.mydomain.com/thisfile.pdf

1. Uploading via Files on disk

By clicking the “Selected Files” button a File dialog is displayed and user can select one or more files to upload.

In event of multiple files, a text control is added to each file in order to write an auxiliary title derived from the main title or a new text. The main title coming from metadata section is provided as default. Auxiliary titles are required and checked before starting the upload.

A progress bar allows controlling the status of upload. In event of failure an error message replaces the progress bar. At the end of upload, the box of uploaded content is removed.

Multiple contents are uploaded sequentially.

The cross on top-right of each upload box allows cancelling the upload.

Resource Section

dc.Language:
it

Select the language of the following digital resource if it has a specific one; otherwise, leave it empty.

Select one or more files for upload.

Decide from where and how you can provide the file(s). Multiple files can be provided via massive upload (first option) or providing them with an hard disk or FTP or database. This last option is available only for partners please content the ECLAP coordinator info@eclap.eu. Single files can be directly provided with this upload page according to the next first option or providing an FTP, HTTP access, the next second option.

One or multiple files on your disk.:

SELECT FILES

dc.Title: *
Title for file 1

ComputerGraphicsParte1.pdf (183.79KB)

dc.Title: *
Title for file 2

ComputerGraphicsParte2.pdf (1.84MB)

dc.Title: *
Title for file 3

ComputerGraphicsParte3.pdf (2.05MB)

dc.Title: *
Title for file 4

ComputerGraphicsParte4.pdf (871.97KB)

dc.Title: *
Title for file 5

ComputerGraphicsParte5.pdf (452.65KB)

With this button you can select multiple files in a single directory with a batch upload. Those files can be of different kind/type extension. Multiple selection of files has to be performed by keeping pressed the control key. For each file, you will have the possibility of specifying a different title. All the others, metadata will be taken from above. And, you have the possibility of editing, changing the metadata later, for each single content, by using the Edit Object action on your content.

File URL:

Examples are: ftp://ftp.yourdomain.com/filename.doc,
ftp://username:password@ftp.yourdomain.com/filename.doc,
http://www.adomain.com/myfile.gif,
http://username:password@www.mydomain.com/thisfile.pdf

2. Uploading via URL

The user can edit the File URL field with the address of the resource. Both FTP and HTTP protocols are available. If the resource to be uploaded is freely accessible without any username and password, the syntax to be used is ftp://ftp.yourdomain.com/filename or http://www.yourdomain.com/filename

If username and password are necessary to access the resource, the syntax has to be:

ftp://username:password@ftp.yourdomain.com/filename

or

<http://username:password@www.yourdomain.com/filename>

3.2.3 Workflow type identification

A workflow type can be associated with the life-cycle of content as defined in the workflow section. This is used by the ECLAP workflow rules to orchestrate content enrichment, validation and publication. As depicted in the picture, four workflow types are available.

▼ Workflow type identification for the uploaded content

This section allows selecting the publication workflow model for the content management, details are reported in terms of use, see below. ECLAP model is set by default.

Choose workflow model:

- EUROPEANA: the uploaded content will be processed and enriched with the aim of publishing it on Europeana (European Digital Library) and on ECLAP portal and devices. If you are an ECLAP partner or affiliated partner, you are going to have a full control about the enrichment and publication on Europeana.
- ECLAP: the uploaded content will be processed with a simplified workflow and only published on ECLAP and on its devices. The provided content can be edit and enriched.
- INTERNAL: the uploaded content is managed only for internal purpose. The content can be edited and enriched and it is published on ECLAP services.
- TEST: the uploaded content will be managed for test and/or trial purposes. This also means that the ECLAP personnel can remove this kind of content without requesting any permission to the uploading user.

3.2.4 Taxonomy Classification

It is a multiple choice classification terms selector, such terms allow classifying contents and will be used in indexing and retrieval.

3.2.5 Groups Section

It is a multiple ECLAP Groups selector. Content is associated to one or more Groups and/or could be public (visible to all uses). This information is used in indexing and retrieval.

▼ Taxonomy Classification

Classification:

- None --
- Genre
- Biography
- Comedy
- Comic
- Drama
- Epic
- Interview
- Life

Select the item you want to insert, you can enter multiple items by holding down the control key

▼ Groups Section

Associate the content to one or more groups, define if it has to be private.

Your groups:

- Sound & Vision (Beeld en Geluid)
- Test - group
- The Institute of Polish Culture University of Warsaw, IKP
- The National Audiovisual Institute
- Thematic Working Groups & Workshops
- TWM
- Universidad de Castilla La Mancha, UCLM
- WG: Digital Libraries Tools
- WG: Intellectual Property and Business Models for Content
- WG: Performing Arts Education and Training tools

Publish this post to these groups. Multiple selection can be performed by keeping pressed the control key.

Public
Show this post to everyone, or only to members of the groups checked above. Posts without any groups are always *Public*.

3.2.6 Terms of Use

In this section the user has to accept the terms of use of Eclap Portal, the **Term of use** link allow to go in the terms page. Acceptance of terms allow user clicking the start upload button.

▼ Terms of Use

To continue in the operation please accept the Terms of Use.

Yes, I have read and accept the Terms of Use.

3.2.7 Metadata checking

When user clicks on the upload button a JavaScript checks if all required fields were filled. The page stops the upload and displays when non-filled required fields are empty and highlights them in yellow.

INSERT CONTENT

▼ Metadata Section

Metadata Definition Language

Select language for the following metadata: *

en ▾
 Language of the following metadata

dc.Title: * ? + 🗑️
 title

dc.Creator: * ? + 🗑️
 creator

dc.Contributor: ? + 🗑️

dc.Publisher: ? + 🗑️

dc.Description: * ? + 🗑️

dc.Subject: * ? + 🗑️

Resource Section

dc.Language:
 ▾
 Select the language of the following digital resource if it has a specific one; otherwise, leave it empty.

Select one or more files for upload.

Decide from where and how you can provide the file(s). Multiple files can be provided via massive upload (first option) or providing them with an hard disk or FTP or database. This last option is available only for partners please contact the ECLAP coordinator info@eclap.eu. Single files can be directly provided with this upload page according to the next first option or providing an FTP, HTTP access, the next second option.

One or multiple files on your disk.:
 SELECT FILES
 Resource is missing!
 Resource is missing!
 With this button you can select multiple files in a single directory with a batch upload. Those files can be of different kind/type extension. Multiple selection of files has to be performed by keeping pressed the control key. For each file, you will have the possibility of specifying a different title. All the others, metadata will be taken from above. And, you have the possibility of editing, changing the metadata later, for each single content, by using the Edit Object action on your content.

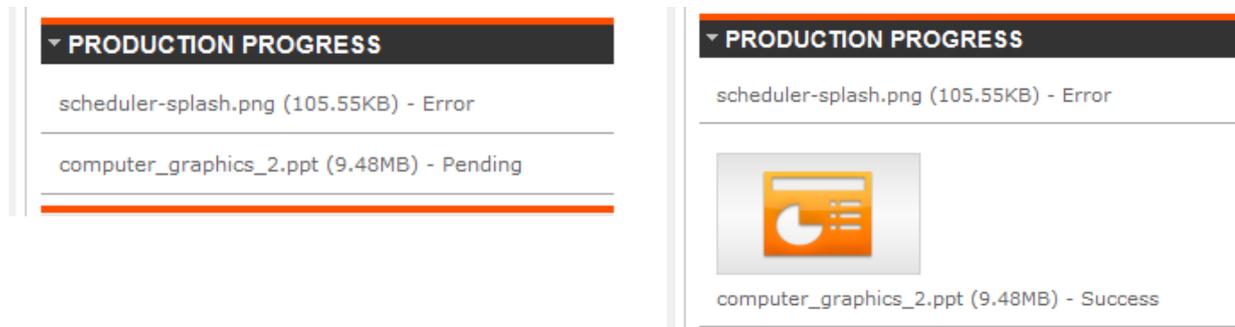
File URL:

 Examples are: ftp://ftp.yourdomain.com/filename.doc,
 ftp://username:password@ftp.yourdomain.com/filename.doc,
 http://www.adomain.com/myfile.gif,
 http://username:password@www.mydomain.com/thisfile.pdf

3.2.8 Production progress

When user uploads files, he can monitor the production in real time by means the Production progress block. In this block the current status of production is reported and updated automatically.

When the production started, the status is “Pending” and in event of success, the status is set to “Success” and the icon associated to the uploaded content is displayed.



3.3 Content & metadata edit page

The page is accessible via the Edit Object link in the “Actions” menu placed on the left into the content description as reported in the following picture



The page allows immediate changes or accessing to specific editing tools.

3.3.1 Update Content

The update content section allows the content owner or the administrator to change the digital resource by replacing the existing one. The form is the same of the Resource Section of the Web Upload page; this allows providing a file on user’s disk or an URL. The form submission is sent to the server that will invoke the content production process on AXCP Grid (update mode) without changing existing metadata.

Change/update a file of an Object by versioning/upload
Reupload/update the file of an object

dc.Language:
hu ▾
Select the language of the following digital resource if it has a specific one; otherwise, leave it empty.

Update Resource (*)
Define the content to update

One file on your disk:
SELECT FILE
Choose a file located on your computer

File URL:

Examples are: ftp://ftp.yourdomain.com/filename.doc,
ftp://username:password@ftp.yourdomain.com/filename.doc,
http://www.adomain.com/myfile.gif,
http://username:password@www.mydomain.com/thisfile.pdf

3.3.2 Change Icon

The user can change the prebuilt icon by uploading a new image. The system generates and assigns the new icon to the object.

Change Icon, preview
Upload an icon to replace the current



File containing the icon preview (gif, png and jpg are accepted of any size) :
 Nessun file selezionato
Choose a file located on your computer

3.3.3 Edit workflow

It allows the content owner or the administrator to change the workflow type associated with the life-cycle of content.

▼ Edit Workflow Model

This section allow changing the publication workflow model assigned to this content.

Choose workflow model:

- EUROPEANA: the uploaded content will be processed and enriched with the aim of publishing it on Europeana (European Digital Library) and on ECLAP portal and devices. If you are an ECLAP partner or affiliated partner, you are going to have a full control about the enrichment and publication on Europeana.
- ECLAP: the uploaded content will be processed with a simplified workflow and only published on ECLAP and on its devices. The provided content can be edit and enriched.
- INTERNAL: the uploaded content is managed only for internal purpose. The content can be edited and enriched and it is published on ECLAP services.
- TEST: the uploaded content will be managed for test and/or trial purposes. This also means that the ECLAP personnel can remove this kind of content without requesting any permission to the uploading user.

[Change](#)

3.3.4 Enrich

It allows the content owner, workflow Enricher user or the administrator to open the Metadata Editor page and then edit metadata.

Modify and Enrich Multilingual Metadata

[Click here to change/enrich object multilingual metadata](#)
[Edit Metadata \(via Metadata Editor\)](#)

3.3.5 Validate

It allows the content owner, workflow Validator user or the administrator to open the Metadata Editor page and then validate metadata.

Validate metadata

[Click here to validate metadata](#)
[Validate Metadata \(via Metadata Editor\)](#)

3.3.6 Ipr Wizard

It allows the content owner, workflow IPR user or the administrator to open the IPR Wizard page

IPR Wizard

[Click here to open the IPR Wizard: define, change and assign IPR models and Europeana.rights, licenses.](#)
[IPR Wizard](#)

3.3.7 Workflow

It allows the administrator to open the Workflow administration page

Workflow Section

[Click here to edit/view workflow](#)
[Workflow](#)

3.4 Metadata Editing, Metadata Editor (DSI)

Metadata editor is the tool for enriching and validating metadata. Since enrichment and validation activities are governed by ECLAP Workflow, the Metadata Editor is a workflow-drive tool.

It allows:

- Opening workflow session by executing:
 - WF Under Enrich transition in event of enrichment activity
 - WF Under Validation transition in event of validation activity
- Closing workflow session and executing the WF Upload transition
- Modifying the metadata values (proposed during the content upload)
- Filling metadata in other languages
- Adding more fields and values
- Validating/ Invalidating metadata
- Comparing metadata in different languages
- Manual translation of metadata in other languages
- Easy recognition of made changes by means of text highlighting
- Undo changes
- Tracing each action by recording them in the workflow tables:
 - Workflow_info table
 - Workflow_metadata table
- Viewing current workflow status of content, metadata counters (i.e. number of current metadata, number of validated metadata, number of native metadata)
- Viewing the name of user who made last change on metadata
- Applying all made changes by saving changed metadata into databases
- Updating index archive in event of changes
- Checking for made changes before leaving session
- Accessing in read-only mode if user has not the permission to edit (eclap workflow permission missing or content already in use by another WF user)
- Browsing the assigned objects by workflow

Only users with ECLAP workflow permission can work fully with the editor. However, according to the user role, the editor works in Enrichment mode for the WF Enricher role and in Validation mode for WF Validation role. Users without a workflow role access anyway to the editor only in read-only mode.

The Metadata Editor depends on metadata settings in User Generated Content module. This means that all DCMI metadata are managed according to current set-up.

3.4.1 Metadata Editor Web Page

The metadata editor web page is structured as a 2 columns page (left column and metadata panel) as depicted in the following picture:

The screenshot displays the ECLAP test web application interface. At the top, there is a navigation menu with options like HOME, ABOUT, PROFILE, CONTENT, COMMUNITY, SEARCH, SERVICES, EVENTS, and HOWTO. Below the navigation is a search bar with a dropdown menu set to 'any types' and a 'deep search' button. The main content area is divided into two columns. The left column, titled 'METADATA EDITOR', contains a 'Search Panel', 'Metadata Panel', 'Apply Changes', 'Content to be processed', 'History', 'Workflow', and 'Close Metadata Editor session' buttons. Below these is a 'WORKFLOW' section with a table of session details and a 'PREVIEW' section showing a small image of a theatrical performance. The right column, titled 'METADATA PANEL', is split into 'Reference' and 'Changes' tabs. Each tab shows a list of metadata fields (dc.Title, dc.Creator, dc.Contributor, dc.Publisher, dc.Description, dc.Subject, dc.Coverage) with input fields and 'By: root' labels. The 'Changes' tab also includes icons for editing and deleting each field.

ECLAP Connecting stages

The **left column** displays a set of blocks for:

Menu: this block provides the access to main functionalities:

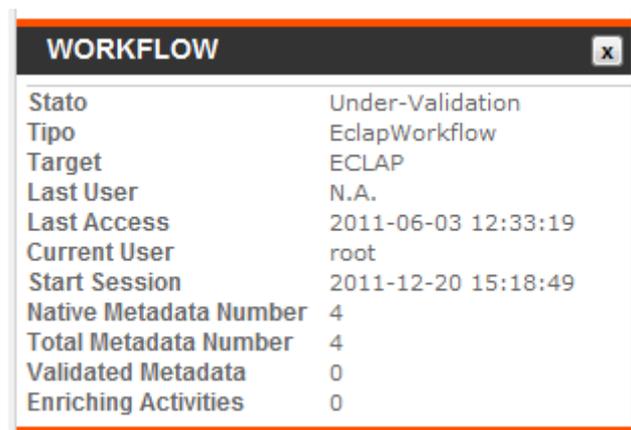
- **Search Panel** (available only for the administrator) opens/closes a panel in the middle area with a search form for content searching
- **Metadata Panel** opens/closes a panel in the middle area with all metadata divided into tabs
- **Apply changes** sends to the server all made changes on metadata (both new data and changes).
- **Content to be processed** open/close the block with all contents assigned by the workflow
- **History** (available only for the administrator) opens/closes a dialog with the list of made changes
- **Workflow** opens/closes the block of workflow properties and info
- **Close Metadata Editor Session** sends to server the end of activity to close the workflow session

A close-up view of the 'METADATA EDITOR' menu. It features a dark header with the title 'METADATA EDITOR' in white. Below the header are six buttons with a light gray background and dark text: 'Metadata Panel', 'Apply Changes', 'Refresh List of Contents', 'Workflow', and 'Close Metadata Editor session'.

Locale: this block displays a language selector (default language is the language selected by the user in the main page) to change the language of the page.

Workflow: This block displays workflow info regarding:

- **State:** the current state of content according the workflow status (Under-Enrichement, Under-Validation or others workflow states)
- **Type:** the workflow model under usage (EclapWorkflow)
- **Target:** the workflow target of content (Eclap, Europeana & Eclap, Test, Internal)
- **Last access:** date and time of last editing
- **Last user:** who made the last editing session
- **Current user:** name of user who is currently editing the object
- **Start Session:** date and time of current editing session
- **Native Metadata Number:** the number of native metadata
- **Total Metadata Number:** the current number of metadata
- **Validated Metadata:** the current number of validated metadata
- **Enriching Activities:** how many time the content has been modified



WORKFLOW	
Stato	Under-Validation
Tipo	EclapWorkflow
Target	ECLAP
Last User	N.A.
Last Access	2011-06-03 12:33:19
Current User	root
Start Session	2011-12-20 15:18:49
Native Metadata Number	4
Total Metadata Number	4
Validated Metadata	0
Enriching Activities	0

Preview: this block embeds the content player for the preview.

Content to be processed: this block displays the list of content assigned by workflow

The **Metadata Panel** is the metadata edit area and it is organized in tabs:

DCMI: it allows editing the Dublin Core set of metadata.

Taxonomy : it allows editing the Taxonomy metadata

Groups : it allows editing the Groups metadata

Properties : it allows editing the Properties metadata

A further tab will be added to cope with “**Performing Arts**” metadata

DCMI Metadata

The DCMI metadata is divided into two sub-areas: Reference and Changes.

The Reference fields set show the current metadata values in read-only mode and for each of them the username who made the last change/edit (**By**). The Changes field set shows a copy of reference set where the user can edit/adding values. In both case each metadata is displayed in a text box with a label reporting the metadata field name. In the changes view each metadata has also a set of iconized buttons that allow performing:

- **View:** It opens a popup dialog displaying a better readable string of metadata 

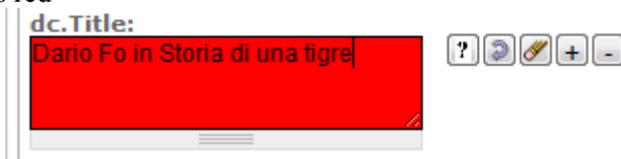
- **Undo:** Reset the value to the first time displayed string 
- **Add more:** it adds a new empty text box with own buttons. 
- **Remove:** it removes the text box/metadata and value 
- **Validate/Invalidate:** it validates/invalidates the metadata 
- **Empty:** it clears the text box 



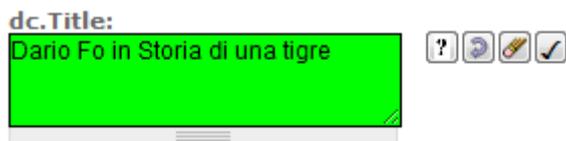
Add and remove buttons are available and displayed with multiple metadata according to the metadata set-up in User Generated Content module.

The validate/invalidate button is displayed/enabled only for administrator and user with WF Validator role.

During the enrichment every time a user made a change or edit a new metadata the text box/selector background colour becomes red



During validation, the text box/selector background colour of validated metadata becomes green, otherwise white.



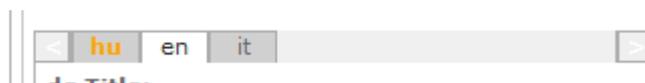
Every time a user invalidates a validated metadata the text box/selector background colour becomes red (as a metadata change)

A language tabs allow browsing on available languages in both views and this allows comparing a language with another or to help the user in translating the metadata.

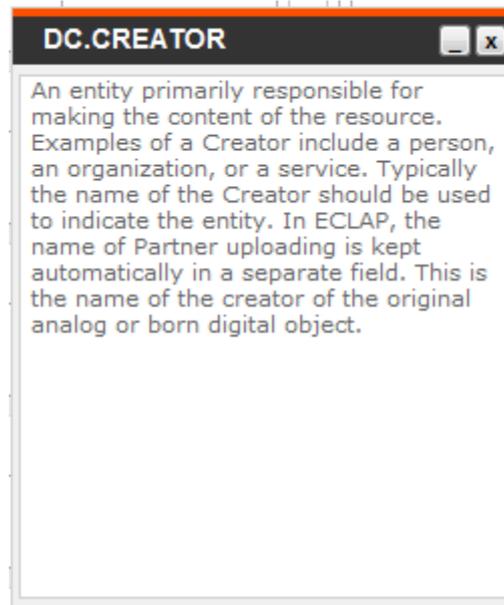


The original language is coloured by orange.

Usually, the tabs show the original language of metadata and the languages of user associated with his workflow role. In the following picture the tabs of a user skilled in English and Italian language for content with Hungarian metadata.

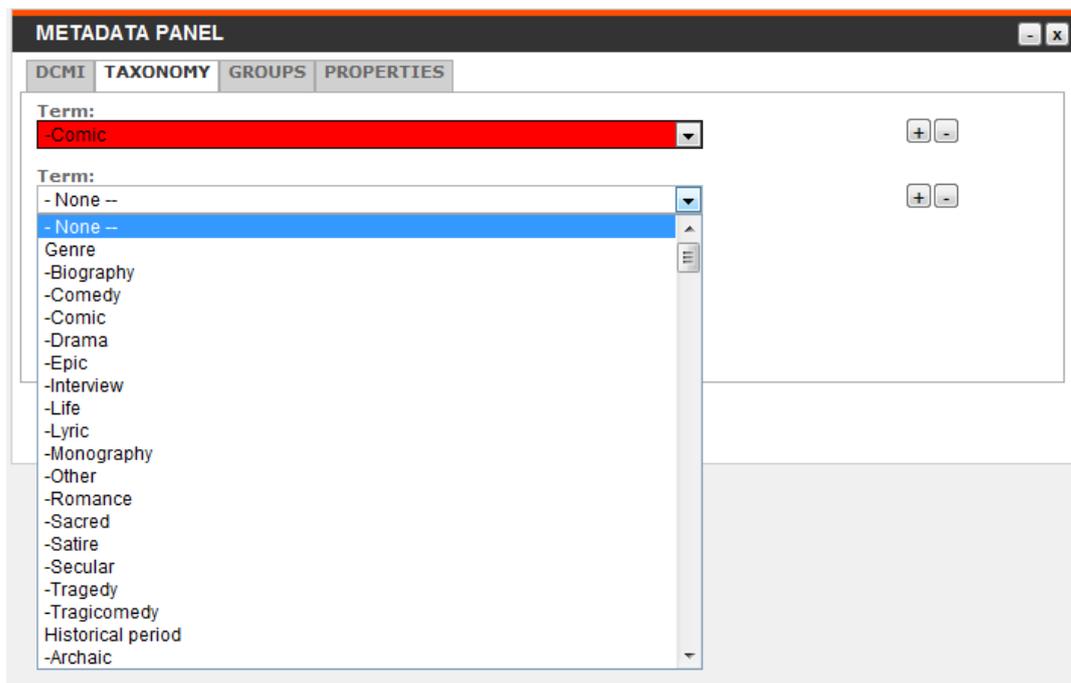


The **View** button display a dialog with the help related to metadata as reported in the following picture:



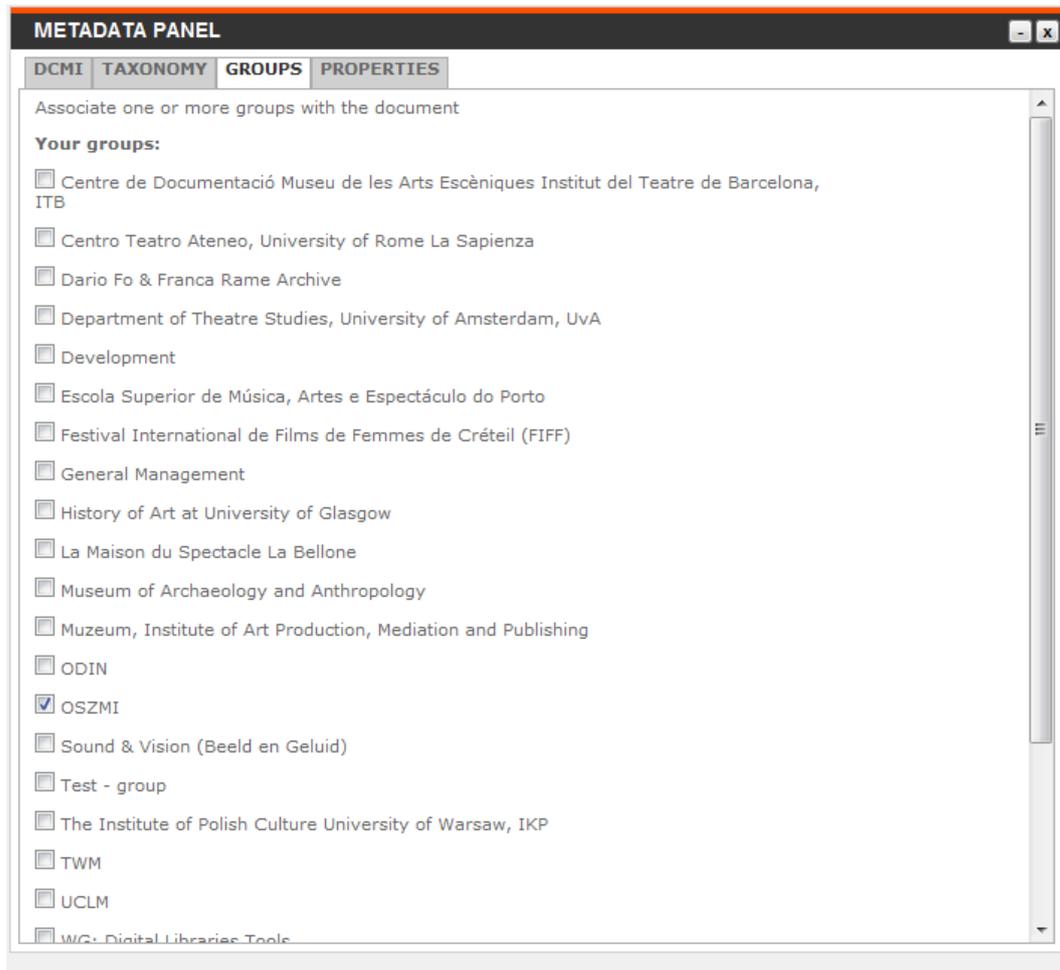
Taxonomy Metadata

The Taxonomy panel allows adding/removing classification terms by means a text selector (single selection) and add/remove buttons. Every time an existing value is changed by selecting a different one, it becomes red to mark that it has been changed/removed.



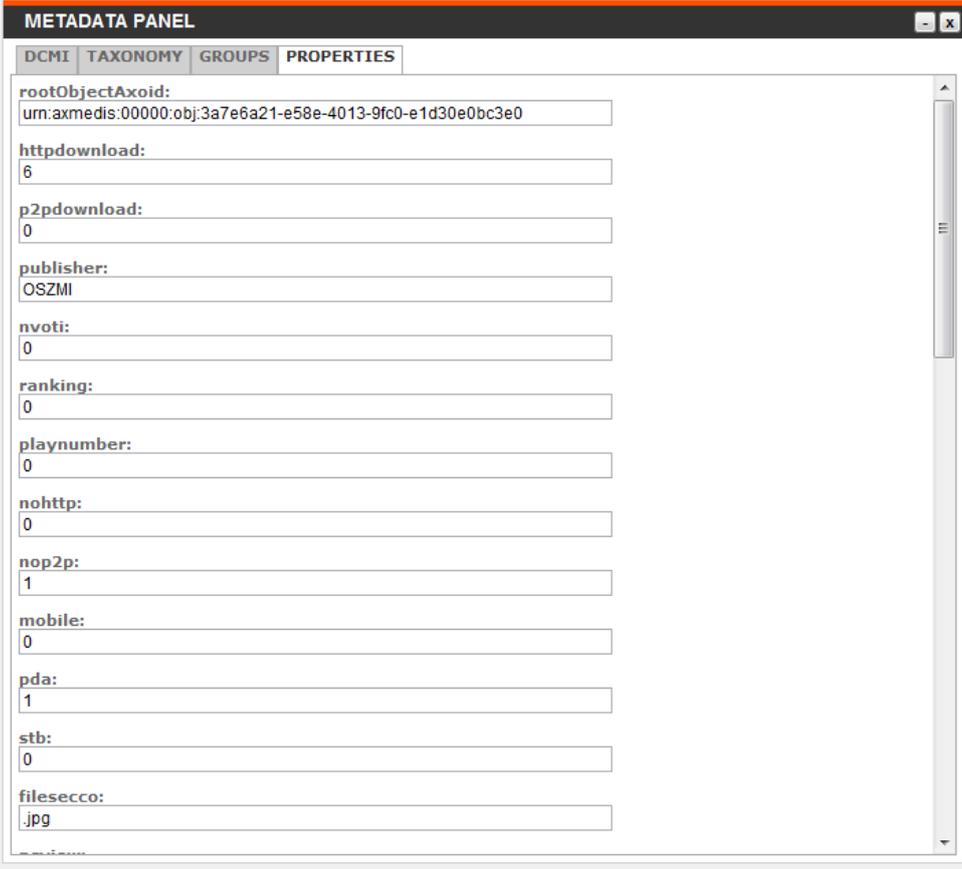
Groups Metadata

The Groups panel shows the list of available groups. Adding/removing a group is possible by enabling/disabling checkboxes. When a checkbox is modified, it becomes red to mark that it has been changed/removed.



Properties Metadata

The Properties metadata shows the technical metadata. This tab is available only for the administrator.



The screenshot shows a window titled "METADATA PANEL" with a tabbed interface. The "PROPERTIES" tab is selected. The form contains the following fields and values:

Field Name	Value
rootObjectAxoid:	urn:axmedis:00000:obj:3a7e6a21-e58e-4013-9fc0-e1d30e0bc3e0
httpdownload:	6
p2pdownload:	0
publisher:	OSZMI
nvoti:	0
ranking:	0
playnumber:	0
nohttp:	0
nop2p:	1
mobile:	0
pda:	1
stb:	0
filesecco:	.jpg

3.4.2 Metadata Editor Enrichment mode

The editor works in Enrichment mode for the WF Enricher role. The editor can be used to edit/enrich metadata. Before starting a workflow transition is invoked according to the workflow model (transition Uploaded → Under-Enrich). If the transition is done with success, this means no other user is working on the content, the workflow tables is updated with the user id and start session date and time, otherwise the content was locked by other user and the editor is opened in read-only mode just for browsing metadata. When user is authorized by the workflow he has full access to editing capabilities and metadata are displayed according to his language capabilities. The workflow box displays the current information.

Apply Changes

The apply changes function allows the user to fix own metadata changes. The browser sends to the server the list changes (new, validate, invalidated, updated and delete metadata) and in event of success all panels are refreshed with the applied changes. The list of changes contains all information according to the data model.

End session

The Close Metadata Editor Session function allows the user to end metadata editing/validating session. The page checks if there are changes to apply. In event of positive check a dialog is displayed to ask for apply changes before exit or to continue discarding all changes. Otherwise, the End Session is sent to the server

The screenshot displays the ECLAP Metadata Editor interface. At the top left is the ECLAP logo (e-library for performing arts). The navigation bar includes links for HOME, ABOUT, PROFILE, CONTENT, COMMUNITY, SEARCH, SERVICES, EVENTS, and HOWTO. A search bar is present with a dropdown menu set to 'any types' and a 'deep search' button. The 'Last EnricherUser' is indicated as 'root' in the top right corner. The main interface is divided into three main sections: a left sidebar, a central metadata editor, and a right-hand 'Changes' panel. The sidebar contains a 'METADATA EDITOR' section with options like 'Search Panel', 'Metadata Panel', 'Apply Changes', 'Content to be processed', 'History', 'Workflow', and 'Close Metadata Editor session'. Below this is a 'WORKFLOW' section with a table showing workflow details: State (Under-Enrichment), Type (EclapWorkflow), Target (ECLAP), Last User (N.A.), Last Access (2011-06-03 12:33:19), Current User (root), Start Session (2011-12-20 15:10:13), Native Metadata Number (4), Total Metadata Number (4), Validated Metadata (0), and Enriching Activities (0). At the bottom of the sidebar is a 'PREVIEW' section showing a small image of a man. The central 'METADATA PANEL' has tabs for 'DCMI', 'TAXONOMY', 'GROUPS', and 'PROPERTIES'. It is split into 'Reference' and 'Changes' views. The 'Reference' view shows metadata fields like 'dc.Title', 'dc.Creator', 'dc.Contributor', 'dc.Publisher', 'dc.Description', 'dc.Subject', and 'dc.Coverage' with their values and a 'By:' field. The 'Changes' view shows the same fields with icons for actions like 'undo', 'redo', 'delete', and 'add'. A red box highlights the 'Last EnricherUser' field in the top right, and a red circle highlights the 'By: serena' field in the 'Reference' view.

ECLAP Connecting stages

3.4.3 Metadata Editor Validation mode

The editor works in Validation mode for the WF Validation role. The editor can be used to edit/enrich/validate/invalidate metadata. Before starting a workflow transition is invoked according to the workflow model (transition Uploaded → Under-Validation). If the transition is done with success, this means no other user is working on the content, the workflow tables is updated with the user id and start session date and time, otherwise the content was locked by other user and the editor is opened in read-only mode just for browsing metadata. When user is authorized by the workflow he has full access to editing capabilities and metadata are displayed according to his language capabilities. The workflow box displays the current information.

The validator could Validate/invalidate single metadata or Validate/invalidate all metadata.

Validate All

The function allows the user to validate not yet validated metadata.

Apply Changes

The apply changes function allows the user to fix own metadata changes. The browser sends to the server the list changes (new, validate, invalidated, updated and delete metadata) and in event of success all panels are refreshed with the applied changes. The list of changes contains all information according to the data model.

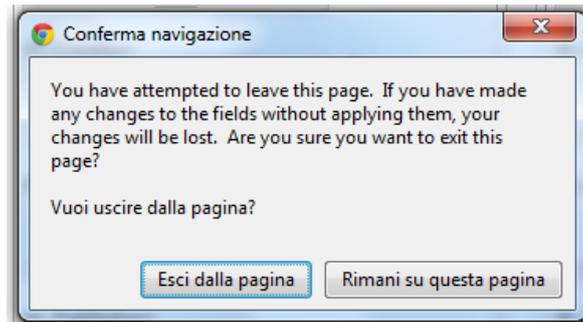
End session

The End Session function allows the user to end metadata editing/validating session. The page checks if there are changes to apply. In event of positive check a dialog is displayed to ask for apply changes before exit or to continue discarding all changes. Otherwise, the End Session is sent to the server

3.4.4 Leaving Metadata Editor

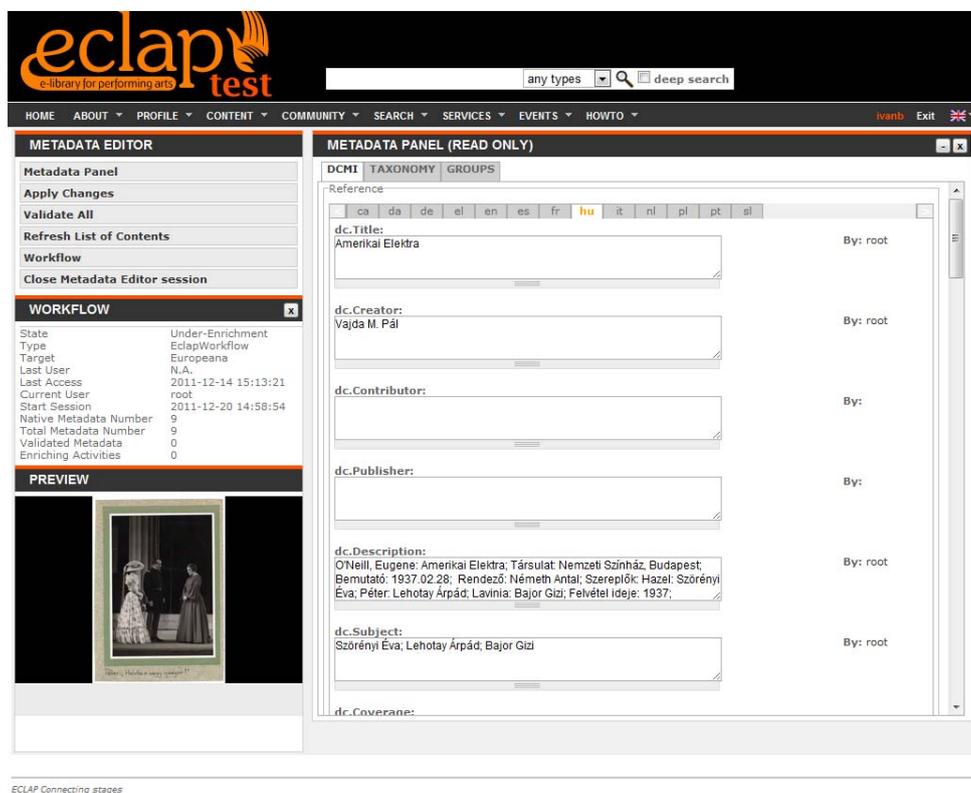
Since Metadata Editor is a web application, the user has to make some operations before leaving the page:

- Users have to apply all made changes before closing
- End Session has to be clicked to close and return back to Edit object page
 - If changes are pending a popup dialog is shown to remind to apply the made changes before exit or to discard them
- If users try to change page/exit a popup dialog reminds to apply the made changes or to discard them



3.4.5 Metadata Editor Read-Only mode

Only users with ECLAP workflow permission can work fully with the editor. Users without a workflow role access anyway to the editor only in read-only mode.



3.5 Functionalities of the ECLAP automated back office (Bruno - DSI)

For AXCP Grid installation and configuration please refer to the AXMEDIS AXCP tools manual.

3.5.1 Configuration of AXCP Grid Node

The AXCP Grid Nodes involved in content production have to be configured in order use some external tool, that are required to have the necessary capabilities in content processing. The following table reports the

third parts software that has to be installed in the *AXCPToolsDir* folder, where all AXCP Rules running on the node find the necessary tools.

Third-party software	Version	Notes
Imagick	6.6.6-3 q8	Image Processing library
FFMPEG + FLVMDI	SVN-R 13712	Video/Audio processing tool & metadata injection in FLV video file
MENCODER		Video/Audio processing tool
MP4Box		MP4 production tool
K-LITE CODECK Pack	6.6.0	Set of audio/video codec coming from AXCP installation
PSFtp		Command-line interactive SFTP client

Each AXCP Rule has to point to these folders in order to build the path for using the tool. Usually, these settings are into the *main.js* script as following:

```

/***** Settings *****/
var diskUnit="C:";
var ffmpegDir = diskUnit+"\\AXCPToolsDir\\ffmpeg";
var psFtpDir = diskUnit+"\\AXCPToolsDir\\psftp";
var imagemagickDir = diskUnit+"\\AXCPToolsDir\\magick";
var mencoderFolder = diskUnit+"\\AXCPToolsDir\\mencoder\\";
var mp4BoxDir = diskUnit+"\\AXCPToolsDir\\mp4box";
    
```

Each AXCP Grid Node has to be connected to the ECLAP database via ODBC connections. Such connections have to be defined on the computer hosting the grid node according to the IP address and credentials of the computer hosting the ECLAP Database. All running AXCP Rules can access to the “axcp.properties” and “axdbv4.properties” databases to load the configuration settings and read/write during the execution.

```

/*****
/*
/* Configuration Manager
/*
/*
/*****
var conf = new configuration();
conf.loadFromDB("axcp.properties");
conf.loadFromDB("axdbv4.properties");
    
```

3.6 Content and metadata production (Bruno - DSI)

In this section the AXCP rules involved in content and metadata processing are described

AXCP Rule	Notes
UploaderAXCP	Backoffice process on AXCP Grid for content production
UploaderAXCP4Devices	Backoffice process on AXCP Grid for content production on PDA devices
ExtractResource	Backoffice process on AXCP Grid for content production for

	IPHONE, IPAD and mobile
ProduceMDHDVideo	Backoffice process on AXCP Grid for content production for MD/HD Video
Automatic translation	Backoffice process on AXCP Grid for automatic metadata translation

3.6.1 UploadAXCP

The rule is the main process in the AXMEDIS AXCP GRID involved in content production and adaptation. The result of the process is an mp21 AXMEDIS object. This object is the PC version and is considered the *father/main-object*.

The AXCP Rule arguments are listed in the following table:

Argument	Description
fileUrl	the URL(s) to get the content(s) and a set of additional information organized as following: {'url':"\$fileUrl",'filename':"\$file",'mime':"\$type",'mode':"\$source",'localFolder':"\$folder"}
validate	Indicates if content has to be controlled by administrator and then stored into Administration database
owner	User and affiliation according to the forma →userId;Affiliation
metadataUrl	the URL to get the metadata XML file
protectedObject	Used for DRM purpose
noHttp	Enable/disable download via http
noP2p	Enable/disable download via p2p
drupalId	Not used
classification	the list of indexes related to taxonomy terms chosen by the user
privileges	the list of indexes related to groups terms chosen by the user
devices	List of derived object (i.e. PDA, MOBILE...)
reqId	Request id and caller identifier (i.e. php for portal, ingestion for the ingester)
device	Current device. The content will be generated by adapting to current device
op	It will be set to 'UPLOAD' when content is uploaded for the first time. It will set to 'UPDATE' when content will be modified
axoidRef	Empty or axoid related to the main object or to the content to be updated
wfType	Type of workflow associated with the content

Configuration options

The configuration options are loaded from the axdbv4.properties table and merged with the axcp.properties table.

Upload mode

The main activities of process when used in UPLOAD mode are:

1. **Content retrieval:** the *fileUrl* parameter is analyzed to get the URL of content. The content is downloaded and stored in the temporary local folder of the AXCP Grid Node
2. **Metadata retrieval:** the *metadataUrl* parameter is analyzed to get the URL of the metadata file and then it is downloaded and stored the temporary local folder of the AXCP Grid Node.
3. **Content adaptation:** an automatic format detection is performed to defining *format* and *type* and then select the proper content adaptation and transcoding process according to the Format and Type and Conversion Features tables
4. **Metadata organization:** the metadata file is parsed to extract metadata set and to generate the Dublin Core section associated with the content when stored as AXMEDIS mp21object.

5. **Extra metadata generation:** the user information (id and affiliation), taxonomy and groups choices are organized as xml generic metadata and stored in the mp21 object according to the AXMEDIS model. Performing arts metadata are also organized separately as xml generic metadata and stored in the mp21 object.
6. **Icon generation:** an animated gif or a single gif image is generated in event of video or image content. This image is used as icon associated with the content in ECLAP portal.
7. **Save content on Database:** the mp21 AXMEDIS object is stored in the AXMEDIS Database. If the validate parameter is set to false, the AXMEDIS database is the Publishing database (*axdbv4*), otherwise the content is stored in the Administrative database (*axdbv4f*). Contextually, the corresponding *rootobjectinfo* table is updated by adding a new row related to new content. The Rule sets the following fields of the *rootobjectinfo* table leaving the other parameters with the default value.
 - **rootObjectAxoid:** the Axoid of the mp21 object
 - **nohttp:** disable/enable http download (default 0)
 - **nop2p:** disable/enable p2p download (default 0)
 - **published:** the publication status (default 0 not published)
 - **sourcefile:** the location of raw resource
 - **sourceUrl:** the original location of the content
 - **defLanguage:** the language used for metadata in the upload form
 - **format:** the format of new digital content
 - **type:** the type of the new digital content
8. **Workflow Tables Update:** the *workflow_info* and the *workflow_metadata* tables are updated in order to store the status of metadata and workflow information. The UploadAXCP Rule sets the following fields of the *workflow_info* table leaving the other parameters with the default value:
 - **Axoid:** the id of AXMEDIS mp21 object
 - **md_number:** the number of inserted metadata
 - **md_original_num:** the number of inserted metadata
 - **type:** the workflow type associated with the contentThe UploadAXCP Rule sets the following fields of the *workflow_metadata* table leaving the other parameters with the default value. It inserts a row for each metadata:
 - **Axoid:** the id of AXMEDIS mp21 object
 - **Id:** the id metadata
 - **Dtable:** the database table where the metadata value is stored
 - **proposerUID:** the id of user who made the upload and provided metadata
 - **status:** 0 means PROPOSED (just uploaded and written for the first time)
 - **production:** 2 means UPLOADED (provided since the beginning via upload form, web page)
9. **Derived Content Production:** the AXCP Rule *UploadAXCP4Device* is invoked in order to generate mp21 AXMEDIS object to cope with different devices (i.e. pda, mobile, etc...) and passes the AXOID of the just created mp21 object. The AXOID is sent in the *axoidRef* parameter together with the input parameters of the rule.
10. **Drupal publication:** If the validate parameter is set to false, the UGC publishing service is invoked in order to publish the content on Drupal/ECLAP portal.
11. **Indexing request:** If the validate parameter is set to false, the indexing service is invoked to index metadata and content in the Publish Index Database.
12. **Extract Resource:** If the validate parameter is set to false, the AXCP rule *Extract Resource* for generating different resolution and formats raw files is invoked
13. **Automatic Translation:** If the validate parameter is set to false, the AXCP rule *Automatic Translation* for automatic metadata translation is invoked.

Ingestion mode

In the Ingestion scenario, when the URL of digital resource is available a full object (resource and metadata) is produced otherwise only metadata are processed an Empty Object is produced, the *hidden* flag in the *rootobjectinfo* table is set to 1 and steps 9, 12 in the UPLAOD flow are skipped.

Update mode

The main steps of process are:

1. **Content retrieval:** the *fileUrl* parameter is analyzed to get the URL of content. The content is downloaded and stored in the temporary local folder of AXCP Grid Node
2. **Content adaptation:** an automatic format detection is performed to defining *format* and *type* and then select the proper content adaptation and transcoding process according to the Format and Type and Conversion Features tables
3. **Mp21 Current version retrieval:** the mp21 AXMEDIS object of the current version is downloaded from the database by means the AXOID value stored the *axoidRef* parameter.
4. **Icon generation:** an animated gif or a single gif image is generated in event of video or image content. This image is used as icon associated with the content in ECLAP portal.
5. **Save content on Database:** the modified mp21 AXMEDIS object is re-stored in the AXMEDIS Database. The AXMEDIS database is the Publishing database (*axdbv4*). Contextually, the corresponding *rootobjectinfo* table is update. The rule updates the following fields of the *rootobjectinfo* table leaving the other parameters with the default/existing value.
 - **sourcefile:** the location of raw resource
 - **sourceUrl:** the original location of the content
 - **format:** the format of new digital content
 - **type:** the type of the new digital content
 - **hidden:** it set to 0
6. **Derived Content Production:** the AXCP Rule *UploadAXCP4Device* is invoked in order to update the mp21 AXMEDIS object already generated for different devices (i.e. pda, mobile, etc...).
7. **Indexing request:** The indexing service is invoked to re-index the updated content in the Publish Index Database.
8. **Extract Resource:** the AXCP rule *Extract Resource* for generating different resolution and raw files is invoked for the new content.

In all rules, when the digital resource is available, the rule performs an automatic format and type analysis according the **Format and Type** table. When needed, the digital content is adapted according to the **Conversion Features** table.

Format and Type table

Format	Type
Audio	Audio
Video	video
Document	<ul style="list-style-type: none"> • document • html • pdf • slide • excel
Archive	Archive
Image	image
Crossmedia	<ul style="list-style-type: none"> • audio • crossmedia • document • flash • html • image

	<ul style="list-style-type: none"> • pdf • slide • smil • video
Tool	Tool
Playlist	Playlist

Conversion Features table

Resource	Coding parameters	Output Format
Video	Size (512)x288 Video codec h263 Video bitrate 320kb Audio codec mp3 Audio bitrate 64kb	FLV
Audio	As original	Mp3
Image	As original	PNG
Icon	200x120	GIF

3.6.2 UploadAXCP4Device Rule

The UploadAXCP4Device rule is the process in the AXMEDIS AXCP GRID involved in content production and adaptation for generating content for different devices such as pda, mobile, stb.

The UploadAXCP4Device Rule works in two modalities: UPLOAD and UPDATE mode.

The UPLOAD mode is used when content is uploaded for the first time together with metadata. The UPDATE mode is used when the content has to modify by providing a new version that will replace the previous.

The AXCP Rule arguments are listed in the following table:

Argument	Description
fileUrl	the URL(s) to get the content(s) and a set of additional information organized as following: {'url':"\$fileUrl.",'filename':"\$file.",'mime':"\$stype.",'mode':"\$source.",'localFolder':"\$folder."}
validate	Indicates if content has to be controlled by administrator and then stored into Administration database
owner	User and affiliation according to the format →userId;Affiliation
metadataUrl	the URL to get the metadata XML file
protectedObject	Used for DRM purpose
noHttp	Enable/disable download via http
noP2p	Enable/disable download via p2p
durpalId	Not used
classification	the list of indexes related to taxonomy terms chosen by the user
privileges	the list of indexes related to groups terms chosen by the user
devices	List of derived object (i.e. PDA, MOBILE...)
reqId	Request id and caller identifier (i.e. php for portal, ingestion for the ingester)
device	Current device. The content will be generated by adapting to current device
op	It will be set to 'UPLOAD' when content is uploaded for the first time. It will set to 'UPDATE' when content will be modified
axoidRef	Empty or axoid related to the main object or to the content to be updated
wfType	Type of workflow associated with the content

Input type: Video, Image, Audio and PDF files.

Format and Types

Format	Type
Crossmedia	<ul style="list-style-type: none"> • Video • Image • Audio
Document	Pdf

Generated Formats

Video: resolution 240x180, video codec mpeg4, video bitrate 128kb, audio codec aac, audio bitrate 32kb output format mp4

Image: jpeg embedded in SMIL format

Audio: mp3 embedded in SMIL a file

Document: only pdf

Upload mode

The main activities of process when used in UPLOAD mode are:

1. **Content retrieval:** the *fileUrl* parameter is analyzed to get the URL of content. The content is downloaded and stored in the temporary local folder of the AXCP Grid Node
2. **Metadata retrieval:** the *metadataUrl* parameter is analyzed to get the URL of the metadata file and then it is downloaded and stored the temporary local folder of the AXCP Grid Node.
3. **Content adaptation:** Automatic format detection is performed to defining format and type and then selects the proper content adaptation and transcoding process (i.e. Mp4 for pda, SMIL for mobile).
4. **Metadata organization:** the metadata file is parsed to extract metadata set and to generate the Dublin Core section associated with the content when stored as AXMEDIS mp21object.
5. **Extra metadata generation:** the user information (id and affiliation), taxonomy and groups choices are organized as xml generic metadata and stored in the mp21 object according to the AXMEDIS model. Performing arts metadata are also organized separately as xml generic metadata and stored in the mp21 object.
6. **Save content on db:** the mp21 AXMEDIS object is stored in the AXMEDIS Database together with the main AMXEDIS Object. If the validate parameter is set to *false*, the AXMEDIS database is Publishing database (*axdbv4*), otherwise the content is stored in the Administrative database (*axdbv4f*). The rule updates the *rootobjectinfo* table by setting the *device* (pda, mobile, stb) column to 1 for the parent mp21 object by means the *axoidref* argument

Update mode

The main activities of process when used in UPDATE mode are:

1. **Content retrieval:** the *fileUrl* parameter is analyzed to get the URL of content. The content is downloaded and stored in the temporary local folder of the AXCP Grid Node
2. **Mp21 Current version retrieval:** the mp21 AXMEDIS object of the current version is downloaded from the database by means the AXOID received in the *axoidRef* parameter. If the device version is not available, the main AXMEDIS mp21 (the PC version of object is loaded. A new mp21 object is generated by copying the metadata of the pc version.
3. **Content adaptation:** Automatic format detection is performed to defining format and type and then selects the proper content adaptation and transcoding process (i.e. Mp4 for pda, SMIL for mobile).
4. **Save/update content on db:** the mp21 AXMEDIS object is stored in the AXMEDIS Database together with the main AMXEDIS Object. The AXMEDIS database is the Publishing database (*axdbv4*).

3.6.3 Extract Resource AXCP Rule

The Extract Resource AXCP Rule produces different resolution and formats raw files. Such files are generated from the original uploaded digital resource and they are accessible by different devices not supporting mp21 format (PDA, iPhone, Ipad, and Smartphone).

In event of video, the rule generates the iPhone video version and invokes the *LD/MD/HD video production AXCP Rule*.

The rule accepts as parameters:

Argument	Description
listAxoid	a list of AXOID related to contents to be processed. The list is formatted using semicolon as delimiter for each AXOID
params	It is a free parameter used to provide the database where axoids are stored: database=Publishing (or Administrative)

At the end of process, the rule updates the *rootobjectinfo* table by setting technical data as reported below:

filesecco= extension of file

pcview= 1 or 0. When 1 is set, the content can be played by a pc,

resolution= L or H depending by the original resource,

dx=width of video/image

dy=height of video/image

duration=duration of the fluid resource such as video, audio,

iphone=1 or 0 if the resource has been generated for Iphone i.e. video

In event of video resource the Iphone/Ipod version is generated according to the following features:

Video size: 640x(360)

Video codec mpeg4

Video bitrate: 512kb

Video fps: 25

Audio codec: aac 96kb

Output format container: mp4

If the original size and/or frame rate too low, the resource is converted maintaining them.

3.6.4 LD/MD/HD video production AXCP Rule

The LD/MD/HD video production AXCP Rule is invoked to generate the Medium Definition and the High Definition versions of a video.

The LD, MD and HD video are generated using the uploaded video file, and the MD/HD version is generated only if the height of the original video it is greater or equal to 480 for MD and 720 for HD.

	LD	MD	HD
Video size	(320)x288	(852)x480	(1280)x720
Video codec	h264	h264	h264
Video bitrate	300kb	800kb	2Mb
Audio codec	aac 96kb	aac 96kb	aac 96kb
Output format container	flv	flv	flv

The video width it is adapted to maintain the same aspect ratio as the source video on the basis of the video height.

The rule accepts as parameters:

Argument	Description
listAxoid	A list of AXOID related to contents to be processed. The list is formatted using semicolon as delimiter for each AXOID
params	It is a free parameter for future purpose

3.6.5 Automatic translation

The Automatic translation AXCP Rule produces different metadata in different languages automatically by using tool or web service for text translation. The rule translates original metadata according to the languages managed by the ECLAP portal.

The rule accepts as parameters:

Argument	Description
listAxoid	A list of AXOID related to contents to be processed. The list is formatted using semicolon as delimiter for each AXOID
params	It is a free parameter used to provide the database where axoids are stored: database=Publishing (or Administrative)

The main activity of the rule is:

1. **Workflow lock request:** a workflow request is done to ask for a status transition from the UPLOADED to the UNDER-AXCP status (according to workflow diagram).
 - a. If the transition is not performed and the *retryCount* is less than a threshold, the *retryCount* is incremented and the Automatic translation is resubmitted to the AXMEDIS Grid by scheduling it to a new time schedule and passing the new *retryCount* value. The current activity is stopped.
 - b. If the *retryCount* is greater than the threshold, the current activity is stopped and the automatic translation is aborted definitively. A notification mail is sent to the administrator.
2. **Metadata retrieval:** load of the metadata set stored in the database (DCMI table in the publishing AXMEDIS database).
3. **Original Language detection:** the original language used during the upload or content generation. It is stored in the *rootObejctInfo* table of publishing AXMEDIS database
4. **Metadata translation:** each metadata is translated for each default languages by accessing to the Google translation service or equivalent. If a metadata is already available in the language under translation, it is skipped.
5. **New Metadata insertion:** translated metadata is inserted in the publishing DCMI AXMEDIS database
6. **Workflow Tables Update:** the *workflow_info* and the *workflow_metadata* tables are updated in order to store the new translated metadata and workflow information as described in the workflow section. The Rule updates the following fields of the *workflow_info* table leaving the other parameters with the default value:
 - **md_number:** the number of inserted metadata

The AXCP Rule set the following fields of the *workflow_metadata* table leaving the other parameters with the default value. It will insert a row for each new translated metadata:

- **Axoid:** the id of AXMEDIS mp21 object
- **Id:** the id metadata
- **Dbtable:** the database table where the metadata value is stored
- **proposerUID:** the id of user who made the upload and provided metadata
- **status:** 0 means PROPOSED (just uploaded and written for the first time)

- **production:** 0 means GENERATED (produced by the AXCP Rule)

7. **Axmedis RootObjectInfo table update:** The rule update the *rootObejctInfo* table in the publishing AXMEDIS database by setting the value to true/1 of the *modified* flag field
8. **Workflow unlock request:** a workflow request is done to ask for a status transition from the UNDER-AXCP to the UPLOADED status (according to workflow diagram).
9. **Indexing request:** The indexing service is invoked to index/update metadata in the Publish Index Database.
10. **Automatic translation resubmit:** The resubmission is done in event of workflow transition failure or connection failure with Google.

Configuration options

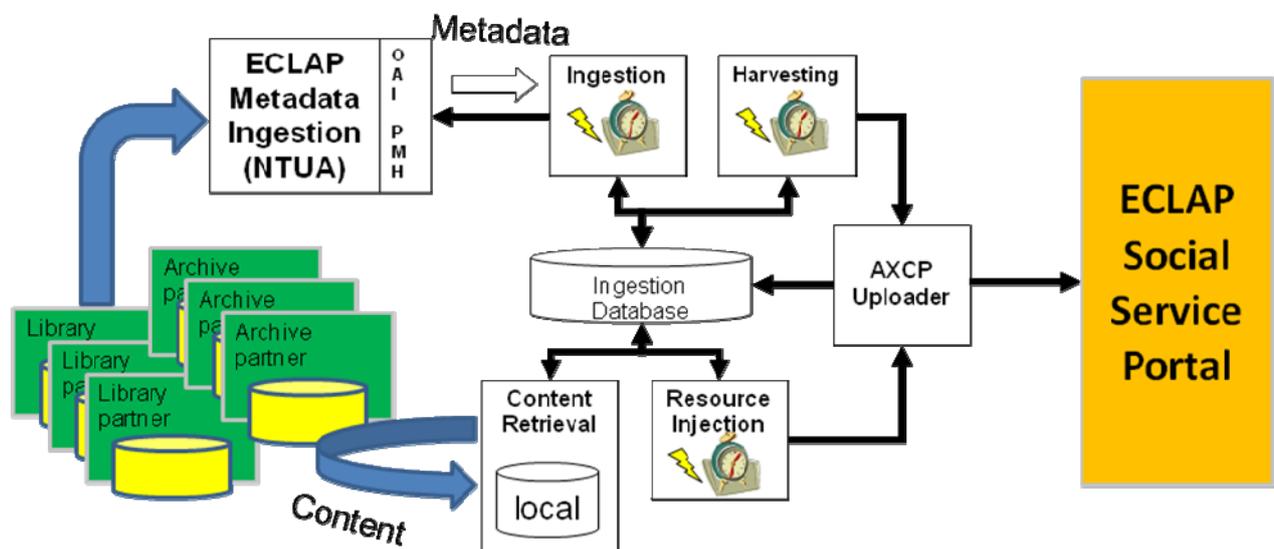
The configuration options is loaded from the axdbv4.properties table and merged with the axcp.properties table. This last table is used by all AXCP rule involved in content production and back-office activity.

3.7 Content & metadata ingestion (Bellini, Bruno - DSI)

Content and Metadata Ingestion is the process that allow importing metadata and content coming from ECLAP partners and Digital Archives. The process allows ingesting both massively and singularly metadata and digital resources.

It could be happen that some digital resource since too big to get via Internet has to be provided by using physical device, so that we have the need to start producing ECLAP content just with metadata and then when the digital resource is available injecting it off-line in the corresponding ECLAP content..

This section describes how the ECLAP portal will work with ECLAP Metadata Ingestion service by NTUA. The picture below describes the processes that are involved in content ingestion:



The activities are divided as following:

- Content and Metadata Retrieval: retrieving the metadata from the EMIS service
 - **Ingestion** is the process that monitors and gets metadata of new content or a new version of metadata for just retrieved contents.
 - **Harvesting** is the process that monitors for new retrieved metadata. It calls the AXCP Uploader process for content production and posting on ECLAP portal.

- **Content Retrieval:** is the process that retrieves and organizes contents coming from the partners via internet or via physical devices
- **Resource Injection:** it is the process that monitors for digital contents availability. It calls the content production to update the existing ECLAP contents just created only with metadata in order to inject/fill with the digital resource by the AXCPUploder process.

All processes share and update a Database of metadata (Ingestion Database) where all metadata and other information are stored and used to manage the whole life cycle of contents ingestion.

All processes are AXCP rules running on AXMEDIS AXCP grid as better described in the following sections.

3.7.1 Ingestion Database

The Ingestion Database will be used to store, trace and orchestrate all ingestion and production activities. Such database will be organized to refer:

- XML files containing structured metadata according to the ECLAP Schema or storing the whole XML as a TEXT
- URL/Location of digital Resource
- XSLT file of rules to map the original XML of metadata according to the ECLAP schema
- XML file of original metadata
- Flag status of production
- Timestamps: production and acquisition
- AXOID related to the generated content both Empty Object and full object
- contentID identifier of the of original content
- Library/Source the name of library/provider of content

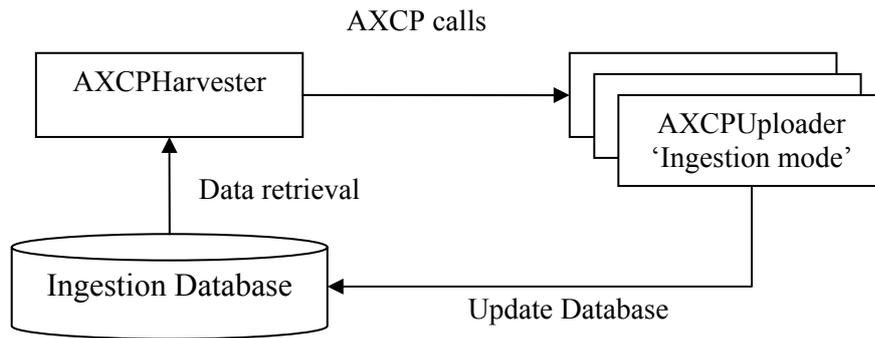
3.7.2 Content and Metadata Retrieval (Ingestion)

This process will be performed by an AXCP Rule called AXCP_OAI-PMH that periodically will

- Monitor and get metadata of new contents or a new version of metadata for just retrieved contents from EMIS.
- OAI_PMH access by NTUA service to get and store new or updated metadata. The response to this call will expect to get:
 - XSLT file of rules to map the original XML of metadata according to the ECLAP schema
 - XML file of original metadata
 - XML file related to the metadata mapped into the ECLAP Schema
 - The remote URL of the digital content
- Download the content if the URL of digital resource is available
- Organizing all collected information in the ingestion database

3.7.3 Content Generation from Metadata (Harvesting)

This process is performed by an AXCP Rule called *AXCPHarvester* that periodically query the ingested metadata database to search for new content. The generation process starts organizing the metadata collected by the database and structured according to the ECLAP Schema. A list of contents/references to metadata will be generated. For each item, the AXCP rule invokes asynchronously the *UploadAXCP* Rule (for more details please see the section of Web Upload) for content production by providing the URL of content file and the structured metadata file associated with content. The result of content generation is an AXMEDIS mp21 object that contain all metadata and the digital content. If any file URL is not available, the AXMEDIS mp21 object contain only metadata and for this reason this kind of “content” is called “*Empty object*”.



The UploadAXCP rule takes in charge the Ingestion Database update by setting:

- flag indicating that the content has been produced
- flag indicating that the digital resource is missing
- Setting date and time of production
- AXOID of the produce AXMEDIS mp21 object

3.7.4 Content Retrieval

During ingestion some metadata could be available before of the associated digital contents and the resource could be retrieved by means physical devices like hard disk or via authorized or free remote access on the web:

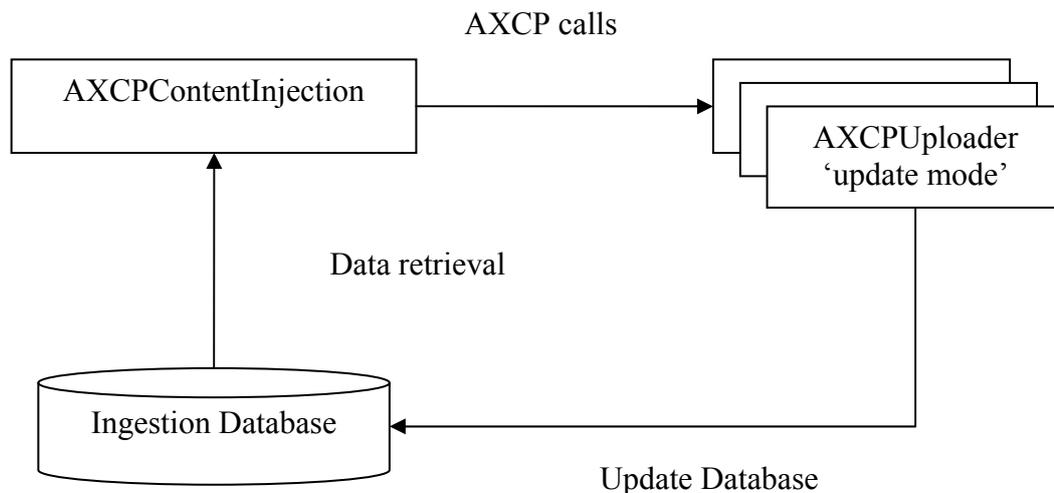
- **Direct access** to database via
 - ODBC, WS, REST, etc.
- **Files** + Excel file (or database) or XML
 - Via HD with db and content
 - Via FTP, HTTP, SFTP, etc.
- **Metadata mapping** is needed

In this case many Empty objects wait for digital content injection in order to complete their production.

The content retrieval from the network is performed by an AXCP Rule called *AXCPRetrieval*. This rule downloads the digital content via authorized or free remote access (via HTTP,FTP or from physical devices) and will build a list of pair <contentID>;<fileUrl> where contentID is the identifier of the metadata set associated with AXMEDIS id (AXOID) Empty Object. The rule updates the Ingestion Database by setting the content location (an URL or File System location).

3.7.5 Resource Injection

This process is performed by an AXCP Rule called *AXCPContentInjection*. This rule query the ingested metadata database to search the *fileUrl* related to Empty Object by analysing the status flag. For each “Empty object” the AXCP rule invokes the *UploadAXCP* Rule (for more details please see the section of Web Upload) for content production by providing the file *fileUrl* information. The *UploadAXCP* Rule is invoked in ‘Update’ mode.



3.7.6 Content management (Bruno - DSI)

In this section the AXCP rules involved in content management are described

AXCP Rule	Notes
updateMetadata	Back office process on AXCP Grid for updating metadata inside the mp21 version of content
deleteObject	Back office process on AXCP Grid for deleting content
produceIcon	Back office process on AXCP Grid for producing icon from content

3.7.7 Update Metadata

The rule generates a new version of mp21 content by injecting the current metadata stored in the Database.

The rule accepts as parameters:

Argument	Description
listAxoid	A list of AXOID related to contents to be processed. The list is formatted using semicolon as delimiter for each AXOID
params	It is a free parameter used to provide the database where axoids are stored: database=Publishing (or Administrative) Example of params: database=Publishing

3.7.8 DeleteObject

The delete object rule removes content from ECLAP/AXMEDIS databases

The rule accepts as parameters:

Argument	Description
listAxoid	A list of AXOID related to contents to be processed. The list is formatted using

	semicolon as delimiter for each AXOID
params	It is a free parameter used to provide the database where axoids are stored: database=Publishing (or Administrative) Example of params: database=Publishing

3.7.9 Producelcon

The Rule produces an icon for the digital content.

The rule accepts as parameters:

Argument	Description
listAxoid	A list of AXOID related to contents to be processed. The list is formatted using semicolon as delimiter for each AXOID
params	It is a free parameter used to provide parameters semicolon separated like the database where axoids are stored, the icon size: database=Publishing (or Administrative) iconW=width (i.e 200) iconH=height (i.e. 120) Example of params: database=Publishing;iconW=200;iconH=120

3.8 Content and metadata analysis (Fuzier – DSI)

Content and related metadata are processed to provide recommendations to user. There are three types of recommendations:

- User to user: each user is presented a list of “similar user” based on profile information and on content viewed information.
- Content to user: each user is presented a list of content he/she may like, based on content viewed by the user.
- Content to content: each time a user accesses content a list of “related objects” is shown.

3.8.1 User to user

User to user recommendations is provided through some AXCP rules and a Drupal module as shown by the following table:

name	type	description	dependencies
potentialFriends	Drupal module	Drupal module that allows the visualization of similar users list	Axfriends drupal module, axmedis drupal module, user_information drupal module, token drupal module, user_relationship_api drupal module

checkNeighbourly	AXCP rule	AXCP rule that checks the status of similar users lists	
CalculateAllNeighbourly	AXCP rule	AXCP that calculates the similar users list	CalculateAllNeighbourly

User to user recommendations are based on profile information stored for each user. For each user, among other, information about content viewed is stored: each user has a list of taxonomy values and formats of objects. Each taxonomy value and format value is associated with a weight which represents the percentage of objects viewed by the user with the corresponding taxonomy value and format value.

The list of taxonomy values and format values are updated dynamically each time an user views an object. When user to user recommendations are calculated the taxonomy values list and the format values list are evaluated.

The distance between taxonomy values lists is calculated using a matrix of distances between taxonomy values. The matrix is created calculating the minimum length of a path in the taxonomy graph between two values.

The distance between format values lists is calculated using a predefined matrix of distances between format values. Since there is not a great number of format values the matrix is simply created by associating to each couple of format values an arbitrary value.

The user to user recommendations are handled by the potentialFriends drupal module and by two AXCP rules: checkNeighbourly and CalculateAllNeighbourly. The drupal module generates the interface shown to the user while the AXCP rules are responsible of calculating the distances between users.

The generated list is shown by the potentialFriends module as a subpage in the user profile and as a block in the home page.

The AXCP rule checkNeighbourly is activated periodically by the scheduler every 24 hours. It selects the users which potential friends list is too old. It divides the selected list of users in fragments of 10 users and calls calculateAllNeighbourly for each fragment.

calculateAllNeighbourly is responsible of calculating the distances for each of the 10 users it receives. The calculateAllNeighbourly retrieves information about the user from the database to calculate the distances. It reads also some csv files where relationships between some information found in user categories are stored. For example the formats.csv maintains the similarity values between each possible object format, the languages.csv maintains the similarity between any couple of possible languages, and so on.

The checkNeighbourly AXCP rule has two internal parameters:

- logFilePath: says where the log file is located
- parameterFile: says where is located the XML file which contains the parameters to call calculateAllNeighbourly.

The calculateAllNeighbourly AXCP rule has one internal parameter, csvPath, which says where files containing the taxonomy distance matrix and the format distance matrix are located

3.8.2 Content to content

Content to content recommendations are provided by some AXCP rules and a Drupal module as shown in the following table:

name	type	description	dependencies
related_objects	Drupal module	Allows the visualization of the similar object list and builds the query for the non-based method of recommendation	Axmedis module

		generation	
DivideAndCallClusteringRules	AXCP rule	Checks the status of content to content cluster based recommendations	
ClusterObjects	AXCP rule	Clusters a reduced set of objects	
AddResourceToClosestCluster	AXCP rule	Assigns an object to the cluster more similar	

There are two ways currently implemented to provide content to content recommendation:

- Cluster based.
- Non-cluster based.

Both methods use textual information of each content: title, description, subject, body of HTML pages, text inside PDF documents, and so on. To use textual information both methods rely on the indexing system: the representation of each document is the same used by the indexing system, that is, a vector of words. Each word in the vector is weighted by a term that depends on term frequency among documents.

The cluster based method finds clusters of similar objects. Each time a user views an object, related objects are presented choosing randomly some other objects belonging to the same cluster.

Due to the great amount of data the clustering is done hierarchically. At first the objects are divided in “chunks”, groups sufficiently small of objects. Then every chunk is clustered and into every cluster a centroid is selected.

A centroid is an object which as the minimum mean distance from all the other objects in the cluster.

To the next level centroids are considered objects, so they are divided in chunk, each chunk is clustered and in each cluster a centroid is selected.

This process is repeated recursively until there is exactly one chunk left at the last level.

The last chunk is clustered and the clusters generated are those considered for the objects. An object belongs to a cluster if there exists path composed by centroids in the tree generated by the process from the leaf (the object) to the cluster created in the last level.

The non-cluster based method uses the vector of words representation of the object, and feeds as a query it to the indexing system, which returns an ordered list of similar objects.

The site administrator can choose which recommendation method to use in the administration section of the portal.

This functionality is provided by three AXCP rules – DivideAndCallClusteringRules,

AddResourceToClosestCluster and ClusterObjects – which are responsible of the distances calculation, and by a Drupal module – related_objects module – which is responsible of generating the user interface.

A third AXCP rule, ClusteringCleaner is used to reset the clustering database.

The DivideAndCallClustering rule is periodic: every time it starts it checks the clustering status. The ClusterObjects rule is called by DivideAndCallClusteringRule. It receives a “chunk” of elements and it divides them in clusters of similar objects.

In particular the DivideAndCallClusteringRule:

- Checks if there are errors in ClusterObjects rules called previously. If there are it calls the corresponding rules again.
- Counts how many not clustered objects there are and calls some ClusterObjects rules to assign them to clusters. The objects left to be clustered are divided in chunks. It calls as many ClusterObjects rules as the number of chunks left, limited by a maximum of 10 rules that can be executed together.
- Eventually, if there are no more objects, it checks if there are other levels. If there are it raises the level of operation.

The ClusterObjects rule:

- Selects the objects to be clustered
- Generates the data file for clustering
- Calls the AxmedisClustererDef.jar file. It uses the third party software Rapidminer to generate the clusters based on a custom distance.
- Writes on the database the results and his status.

The AddResourceToClosestCluster rule is called whenever an object is added to the portal or by the DivideAndCallClustering rule if the clustering pyramid is complete but there are some objects at level zero not assigned to clusters and not sufficient to create another chunk. This rule calls AddResourceToClosestCluster.jar which assigns the object to the cluster with the more similar centroid. These objects are “marked” so that when they reach the number necessary to form a chunk, the DicideAndCallClusterin rule can assign them to a chunk and call the corresponding ClusterObjects rule. The pyramid is then refreshed starting from level 1.

The related_objects drupal module shows the list of objects belonging to the same cluster of the one viewed by the user.

DivideAndCallClustering rule has some internal parameters:

- inputFileDir: says where files containing object info are located.
- outputFileDir: says where log files for the ClusterObjects rules should be located.
- parameterFile: says where is located the parameter file necessary to call the ClusterObjects rule.
- logFile: says where should be located the log file.
- rulesLimit: number of clusterObjects rules that can be executed together.
- Method: indicates the clustering algorithm chosen. Its value can be “Agglomerative” or “DBScan”.
- numClusters: Number of clusters at each level and at the end of the process. Used only if method is “Agglomerative”
- chunkSize: number of objects in each chunk.
- epsilon: parameter for the clustering algorithm “DBScan”.
- rulesTimeout: time after which the ClusterObjects rules that have been called but haven’t completed the execution are considered in an error state and must be restarted.

The ClusterObjects rule has some internal parameters:

- jarFile: states where is located the AxmediClusterer.jar file.
- processFile: path of the process file needed by Rapidminer to cluster the objects. The process file specifies some parameters like the algorithm used.
- taxonomyFile: file path of the taxonomy.csv file, containing the similarities between each taxonomy term.
- rapidMinerHome: path of the local Rapidminer install.

3.8.3 Content to user

Content to user recommendations are provided by some AXCP rules and a Drupal module as shown in the table below:

name	type	description	dependencies
user_recommendations	Drupal module	Allows the visualization of user recommended objects list	Axmedis Drupal module
checkUserRecommendations	AXCP rule	Checks the user recommendations status	
UserRecommendations	AXCP rule	Calculates user	

		recommendations	
--	--	-----------------	--

Content to user rely on the cluster-based method for content to content recommendation. The taxonomy values list and the format values list of each user are compared with each centroid of each cluster. The similarity between taxonomy values list and centroid taxonomy value are calculated using a taxonomy distance matrix, as for user to user recommendation. The similarity between the format values list of the user and the centroid format are calculated using the format distance matrix. The user is then presented with a list of objects belonging to the cluster of the centroid more similar to the user.

This functionality is provided by two AXCP rules – checkUserRecommendations and UserRecommendations – which are responsible of the back-end distances calculation, and by a Drupal module – user_recommendations - which is responsible of generating the user interface.

The checkUserRecommendations rule is periodic and calls the UserRecoomendations rule, passing it a list of 10 users.

The UserRecommendation rule calculates, for each user, the distance whit each centroid of the computed clusters. The cluster with the centroid more “similar” to the user is chosen to recommend objects to the user.

The “suggested content” list in the home page is generated by the user_recommendation_recommended_content PHP function in user_recommendation module. This function selects randomly objects from the cluster closer to the user and returns the rendered HTML list.

4 ECLAP Portal users and groups (Michela Paolucci - DSI)

New services have been developed both for single users and for groups. In order: the Contact Ingestion and the Content Management, that will be described in the following paragraphs.

4.1 Contact Ingestion

This service is available for all the logged users. The main functionalities are present and describe in the following paragraphs: *Import contacts from e-mail providers and Social Networks, Import contacts from file, Import contacts view.*

These functionalities have been developed in order to import contacts that each user has previously registered in other modalities, than manage them in order to invite new users in the Portal or to see if one of the contacts is just present in the Portal.

4.1.1 Import contacts from e-mail providers and Social Networks

In this case the user has to select the e-mail provider or the SN from the list (FIG.2) and fill in the requested fields. Moreover you can view all the previous operations (data, number of Imported contacts, New contacts -> Means not enrolled to the Platform – basing on their e-mail). The path is: 'Profile > view my Profile > Import contacts'.

Provider	Date	Imported contacts	New contacts
CSV	2011-07-08	6897	337
CSV	2011-07-08	1055	22
CSV	2011-07-06	1	0
CSV	2011-07-03	6900	4012
CSV	2011-07-03	109	3
CSV	2011-07-03	96	40
GMail OAuth	2011-06-30	7	3
twitter	2011-06-30	20	20
Facebook OAuth	2011-06-30	48	21
CSV	2011-06-30	1058	506
Linkedin OAuth	2011-06-30	414	412

Fig. 5.1

4.1.2 Import contacts from file

In this case each user has to select the text (csv, txt, dat...) containing mail address of contacts to be imported in the platform (see Fig. 5.2), then can go to the ‘Imported Contact’ page described in the following paragraph. The path is: ‘Profile > view my Profile > Import from file (any kind)’

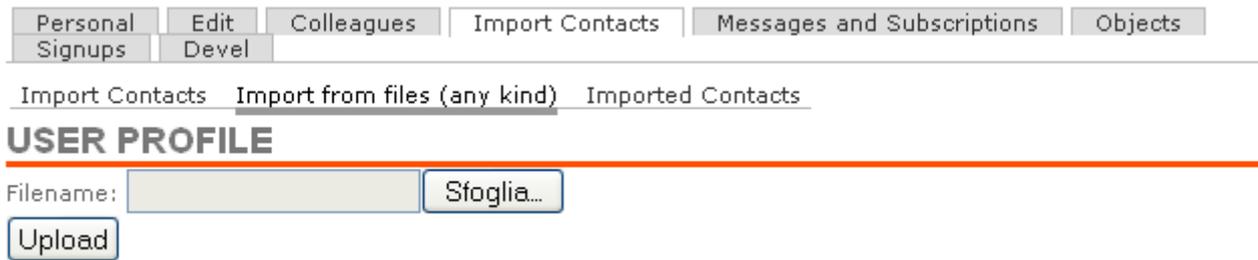


Fig. 5.2

4.1.3 Import contacts view

Each user can see the contacts imported in the platform (see Fig. 5.3): they are visible in a table format in which each row represents a contact and the columns the most relevant details related to the contact.

Table columns:

- **e-mail**
- **name**
- **surname**
- **nickname** (present if enrolled in the platform)
- **state** of each contact in the platform:
 - o *present*. The contact is registered to the Platform;
 - o *invited*. The contact is not registered to the Platform and the user has sent to him/her an invitation via e-mail;
 - o *connected*. The contact has made a request to be enrolled in the site and, if it is provided in the settings of the platform, is awaiting for the approval from the site administrator.
 - o *new*. The contact is not registered to the Platform;
 - o *new_nomail*. The contact is unknown and is classified as not registered in the platform (no e-mail is not available).



Fig. 5.3

4.1.4 Dependencies

(Content Management Dependencies)

Internal component	Type (drupal module, AXCP rule, Java Servlet, Other)	Notes
OpenInviter	Drupal module	

Third-party software	Version	Notes
OpenInviter		Php library

4.2 Content Management in groups

Given the large amount of contents uploaded on the portal, the development of a service that would allow the group managers to manage a set of contents (and not one by one) was deemed essential.

The Content Management service developed, offers three main functionalities that will be detailed here after:

Before proceeding in the description of individual functionality is useful to recall some present relations between users, groups, and affiliated partners to the project:

- many groups are available in the platform: groups for discussions (create basing on the same interests), technical groups (for the site maintenance), groups associated to ECLAP affiliated Partners
- each affiliated Partner has its group in the platform
- each user has to compile his/her personal profile in the platform specifying the affiliation (one affiliation for each user)
- a content is uploaded by a user or by an affiliated partner (one content is automatically associated to one publisher: the affiliated partner)
- each content can be **uploaded** by only one affiliated partner but can be **associated** to more than one partner (mean that will be retrievable in more than one group object list)

Moreover each object in ECLAP has (at least) the following features:

- o has a title of the object;
- o **Hidden**. ECLAP content can be hidden/not hidden. It is hidden when it is not indexed: it does not appear in the lists of ECLAP contents (e.g. Last posted, Featured, Popular etc.);
- o **WF type**. ECLAP content can be assigned to different workflow types. Now the WorkFlow (WF) available type are: Internal, ECLAP, Test, Europeana;
- o **Void content**. ECLAP content can be void/not void. An object is void if only the metadata has been uploaded in the ECLAP Portal without the real content (e.g. in the portal are present only metadata but not the image or the pdf/video, etc.);
- o **Published in Europeana**. ECLAP content can be just published in Europeana or NOT
- o **IPR Model applied**, To each ECLAP content can be applied an IPR Model
- o **IPR model type**. A model can be public or not public depending on the permissions and on the license chosen to access to it;
- o **Published by**. The Affiliated Partner from which the content has been updated;
- o **WF state**. A specific wokflow has been designed in order to better organize and manage the operations that have to be done on the objects, before classify them as collected by Europeana. So a set of WF states have been created (Uploaded, Under-Enrichment, Under-IPR, Under-AXCP, Under-Validation, Under-Approval, Published)

The service is accessible only for the Group Managers (GM) and for the site administer. The path is: “Group Page > Objects > Manage Objects”.

<input type="checkbox"/> Content	hidden	WF type	Void content	Published in Europeana	IPR model applied	IPR model type	Published by	WF state
<input type="checkbox"/> il maschio prepotente	not hidden	ECLAP	not void	not published	PrivateAsNow	not public	CTFR (Coherent)	Uploaded
<input type="checkbox"/> Dario Fo in Storia di una tigre	not hidden	ECLAP	not void	not published	None	NONE	CTFR (Coherent)	Uploaded

Fig. 5.4

4.2.1 View the contents

As Fig.5.4 shown, when a Group Manager goes to the ‘Manage Objects’ page, can access to the list of contents *associated* to his/her group. The list of objects is organized as a table in which the rows represent items in columns and lists the characteristics deemed most important for content management (classification, search, update content, etc.).

- **Content.** Is the title of the ECLAP content;
- **Hidden.** (indexed or not);
- **WF type.** Workflow type related to each ECLAP content (Internal, ECLAP, Test or Europeana);
- **Void content.** To each ECLAP content can be void/not void;
- **Published in Europeana.** ECLAP content can be just published in Europeana or not;
- **IPR Model applied.** an IPR can be applied to each ECLAP content;
- **IPR model type.** A model can be public or not public;
- **Published by.** The Affiliated Partner from which the content has been updated;
- **WF state.** The workflow state in which the ECLAP content is (Uploaded, Under-Enrichment, Under-IPR, Under-AXCP, Under-Validation, Under-Approval or Published).

4.2.2 Filter the contents

As Fig.5.4 shown, when a Group Manager goes to the ‘Manage Objects’ page, can access to the list of contents *associated* to his/her group and also filter the list. The filter is really useful in order to selected set of objects having the same features. The Group Manager can use one or more filter in order to have detailed lists of objects. The Group Managers can filter the objects on the basis of following features:

- **Hidden.** (indexed or not);

- **WF type.** Workflow type related to each object (Internal, ECLAP, Test or Europeana);
- **Void content.** An object can be void/not void;
- **Published in Europeana.** An object can be just published in Europeana or not;
- **IPR Model applied.** To each object can be applied an IPR;
- **IPR model type.** A model can be public or not public;
- **Published by.** The Affiliated Partner from which the content has been updated;
- **WF state.** The workflow state in which the object is (Uploaded, Under-Enrichment, Under-IPR, Under-AXCP, Under-Validation, Under-Approval or Published).

4.2.3 Make actions on the content

Each GM can select a list of objects and apply to the content selected an action. A list of actions is present in the platform, not all the actions are available for all the Group Managers.

As declared in the previous DE the platform is structured in way in which the users are organised/classified assigning them one (or more than one) role, to each role correspond a set of permissions and action that can be done in the platform. The Group Manager is one role and have some permission related to actions that can be done in the groups to organise the member, moderate the discussions, access to the ‘Manage Objects’ page, etc. Other fundamental roles (in addition to the “Group Manager”) that are relevant to go in details in this description are: ‘Publisher’, “IPR manager” (regarding a specific Affiliated Partner), ‘site administer’.

- Actions available for all the GMs:
 - o **Set to hidden** (eg: set all the objects selected in the check list to ‘hidden’)
 - o **Set to not hidden**
- Actions available for all the GMs that also have the ‘Publisher’ role:
 - o All the previous actions
 - o **Publish to Europeana**
 - o **Set the WF type to Internal**
 - o **Set the WF type to ECLAP**
 - o **Set the WF type to Europeana**
 - o **Set the WF type to Test**
- Actions available for the site administer on all the groups of the platform:
 - o All the previous actions
 - o **Delete object**
 - o **Copy metadata from DB to object**
 - o **Copy metadata from DB to object**
 - o **raw resource and xml extraction and video production (iPhone,LD,MD,HD)**
 - o **video production (LD,MD,HD)**

Moreover an additional action will be provided (now is not available), to the Group Managers that also have the “IPR manager” role of the objects affiliated to their group, is the following one:

- **Apply the same IPR model**

4.2.4 Dependencies

(Content Management Dependencies)

Internal component	Type (drupal module, AXCP rule, Java Servlet, Other)	Notes
iprWizard	Drupal module	Custom module
axmedis	Drupal module	Custom module
eclapworkflow	Drupal module	Custom module
workflow	Drupal module	Drupal core module

*DE3.3.1– Infrastructure: Content and Metadata processing and semantification
Best Practice Network*

node	Drupal module	Drupal core module
Copy metadata from DB to object	AXCP rule	
Copy metadata from DB to object	AXCP rule	
raw resource and xml extraction and video production (iPhone,LD,MD,HD)	AXCP rule	
video production (LD,MD,HD)	AXCP rule	

5 Eclap User Manual

For ECLAP User Manual please refer to “ECLAP Best Practice Network User Manual, This portal and services”, ANNEX of DE 3.1: “Infrastructure: ingestion and processing content and metadata”, at <http://bpnet.eclap.eu/drupal/?q=en-US/home&axoid=urn%3Aaxmedis%3A00000%3Aobj%3Ab828710e-b77c-4074-993c-3efddfbfaad7>

6 Multilingual taxonomy

This section describes the multilingual taxonomy editor. In D3.2: *Accessibility and Multilingual support for the ECLAP solution*, the rationale and the functionalities of the multilingual taxonomy editor were described in detail. This section presents an updated version of this.

6.1 Purpose & scope

The purpose of the multilingual taxonomy editor is to offer an interface for users to translate/validate terms in the ECLAP taxonomy into one of the 13 defined languages for the portal. These 13 languages are: Danish, Polish, Slovenian, Greek, English, Italian, French, Dutch, Spanish, Hungarian, German, Portuguese and Catalan.

6.2 Software architecture

As the ECLAP portal is implemented using the Drupal framework version 6, the modules used for the Multilingual Taxonomy are all Drupal 6 compliant. In the table below is an overview of the standard, freely available Drupal modules that have been used for implementing the functionalities of the Multilingual Taxonomy.

Module name	URL
Drupal Internationalization Module	http://drupal.org/project/i18n
Drupal Taxonomy Manager	http://drupal.org/project/taxonomy_manager

Next to these standard modules, a custom module was implemented for the export module described in 7.4.2. This module was called the Multilingual Taxonomy Export module and was also implemented to be compliant with Drupal version 6.

6.3 Procedures for creating & updating taxonomies

This section describes the rationale of the workflow needed for creating a professional multilingual taxonomy for the languages defined for ECLAP. The workflow is shown in the table below:

#	Action	Description
1	Taxonomy working group	The first step is to assemble a working group of professionals from the field of performing arts. The goal of the working group is to define the structure and terms of the taxonomy. In ECLAP these professionals are members from the ECLAP consortium.
2	Create initial taxonomy	After the definition of the taxonomy by the working group, the technical staff of the ECLAP portal imports the taxonomy into the system. The terms in this taxonomy are written in English.
3	Assign translators	In order to translate the taxonomy into the 13 different languages, for each language a translator has to be assigned. In ECLAP, each consortium partner representing a language assigns a translator for this purpose.

4	Translate taxonomy	Each translator is given a login with the privilege to translate the taxonomy into the assigned language.
5	Finalize translation	Whenever the taxonomy has been translated, the system administrators of the ECLAP portal will make the functionalities of the taxonomy available to be used in the portal (for annotation and searching).

6.4 Functionalities

This section describes the functionalities that were implemented for the Multilingual Taxonomy.

Editor

The main functionality described here is the editor, which allows users to translate terms from the taxonomy into one of the 13 languages defined for ECLAP. As mentioned in section 7.2, the Multilingual Taxonomy editor is freely available in the form of two Drupal modules. The Taxonomy Manager module offers an advanced user interface for editing Drupal vocabularies [1]. The Internationalization Module enables Drupal vocabularies to be multilingual. Also when the Internationalization Module is enabled, the Taxonomy Manager is extended with user interface elements for editing the translations of taxonomy terms. In this section only the functionalities related to the work of the translators are addressed.

Functions that are only available for administrators (the technical staff of ECLAP) such as adding/deleting terms in the taxonomy are not described in this section (see 7.1. Purpose & scope).

The initial screen of the editor is displayed in the figure below:

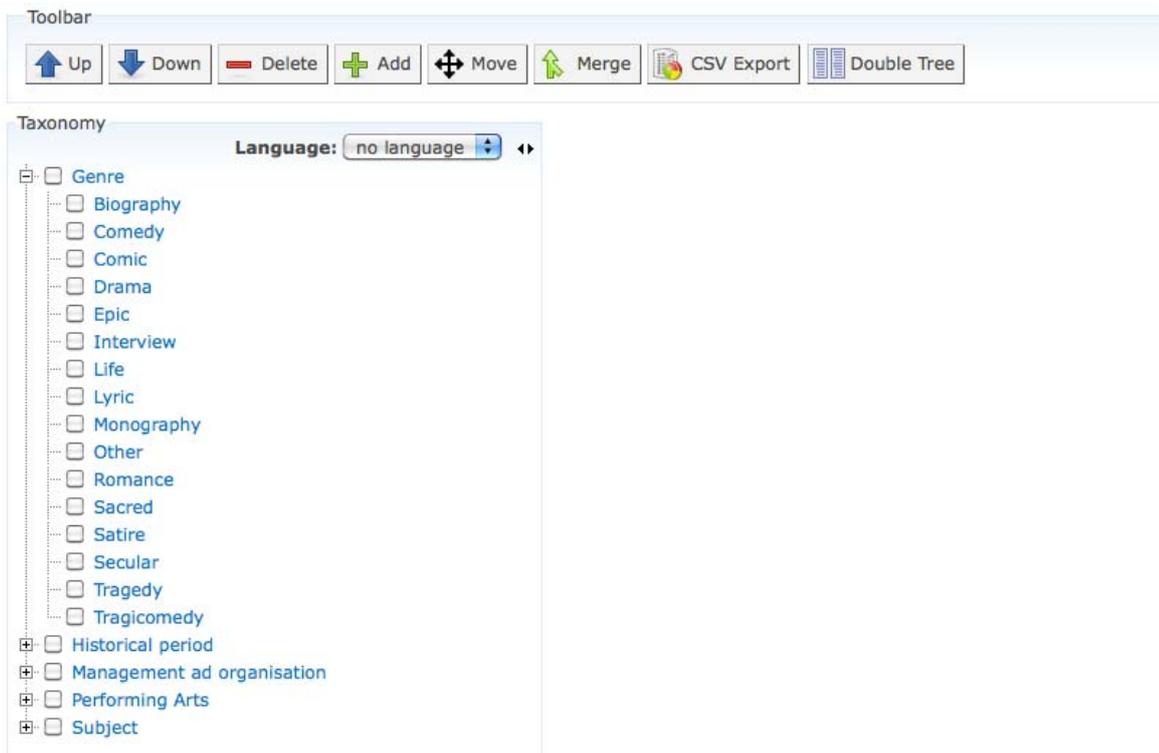


Figure 6-1 Multilingual Taxonomy Editor: show taxonomy translations

As seen in the figure, the user can select a language from the pull-down list. The list contains all of the 13 languages defined for ECLAP. By default the taxonomy has no language assigned to it, so selecting a different language will result in showing an empty taxonomy. Whenever a translator wishes to edit a translation it is advised to keep *no language* selected, as it will always show the full taxonomy.

In order to edit a term, a user simply has to select a term from the tree. In the figure below the term *Biography* is selected and a new panel containing a form appears on the right side.

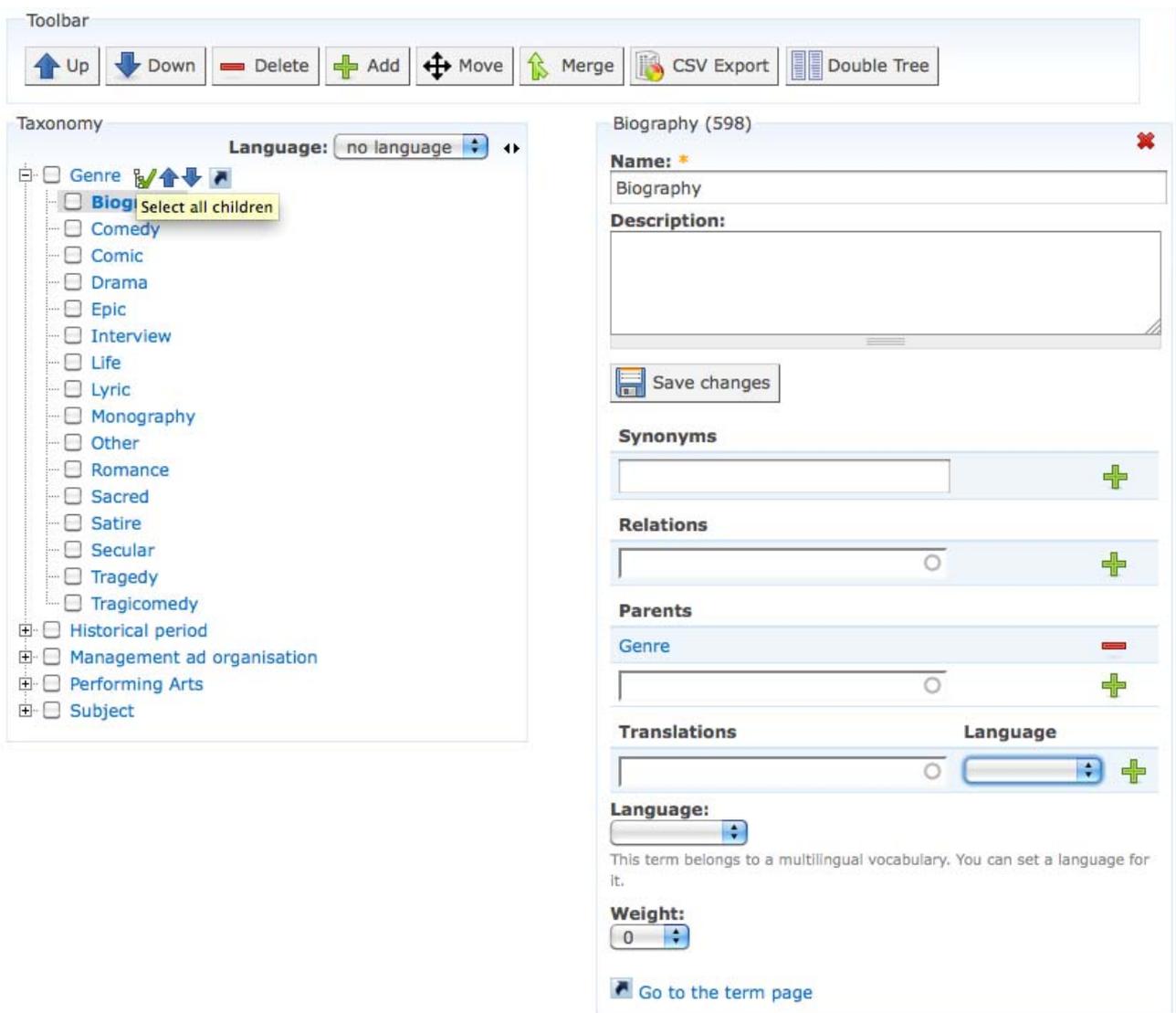


Figure 6-2 Multilingual Taxonomy Editor: edit terms

In this new panel, the information of the term is loaded. By using the form, a user can edit the following properties:

Name	The name/translation of the selected term
Description	Not used in ECLAP

Synonyms	Not used in ECLAP
Relations	Not used in ECLAP
Parents	Parent term of the selected term (only administrators can edit this)
Translations	Translations made for the selected term
Language	Language of the selected term
Weight	Not used in ECLAP

An example of a term having multiple translations is shown in the figure below:

Animal (737)

Successfully added translation

Name: *

Animal

Description:

Save changes

Synonyms

Relations

Parents

Subject

Translations	Language
Dier	Dutch
Tier	German
Dyr	Danish
Állat	Hungarian
Animal	Portuguese, Portugal
Zwierząt	Polish
Animal	French

Language:

English

This term belongs to a multilingual vocabulary. You can set a language for it.

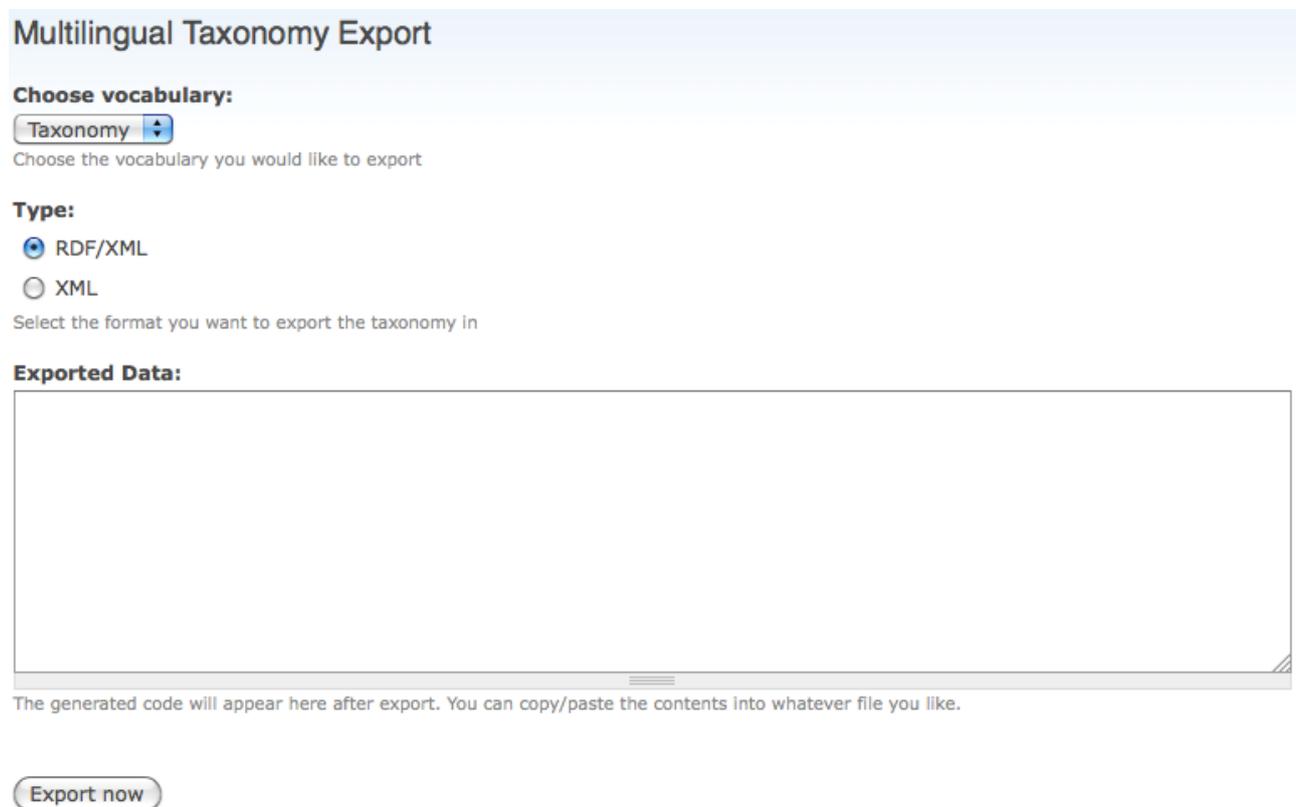
Figure 6-3 Multilingual Taxonomy Editor: Adding translations

Export module

The export module is a custom made Drupal module that was implemented with the goal of having the possibility to export the multilingual taxonomy into XML or RDF/XML, a functionality the standard Drupal modules do not support.

When performing arts institutes are interested in ECLAP, it can be useful to be able to offer an XML or XML/RDF file to experiment with the ECLAP taxonomy.

The export module can be found in the standard content management menu that is present in Drupal 6 installations. Only users with the administrator role can access the module. The export module page is shown in the figure below:



Multilingual Taxonomy Export

Choose vocabulary:
Taxonomy
Choose the vocabulary you would like to export

Type:
 RDF/XML
 XML
Select the format you want to export the taxonomy in

Exported Data:

The generated code will appear here after export. You can copy/paste the contents into whatever file you like.

Export now

Figure 6-4 Multilingual Taxonomy Export module

As shown in the figure, the user can select a Drupal vocabulary to be exported into RDF/XML or XML. There is however one prerequisite: the vocabulary must be configured in such a way that it supports multilinguality.

After choosing either RDF/XML or XML the system will fill in the text area with the exported data in the chosen format:

Exported Data:

```
<?xml version="1.0" encoding="utf-8"?><rdf:RDF xmlns:eclap="http://eclap.eu/taxonomy#" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:skos="http://www.w3.org/2004/02/skos/core"><rdf:Description rdf:about="http://eclap.eu/taxonomy#Biography"><dc:title xml:lang="en">Biography</dc:title><skos:prefLabel>Biography@en</skos:prefLabel><dc:title xml:lang="nl">Biografie</dc:title><skos:prefLabel>Biografie@nl</skos:prefLabel><skos:broader>http://eclap.eu/taxonomy#Genre</skos:broader></rdf:Description><rdf:Description rdf:about="http://eclap.eu/taxonomy#Animal"><dc:title xml:lang="da">Dyr</dc:title><skos:prefLabel>Dyr@da</skos:prefLabel><dc:title xml:lang="de">Tier</dc:title><skos:prefLabel>Tier@de</skos:prefLabel><dc:title xml:lang="en">Animal</dc:title><skos:prefLabel>Animal@en</skos:prefLabel><dc:title xml:lang="fr">Animal</dc:title><skos:prefLabel>Animal@fr</skos:prefLabel><dc:title xml:lang="hu">Állat</dc:title>
```

Figure 6-5 Multilingual Taxonomy Export: exported RDF/XML data

Exported Data:

```
<?xml version="1.0" encoding="utf-8"?><taxonomy><term id="http://eclap.eu/taxonomy#Biography"><title>Biography</title><language>en</language><title>Biografie</title><language>nl</language><parent>http://eclap.eu/taxonomy#Genre</parent></term><term id="http://eclap.eu/taxonomy#Animal"><title>Dyr</title><language>da</language><title>Tier</title><language>de</language><title>Animal</title><language>en</language><title>Animal</title><language>fr</language><title>Állat</title><language>hu</language><title>Dier</title><language>nl</language><title>Zwierząt</title><language>pl</language><title>Animal</title><language>pt-pt</language><parent>http://eclap.eu/taxonomy#Subject</parent></term><term id="http://eclap.eu/taxonomy#Biology"><title>Biology</title><language>en</language><title>Biologie</title><language>nl</language><parent>http://eclap.eu
```

Figure 6-6 Multilingual Taxonomy Export: exported XML data

7 Europeana module

DE3.1: *Infrastructure: ingestion and processing content and metadata* described the functionalities of the Europeana module (in section 7.5: *Searching on Europeana*). This section briefly summarizes the main points mentioned in DE3.1 and subsequently describes future plans and new updates in detail.

The purpose of the module is to offer users of the portal the possibility to discover the richness of the content on the portal of Europeana <http://www.europeana.eu>, to which ECLAP content will also be aggregated. The module was implemented according to the descriptions of DE3.1 and resulted in a widget that was integrated into the portal. This widget works in the way that whenever a user searches for content on the ECLAP portal, the Europeana module, or widget, shows relevant results from the Europeana portal (see Figure 8-1 below).



Figure 8-1. Europeana widget on the ECLAP portal

7.1 Future plans

As the Europeana API was recently updated with several interesting features, updates are planned for the Europeana module in order to be able to offer these features also to visitors of the ECLAP portal.

Show nearby content

When available, the Europeana API now also exposes the geographical coordinates (see next section: *Improve search*) of items in its collection. Using this information, the Europeana widget will be extended with an additional panel showing Europeana objects (related to the user's search terms) that originate from collections and / or institutions near the geographical location of the ECLAP user. The user's location is detected by using new standard functionalities available in HTML5 [2]. Using the HTML5 standard, the user's privacy is ensured as the user is asked for permission to share his or her location (See figure 8.1-1). Only if the user accepts to share his or her location, the widget will fetch the localized data from Europeana.

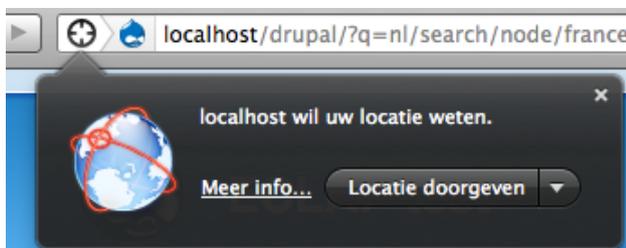


Figure 8.1-1 Browser asks the user for permission to share his/her location

Improve search

With respect to searching, the Europeana API has been enhanced with the possibility of querying semantically enriched metadata. The AnnoCultor [3] tool was used to enrich the geographical and temporal metadata fields, for Europeana being the Dublin Core [4] fields below:

- dcterms:spatial (e.g. having the value: 'Venice')
- dc:date fields (e.g. having the value '15e siècle')

The way the AnnoCultor tool operates is that for each of the above annotations it tries to look up corresponding terms in specialized databases, called vocabularies, of places and periods. In this sense for geographical locations GeoNames [5] is used and for periods the AnnoCultor Time Ontology [6] is used. After linking annotations with corresponding terms from the vocabularies, AnnoCultor retrieves additional information related to these terms, enriching the Europeana metadata with the following:

- **Place labels:** βενετία, velence, венеция, Venice, etc (for 'Venice')
- **Broader place labels (e.g. region, country):** Italy, Italie, olaszország (for 'Venice')
- **Geo coordinates:** 45.43861, 12.32667 (for 'Venice')
- **Time labels:** 15th, 15de eeuw, 15e siècle, 15й век, etc (for '15e siècle')
- **Broader time labels (e.g. century):** end of the 20th century, 2nd millennium (for '1980')
- **Temporal endpoints:** 1401-01-01T00:00:00Z, 1500-12-30T23:59:59Z (for '15e siècle')

Having the multilingual place labels, it's possible to find objects across languages, e.g. searching for 'венеция' in Russian makes it possible to find objects that were originally annotated with 'Venice' in English. Having broader place labels allows objects to be found via a broader geographical context, e.g. an object with location 'Venice' can be found when searching for 'Italy' (or 'Italy' written in another language). Geo coordinates can be used for instance as shown in the previous section: '*Show nearby content*'

In similar ways, the temporal metadata enrichments (time labels, broader time labels and temporal endpoints) can increase the possibilities of locating Europeana objects.

The update of the widget will utilize Europeana's improved searching capabilities, which will hopefully improve the relevance of the objects shown to the ECLAP user.

Filter ECLAP results

As ECLAP also has a collection on Europeana, it is possible that items from it are shown in the widget. To avoid this, as having external duplicate ECLAP objects on the ECLAP portal would be redundant, the widget will be updated to filter out results from the ECLAP collection.

8 Bibliography

- [1] Drupal Vocabularies - <http://drupal.org/node/22272>

- [2] HTML5 specification - <http://dev.w3.org/html5/spec/Overview.html>

- [3] AnnoCultor - <http://annocultor.eu/>

- [4] Dublin Core - <http://dublincore.org/>

- [5] GeoNames: geographical database - <http://www.geonames.org/>

- [6] AnnoCultor Time Ontology - <http://annocultor.eu/time/>