EUROPEAN COLLECTED LIBRARY OF ARTISTIC PERFORMANCE

www.ECLAP.eu

Grant Agreement No 250481

DE3.2 Accessibility and Multilingual Support for ECLAP Solution

Version: v1.1

Date: 11/07/2011

Project Title: ECLAP
Project Number: ICT-PSP-250481
Deliverable Number: DE3.2
Accessibility: public
Work-Package contributing to the Deliverable: WP3, WP3.3
Nature of the Deliverable: document
Status: draft

Contractual Date of Delivery: 30/6/2011
Approve for quality control by: Paolo Nesi
Finally approved by coordinator: 11/07/2011
Actual Date of Delivery: 12/07/2011

Document responsible: Natasa Sofou
Email address: natasa@image.ntua.gr
Affiliation acronym: NTUA

Authors:
• Natasa Sofou, Theofilos Malis (NTUA)
• Jaap Blom (B&G)
• Michela Paolucci, Ivan Bruno, Daniele Cenni, Pierfrancesco Bellini, Paolo Nesi (DSI)

Revision History:

Revision	Date	Author	Organization	Description
v0	16/05/2011	Natasa Sofou	NTUA	ToC
v0.2	21/6/2011	N. Sofou, J. Bloom, M. Paolucci, I. Bruno, P. Bellini	NTUA, BNG, DSI	Contributions
v1.0	6/7/2011	N. Sofou, J. Blom, P.Nesi, P. Bellini	NTUA, BNG, DSI	Contributions, Modifications
V1.1	11/7/2011	P.Nesi	DSI	Control and approval

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This deliverable contains original unpublished work except where clearly indicated otherwise.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	DE3.2 Accessibility and Multilingual Support for ECLAP Solution, M12
Identifier.de	DE3.2
Identifier.ISBN	
Creators	Natasa Sofou, Jaap Blom, Michela Paolucci, Ivan Bruno, Daniele Cenni, Pierfrancesco Bellini, Paolo Nesi
Subject	Accessibility, multilingual, eclap portal,
Description	Deliverable on Accessibility and Multilingual Support for ECLAP Solution. The document describes the configuration and decision taken to to cope with multilingual aspects, accesibility etc and their integration to ECLAP Social Service Portal and ECLAP automated back office to translate directly queries and web pages.
Keywords	Accessibility, multilingual, eclap portal,
Publisher	ECLAP
Date	11/07/2011
Format	Document
Type	PDF or 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:

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 Executive Summary and Report Scope

Scope

The aim of the ECLAP project is to create a considerable online archive and portal for all the performing arts in Europe, which will also become searchable in Europeana. ECLAP is creating a best practice network, and is going to develop best practise guidelines covering key areas of making digitised performing arts accessible, such as metadata and content modelling, content enrichment, IPR issues and management, and tools for education and leisure use. 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 aims at describing the configuration and decision taken to cope with multilingual aspects, accessibility, etc., and their integration with ECLAP Social Service Portal and ECLAP automated back office to translate directly queries and web pages. Associated with the report is the delivery of the integration with ECLAP automated back office and ECLAP Social Service Portal. This document will be also produced in a publishable manner, as a guideline for public access and spreading out.

Related Documents

- DE1.1 "Assessment and Evaluation Manual" which provides guidelines for the self control of the performed work, reference values for the metrics and methods for their estimation. The manual also includes guidelines on self assessment and for external peer reviewers.

- DE2.1.1 "User requirements and use cases" which provides the methodology on how to gather user requirements, as well as how to perform evaluation. It also contains description of the user requirements and the corresponding use cases and test cases in UML.

- DE3.1 "Infrastructure: ingestion and processing content and metadata". This document is divided in two parts. (i) A report describing the model for metadata and content, procedure for content enrichment, annotation, contextualization, etc. The report also describes the content format which can be accessed and ingested with the usage of the AXCP Ingestion tool and its configuration scripts. (ii) A report describing the functionalities of the ECLAP automated back office and ECLAP Social Service Portal and their installation parameters/settings, user manual, scripts, activities, configuration settings, procedures for creating users, groups, updating taxonomies, etc.

- D4.1 "Metadata Descriptors, Identification and Definition". This document includes the identified metadata and their corresponding semantics in relationship with those of Europeana, taking into account all kinds of metadata coming from different content providers that participate in the project.

- D4.2 "Content and Metadata, Selection, Aggregation and Augmentation". This document includes the interoperability among metadata and classifications, taking into account the Europeanan standard, the selection criteria for content and for their aggregation and contextualization, as well as the description of the solution used for content augmentation and enrichment.

- DE6.1.1 "Early validation and service optimisation" This report includes: Early validation and service optimisation, procedures and results regarding the validation of the performed activities.

FROM THE DOW: regarding DE3.2

A report describing the configuration and decision taken to cope with multilingual aspects, accessibility, etc., and their integration with ECLAP Social Service Portal and ECLAP automated back office to translate directly queries and web pages. Associated with the report is the delivery of the integration with ECLAP automated back office and ECLAP Social Service Portal. This document will be also produced in a publishable manner, as a guideline for public access and spreading out.

Moreover:

DE3.1 – Infrastructure: ingestion and processing content and metadata, M7

6   Content and metadata ingestion and management

2   Introduction

The aim of the ECLAP project is to create a considerable online archive and portal for all the performing arts in Europe, which will also become searchable in Europeana. ECLAP is creating a best practice network, and is going to develop best practise guidelines covering key areas of making digitised performing arts accessible, such as metadata and content modelling, content enrichment, IPR issues and management, and tools for education and leisure use. This will result in cultural enrichment and promotion of European culture, and in improvements in learning and research in the field of performing arts.

WP3 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. The approach is based on a distributed architecture which provides local ingestion tools optionally installed in the factory of the archives in the network. The ECLAP architecture supposes that not all the archive may be entitled to use some local tools for content ingestion and posting on ECLAP. In other cases, the ECLAP processing tools will have to be capable to get into the remote archives to harvest their metadata and descriptors and to really get the files.

WP3.3 focuses on Accessibility and Multilingual support for ECLAP solution. Specifically, information or objects in ECLAP need to be searchable and presentable independent of the language. Therefore, multilingual interoperability is a key task in the development of the ECLAP infrastructure. This task is particularly concerned with enabling ECLAP users to navigate and find relevant content not described in their native or preferred language.

This is also a requirement from Europeana. Content in Europeana is described and accessible through its metadata descriptions (or surrogate descriptions), but appear in their original language. The objective of this task is to ensure truly multilingual interoperability for all features within ECLAP and with Europeana. This task will implement solutions to cope with multilingual access issues for users and objects alike within ECLAP. In order to provide multilingual access capabilities for ECLAP, the following services are proposed:

- Multilingual thesauri SKOSification as vocabulary/taxonomy. An automatic service will enable the production of the multilingual vocabulary/taxonomy.

- Multilingual mapping tool for the alignment of controlled vocabulary/taxonomy. This service will be used to automatically align the SKOS.

- Automatic translation tools for GUI and vocabulary/taxonomy. A translation tool will produce the necessary multilingual representations of the Users' queries. Possible translation tools are for example: Worldlingo or Google Translation tools.

- Coverage of 12/13 major languages for metadata: Danish, Polish, Slovenian, Greek, English, Italian, French, Dutch, Spanish, Hungarian, German, Portuguese, Catalan.

DE3.2 that is mainly a report describing the configuration and decision taken to cope with multilingual aspects, accessibility, etc., and their integration with ECLAP Social Service Portal and ECLAP automated back office to translate directly queries and web pages. Associated with the report is the delivery of the integration with ECLAP automated back office and ECLAP Social Service Portal.

The remaining of this document has the following structure: in section 3 we provide a description of multilingual metadata automated ingestion focusing on services for multilingual access such as automated metadata translation and thesauri SKOSification. Section 4 refers to multilingual metadata editing and validation describing metadata panel, metadata editor enrichment and validation mode. In section 5

multilingual metadata indexing and query is reported. Details about the elements that have to be indexed, metadata indexing, multilingual indexing and search and faceted search are provided. Section 6 refers to multilingual portal interface and automated translation, while section 7 deals with multilingual portal interface validation and acceptance process. Multilingual SKOS automated translation is addressed in section 8, and multilingual taxonomy editor is described in detail in section 9. Section 10 refers to multilingual metadata and GUI on mobile devises. In section 11 accessibility analysis issues are addressed. More precisely, there is an overview of available W3C accessibility assessment methodologies and tools, concluding to strategies for assessing the accessibility of ECLAP portals: metadata ingestion servers, social service portal and mobile portal. Finally, in section 12 there is reference to automatic translation tools.

3 Multilingual Metadata Automated Ingestion

3.1 Metadata Ingestion Overview

As described in DE3.1 (section 6) and DE4.1 the basic workflow for massive content and metadata ingestion within ECLAP consists of the following steps:

1. Content partners provide metadata in their native or desired language using the ECLAP Metadata Ingestion Service (EMIS) portal, metadata should be provided as XML, it is directly uploaded as a file or harvested them via an OAI-PMH access.

2. Each content partners maps its own metadata XML structure to the ECLAP metadata XML format , this will is done using the EMIS portal to define a XSLT that is used in the mapping phase;

3. In case of a OAI-PMH access the EMIS will crawl the content partners archives and it acquires the original metadata;

4. When the original metadata is acquired it is mapped to the ECLAP metadata format and stored and it is made available toward the ECLAP Social Service Portal (ESSP) via an OAI-PMH server;

5. The ESSP regularly crawls the EMIS to acquire metadata represented using the ECLAP metadata format and in the original metadata format

6. For each metadata record acquired the ESSP will download the content files associated with it, ESP will use the content urls defined in the ECLAP metadata (content may be also acquired using Hard Disks, DVDs in case content size is too high for a rapid download)

7. Each piece of content acquired will be formatted, adapted and a complete media content will be produced in the ESSP portal with the associated metadata;

8. Each produced content will be available in the ESSP portal for:

   - metadata enrichment

   - metadata translation

   - IPR definition

   these activities are orchestrated by using an internal workflow management tool integrated in the portal.

9. When the content and the associated metadata are ready they will be made available (on the basis of the IPR defined) for access to the final users accessing on the ESSP portal.

10. Metadata will be published on Europeana and Europeana users will reach the ESSP portal to see the content.

In the Cultural Content Metadata Space, the largest technological challenge is to ensure syntactic and semantic interoperability across the different types of metadata that exist in the Cultural Heritage sector. The technical standards enabling interoperability form an important dimension of this work. In order to achieve semantic interoperability within ECLAP we need a common automatic interpretation of the meaning of the exchanged information, i.e. the ability to automatically process the information in a machine-understandable manner. The first step of achieving a certain level of common understanding is a representation language that

exchanges the formal semantics of the information. Then, systems that understand these semantics can process the information and provide web services like searching, retrieval etc.

The following figure illustrates the proposed workflow for ingesting metadata in ECLAP.
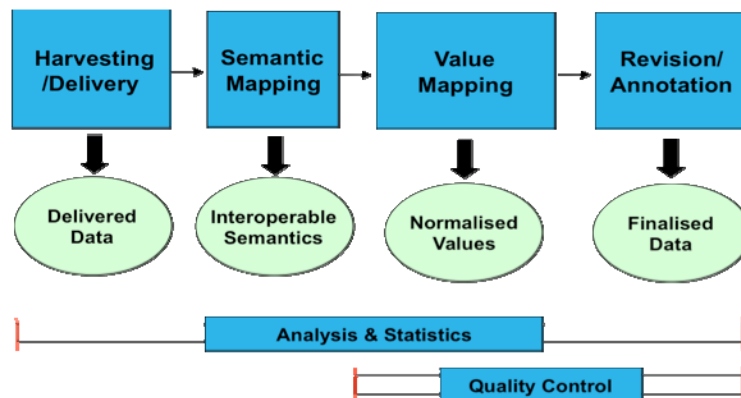
Figure 3-1 *Ingestion workflow*

The workflow consists of four phases. Each phase is responsible for specific services all needed to ensure the quality of the ingestion process.

Harvesting/delivery is responsible for collecting the metadata. Metadata can be in different languages. There is an interface for different methods of data delivery including, OAI-PMH, HTTP upload/download FTP upload/download.

Semantic Mapping will provide the service for assigning semantics to the harvested metadata. It will assist to manually map Providers fields to a reference rich schema. Providers that have metadata in supported known formats might be able to omit this step (use stored transformations from selected schemas to the reference schema based on existing crosswalks).

Value Mapping will take existing attribute values and produce different/edited values. In particular:

- It will enable providers to resolve data issues, e.g. map own terminology list to selected terminology lists
- It will then automatically normalize data e.g. dates, geographical locations, nationality/language, name writing convention to selected vocabulary standards.

Revision/Annotation will enable the addition of data that is not in the original metadata (e.g empty fields, fields that take values from controlled vocabularies).

Analysis & Statistics service will provide detailed analysis and statistics of metadata contributed by a provider. (i.e. number of items imported, total values per field etc).

Quality Control will automatically check and report on Content Provider's data (i.e. missing values, malformed data). Error reports and warnings will be produced to facilitate editing the semantic mappings, value mappings and/or edit items until the Provider's data successfully passes the Quality control checks.

For information about the ingestion and mapping process the reader can refer to DE3.1 (section 6.1) and DE4.1 (section 3).

3.2   Services for Multilingual Access

This section describes the services that can be employed when aiming at multilingual access. Basically we provide a general description of SKOS technology, the main features, and guidelines from technical point of view for proceeding to SKOSification of the terminology of an organization, thus enabling multilingual access.

3.2.1   Multilingual Thesauri SKOSification

3.2.1.1   *Simple Knowledge Organization System - SKOS*

The Simple Knowledge Organization System (SKOS) is the result of several years of work in the field of the Semantic Web. SKOS is a common data model for sharing and linking knowledge organization systems via the Web. It was designed within the Semantic Web Advances Development for Europe Project (SWAD-Europe) working group and it has been acknowledged as a W3C recommendation in August 2009.

SKOS is a concept-oriented data model as the fundamental element of any terminology designed in SKOS is the concept and not the term that expresses this concept. It consists of a basic structure that can be extended by specific classes for detailing lexical parts or semantic relations between the concepts of the terminology. SKOS data model is comprised of three main components: classes, properties and relations. They always start with the prefix "skos:" If the element is a class then the "skos:" suffix starts with an upper-case character and if it is a property then it starts with a lower case character.

SKOS data are expressed as RDF triples. The concepts may be subject or object and related via a SKOS property which would be the predicate. As RDF triples, SKOS concepts can be identified using URIs. Although the SKOS data model does not require the use of persistent identifiers, the exploitation of SKOS concepts in Linked Opened Data necessitates their use.

3.2.1.2   *Main Features*

From a terminology point of view a concept can be defined as an idea, notion or unit of thought. A concept in SKOS is introduced as a class *skos:Concept.* The skos:Concept class is an instance of *owl:class*, which is a class from the OWL data model. In this way, connections between the two data models are enabled. SKOS concepts can be brought together into two classes: SKOS concept scheme (*skos:ConceptScheme*) and SKOS collections (*skos:Collection).* Each concept must be identified in a unique way in order to avoid any ambiguity. The identifiers are introduced by a specific RDF property *rdf:resource.*

There is a distinction between the concept itself and the terms that may used to express this concept. The SKOS data model enables the expression of terms referring to a concept via lexical labels. Three types of lexical label are defined: preferred label (*skos:prefLabel*: corresponds to the notion of descriptor from the standards for the elaboration of thesauri), alternative label (*skos:altLabel*: is used to give synonyms or other ways to refer to preferred label) and hidden label (*skos: hiddenLabel*, may be used for mentioning the misspellings of preferred or alternative labels but also for mentioning obsolete forms of a term).

Further information about a SKOS concept can be provided by the property *skos:note.* Notations are symbols or codes that are not recognizable or understandable in any natural language. The *skos:notation* can then be used for example in the case of classifications where a code refers to a term referring itself to a concept. The SKOS model offers a variety of possibilities to provide information related to concepts. The different types of notes that can be used to document a concept are: note (*skos:note*), change note (*skos:changeNote*), definition (*skos:definition*), editorial note (*skos:editorialNote*), example (*skos:example*), history note (*skos:historyNote*) and scope note (*skos:scopeNote*).

SKOS model enables the specification of semantic relations between different concepts. Hierarchical relations are introduced via the properties *skos:broader* and *skos:narrower*. The *skos:broader* property is used to assert that a concept has more general meaning, whereas the *skos:narrower* to assert that it has a more specific meaning. In order to enable non-immediate link between two concepts, the two transitive properties *skos:broader-Transitive* and *skos:narrowerTransitive* are provided. If two concepts are not equivalent nor connected via any hierarchical relation, but are related to each other, then the associative link between them is expressed through the property *skos:related.*

The SKOS data model provides five mapping properties for making alignment between concepts from different concept schemes. The *skos:closeMatch* and *skos:exactMatch* properties are used to make a mapping link between concepts that are very similar or equal. The properties *skos:broadMatch,*

skos:narrowMatch are used for a hierarchical mapping link between concepts. Finally, the property *skos:relatedMatch* is used to associate two concepts.

The term SKOSification implies the process of conversion or transformation of a terminology into SKOS. In this paragraph we will illustrate the guidelines as suggested in detail in deliverable D.4.2 and the tools that at this time are provided for the SKOSification process.

3.2.1.3 Guidelines for SKOSification

Below we describe in brief the guidelines for proceeding to this conversion from a technical and organisation point of view.

1. **Evaluate the main features of the terminology to be migrated:** firstly, the institution must have defined the purpose of its terminology (e.g. indexing and retrieval, only indexing, or only retrieval). Then, the institution must evaluate if SKOS is the appropriate format considering the content of its terminology

2. **Identify your concepts:** W3C proposes two main steps to proceed to the identification of concepts. At first, to create (or reuse) a Uniform Resource Identifier (URI) to uniquely identify the concept, and then to assert in RDF, using the *rdf:type* property, that the resource identified by this URI is of type *skos:Concept*

   Several standards have been developed in order to normalise the definition of URIs: PURL (Persistent Uniform Resource Locators), URN (Uniform Resource Name), NBN (National Bibliography Numbers, Archival Resource Key (ARK), OpenURL, Digital Object Identifier (DOI).

   i. **Use of a Persistent Identifying System for the definition of the URIs:** as the identification of concepts is achieved with the definition of HTTP URIs, these URI must be declared to persistent identification systems such as PURL which is normalised. This will also be of a great benefit since it is location-independent, e.g. if the terminology is moved from one housing server to another, the URIs identifying the concepts of this terminology will not have to be modified.

   ii. **Use of non-explicit URIs:** in order to avoid the reuse of a same URI for identifying two different concepts.

3. **Define with precision the labels expressing concepts:**

   i. **Preferred labels must be unique within a concept scheme:** any pair of concepts from a same concept scheme should have the same preferred label in a given language. The SKOS data model does not forbid that one concept can have two same preferred labels in two different languages

   ii. **Each concept must be expressed with one preferred label per language (mandatory):** labels are meant to help the understanding the meaning of a concept. Besides their contribution in a multilingual context, they are also helpful for purposes of administration and maintenance

   iii. **Avoid the concatenation of several words for a same label:** for example, double concepts such as "dwelling/houses" must be considered as two different concepts that are linked by a semantic relation. We can state that "dwelling" and "houses" are synonyms; then the double concepts can be modelled as: dwelling: preferred label and houses: alternative label

   iv. **Privilege the use of the lemma for the preferred label and possibly the other labels:** any artificial word or code to label a concept must be defined using the *skos:notation* property. In the case of compound terms the addition of adjectives or verbs to a noun phrase should be limited. The use of articles and prepositions should be avoided in order limit the length of the label

   v. **Privilege the typography in use by convention in the languages involved:** the labels should respect the typographical rules that are usually in use in the languages of the labels. The conventions involved in the use of each language should be respected.

4. **Avoid the duplication of information:** as some of the properties available in SKOS model are proposed as pairs (inverse or symmetric), this supposes that the use of one property implies the opposite or the reverse. The less redundant information there is, the faster the results of a query can be retrieved.

5. **Provide precision to the semantic relations of your concepts:**

    i.   Non-immediate hierarchical relation: for instance, the pair of properties *skos:broaderTransitive* and *skos:narrowerTransitive* describes with precision the relation between concepts when two levels of hierarchy are impacted

    ii.   Consistency of the semantic relations: to ensure consistency, mixing hierarchical relationships with associative ones should be avoided.

6. Enable the multilingualism:

    i.   Provide for each concept an equivalent label in the languages involved in your terminology: special attention must be paid to the multilingual labels expressing the concepts

    ii.   Use the same system of language tags for defining the language of label: in the case where a specific language tags system is not required, the use of the language systems defined in ISO 639-1 is recommended. In ISO 639-1 the language tags are coded on two letters in lower case.

7. Ensure the documentation of concepts and the terminology:

    i.   Provide documentation for each change that may occur to a concept and its labels: for the purposes of administration and maintenance of the terminology, each change must be reported in the SKOSified terminology using change notes (*skos:changeNote*) or editorial notes (*skos:editorialNote*)

    ii.   Provide as much as possible documentation to concepts with scope notes: The use of scope notes (*skos:scopeNote*) can be very helpful in enabling a better understanding of the concepts with contextual information. Examples may also be provided via *skos:example* property.

From a technical point of view, in order to check the consistency of the converted terminology to the SKOS model the free online web service Pool Party is recommended. Pool Party offers a tool for validating SKOS files that may be already online or stored on your local repositories.

This tool checks the consistency of the SKOSified terminology with respect to: valid URIs, missing language tags, missing labels, loose concepts, disjoint OWL classes (w.r.t consistency), consistent use of labels, consistent usage of mapping properties, consistent usage f semantic relations. All these checking points are referred by the guidelines of SKOSification.

From the content point of view, only the administrators and users of the terminology can validate the final migration of the terminology into SKOS format. At least for an initial transformation process, they will be the one able to confirm or modify the general design of the terminology and its semantic relations according to the indexing and retrieval efficiency.

3.2.1.4   Available Tools

Athena project (http://www.athenaeurope.org/) has proceeded to identify, review and evaluate a set of tools that can assist in one or more steps of the procedure of SKOSification. Available tools are evaluated and presented at http://www.athenaeurope.org/athenawiki/index.php/Benchmark having been classified with respect to their functionalities, more specifically:

- registration
- skosification
- search
- navigation
- mapping
- enrichment
- moderation
- collaboration
- graphical interface
- navigation map
- file checking

- format recognition
- syntax checking
- search engine
- folksonomy
- classification
- alignment recommendation engine
- annotation
- discussion
- UGC management
- account management
- versioning management
- social web service
- dependency management
- methodology

There is naturally no generic tool that can convert any input file into SKOS, as this process has to be managed in a very technical way and knowledge and technical skills of a computer engineer may be required. A list of the existing tools that can contribute in the SKOSification process is the following:

- ThManager: is an editing tool for SKOS thesauri which allows to registering in an internal database several thesauri.
- SKOSed: is a plug-in for the Protege software, which is an ontology management tool.
- Annocultor: is a set of command-line tools which allow to SKOSify a large number of terminology resources.
- xTree: is a web service for editing and enriching a SKOS terminology.
- Ingester: presents interesting and useful features for terminology management.
- ASKOSI: among others its establishes what works when trying to integrate different SKOS aware software.
- XL2XML: has conversion rule sets to generate various SKOS/RDF/XML files from an Excel spreadsheet for Vocabulary management.
- PoolParty: offers a free SKOS Thesaurus Consistency Checker service that lets you upload a thesaurus file in SKOS format and checks if the data is consistent.
- OSIM SKOS manager produced by DISIT. Which can be used to ingest textual resources and covert them in keywords and concepts that can be organized in a multilingual SKOS.
- internal tool of DRUPAL to cope with taxonomies and SKOS also adopted in ECLAP.

3.2.2 Automatic Metadata Translation

The automatic metadata translation has to cope with the multi-language version of metadata. The translation is performed by the AXMEDIS AXCP Grid and is implemented as an AXCP rule. The translation is made by using translator tools (i.e. Google, Bings).

According to the Eclap workflow, the translation of content metadata is a workflow activity. The translation of metadata can be done if the workflow status is put to UNDER-AXCP value to get the lock on the content and perform changes on metadata. In event of failure when taking the lock, the translation is delayed by resubmitting a call to the AXCP scheduler to run again after a time interval. If the lock acquisition can not be performed after a number of retry, the process will be aborted and a notification mail is sent to the administrator.

The list of languages to use for translating metadata is defined as configuration parameter in the configuration options. To speed up the transaltion, when a metadata is already available in the language under translation, the translator uses the available term.

3.2.2.1 Automatic translation

The main activities are in the order:

1. **Workflow lock request**: a workflow request is sent to the Drupal workflow manager to ask for a status transition from the UPLOADED to the UNDER-AXCP status (according to workflow state diagram).

   a. If the transition is not performed and the retryCount is less than a threshold, the retryCount is incremented and the Automatic translation resubmitted to the AXMEDIS Grid by scheduling it to a new time schedule and passing the new retryCount value. The current activity is immediately stopped.

   b. If the retryCount is greater than the threshold, the current activity is immediately stopped and the automatic translation aborted definitively. A notification mail is sent to the administrator.

2. **Metadata retrieval**: 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 rootObjectInfo table of publishing AXMEDIS database

4. **Metadata translation**: each metadata is translated for each default languages. If metadata are already available in the language under translation, the process skips them.

5. **New Metadata insertion**: translated metadata are inserted in the publishing DCMI AXMEDIS database

6. **Workflow Tables Update**: The AXCP Rule updates the following:

   - **md_number** (*workflow_info* table): the new total number of metadata

   The AXCP Rule inserts a row for each new translated metadata and sets the following fields of the *workflow_metadata* table:

   - **Axoid**: the id of AXMEDIS mpeg-21 object
   - **Id**: the id of 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 updates the rootObjectInfo 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 sent 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**: retryCount is increased and the resubmission is done in the event of workflow transition failure or connection failure.

4 Multilingual Metadata Editing and Validation

Metadata editor is the tool for enriching and validating metadata. The enrichment activity includes metadata translation and it is performed by MetadataEnricher users as defined in the workflow.

Metadata translation validation is done by the MetadataValidator users as defined in the workflow. Each MetadataValidator can validate the metadata of a specific language using the metadata editor.

Metadata editor can be activated by the user:
- o   clicking on a direct link according to requests received via notification
- o   spontaneously by using |Edit Object| and then |Edit Metadata|
- o   when needed for validation by using |Edit Object| and then |Validate Metadata|

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 MetadataEnricher role and in Validation mode for WF Validation role. Users without a workflow role access the editor only in read-only mode.

4.1.1 Metadata Panel

The panel is the metadata edit area and is organized in tabs so as to organize metadata seta.

Possible metadata sets will be:

- **DCMI**: Dublin core set

- **Taxonomy**: classification terms

- **Groups**: List of Groups associated with content and/or Public flag

- **Properties**: Technical set of metadata accessible only by administrator.

Each tab shows the list of metadata and iconized controls. To help the user each control has a tip with the name of action.

The DCMI panel is divided into two sub-areas: *Reference and Changes* (see Figure 4-1).
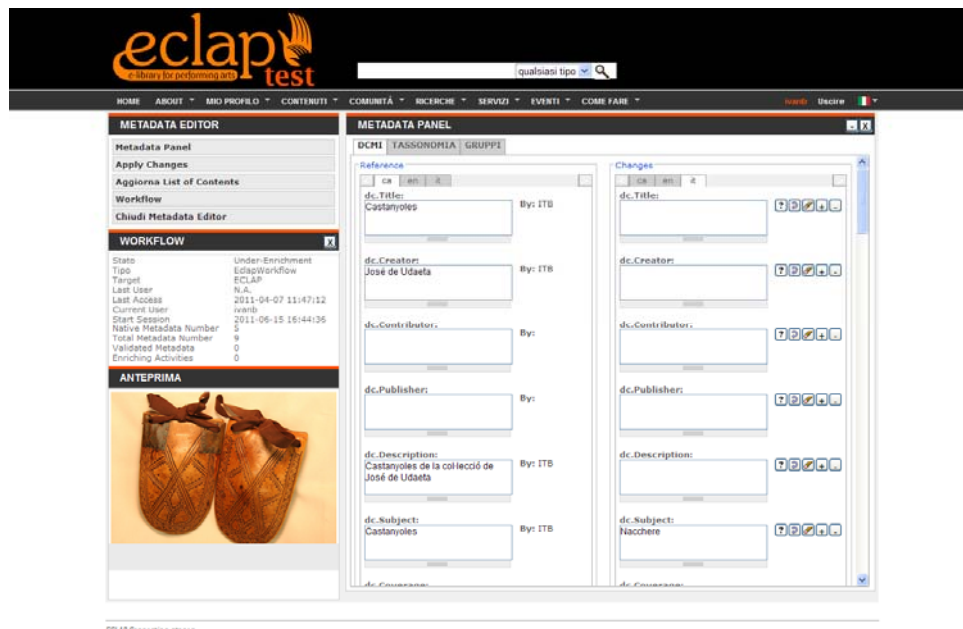

Figure 4-1 Metadata Editor main view

The Reference fields set shows the current metadata values in read-only text box with a label reporting the metadata field name and the username of the user that made the last change/edit (By) or/and validated the metadata displayed close to the metadata text box:

Figure 4-2 Reference text box

The Changes field set shows a copy of reference set where the user can edit/adding values. Metadata are displayed in a text box and a label reports the metadata field name. In the changes set each metadata has also a set of iconized buttons.

When a user work as MetadataEnricher:

- **Help**: It opens a popup dialog displaying an help about the kind of metadata
- **Undo**: Reset the value to the fist time displayed string
- **Clear**: it cleans the text box
- **Add more**: it adds a new text box with own control buttons.
- **Remove**: it removes the text box/metadata value

Figure 4-3 Changes text box

When user work as validator:

- **Help**: It opens a popup dialog displaying an help about the kind of metadata
- **Undo**: Reset the value to the fist time displayed string
- **Clear**: it cleans the text box
- **Validate/Invalidate**: it validates/invalidates the metadata

Figure 4-4 Validated text box

Both Reference and Changes columns have a language tab that allows browsing on available languages and this enables users to compare a language with another or to help the user in translating the metadata.

Figure 4-5 Language tabs

The "language tabs" is based on languages selected by users according to their translation and understanding capability. Thus, a user with two languages can work only on metadata related to them. The "add" and "remove" buttons are associated with multiple metadata. The "validate/invalidate" buttons are displayed/enable only for administrator and user with WF MetadataValidator role.

During the enrichment, every time a user makes change or edits new metadata the text box/selector background colour becomes red.

Figure 4-6 Modified changes box

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

Figure 4-7 Validated metadata

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

4.1.2   Metadata Editor Enrichment mode

The editor works in Enrichment mode for the WF MetadataEnricher role. The editor can be used to edit/enrich metadata when the workflow transition Uploaded → Under-Enrichment is done with success. This means no other user is working on the content, the workflow tables are updated with the user id and start session date and time, otherwise the content is locked by other user and the editor is opened in read-only mode just for browsing metadata. When a 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.

Figure 4-8 Workflow info block (enrichment)

4.1.3   Metadata Editor Validation mode

The editor works in Validation mode for the WF Validation role. The editor can used to validate/invalidate metadata when the workflow transition Uploaded → Under-Validation 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 is 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 MetadataValidator could "validate/unvalidate" single metadata or "validate/unvalidate" all metadata.
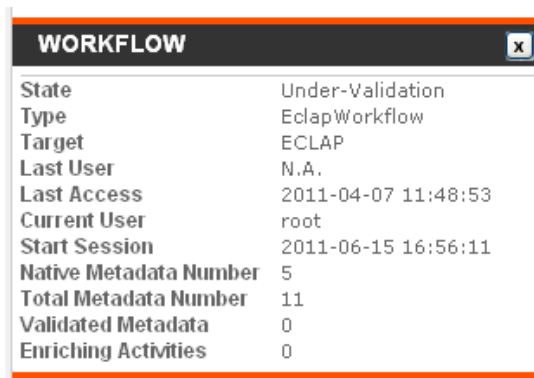
Figure 4-9 Workflow info block (validation)

4.1.4 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 the changes made before closing.
- End Session has to be clicked to close and return back to Edit object page
  - If changes are pending a popup dialog box will appear to remind him apply the made changes before exit or discard them.
- If users try to change page or exit, a popup dialog reminds them to apply the made changes or discard them.
- If users leave the page without clicking End Session, the page before unloaded calls the server to end the session.
- If an object is in a pending UNDER-xxx status, only the administrator or the user coming back on the content can complete the back-transition.

5 Multilingual Metadata Indexing and Query

This section reports how the metadata, content, comments, and web pages indexing can be done and how it is accessed. Indexing is based on a *Solr* service [solr] installed on a Tomcat servlet container.

5.1 Elements to be indexed

	Language	Reindex	Add to index	Update on index	Delete from index
Media Object	Their own defined into RootObjectInfo table as deflanguage or into DCMI.language	via reindex rule	via upload rule	via update rule, may be not useful. Also when the generation of FileSecco is done, the rule update the Lucene index since the flags into RootObjectInfo are changed as well	Via delete rule called from XMF admin
				via drupal, MetadataEnricher. The changes are done into the axDB and into the Lucene Index. This happen from XMF admin and from MetadataEnricher.	
Web pages	Their own language	via reindex	via drupal when	via drupal when modified	via drupal when deleted

		rule	created		
Forum topic	The language of the group	Via reindex rule	via drupal when created	via drupal when modified	via drupal when deleted
Comments	To a web page: The page language To a content: content language In a forum: group language	via reindex rule	via drupal when created	via drupal when modified	via drupal when deleted

5.2 Metadata indexing

The following table reports the metadata set used for indexing:

Name[1]	Note[2]	Present in Play window[3]	Multiple instances[4]	Language[5]	Frontal Search & Advanced Multilanguage Search[6]	Advanced content search Supported Metadata[7]
Contributor	DCMI term	Yes	Yes	Yes	No	Yes
Coverage	DCMI term	Yes	Yes	Null	No	Yes
Creator	DCMI term	Yes	Yes	Yes	No	Yes
Date (DC.date)	DCMI term Last update of metadata	Yes	No	Null	No	Yes
Date_created	DCMI term	No	No	No	No	Yes
Date_dateaccepted	DCMI term	No	No	No	No	Yes
Relation_ispartof	DCMI term	No	No	No	No	Yes
Relation_isreferencedby	DCMI term	No	No	No	No	Yes
Relation_isreplacedby	DCMI	No	No	No	No	Yes

---

[1] Name of the metadata term.

[2] Can be: DCMI if it is a Dublin Core term; ECLAP if it is an internal term managed by the ECLAP BPNET.

[3] If the term is shown during the play of the resource.

[4] The term can be present in multiple instances.

[5] Term available in different languages.

[6] If YES the metadata is used in the full text search and in Advanced Search in the "Advanced Multilanguage Search of Objects" field.

[7] If YES the metadata is used in the advanced search (objects, pages, forums, comments).

Relation_isrequiredby	DCMI	No	No	No	No	Yes
Description	DCMI term	Yes	Yes	Yes	Yes	Yes
Format	DCMI term	Yes	NO, not only our format but also other kind such as tape, dvd, etc.	No	Yes	Yes
Format_extent	DCMI term	Yes	No	No	No	Yes
Workflow_type	ECLAP term	Yes	No	No	No	Yes
Identifier	DCMI term	Yes	Yes	No	No	Yes
Language	DCMI term	Yes	No	No	No	Yes
Relation	DCMI term	Yes	Yes, it should be a link to the source files	No	No	Yes
Rights	DCMI term	Yes	Yes	Yes, it may be a general statement	No	Yes
Source	DCMI term	Yes	Yes	Yes	No	Yes
Subject	DCMI term	Yes	Yes	Yes	Yes	Yes
Title	DCMI term	Yes	Yes	Yes	Yes	Yes
Type	DCMI term	Yes	No	Null	Yes	Yes
Group	ECLAP term	Yes	Yes	Yes	No	Yes
Taxonomy	ECLAP term	Yes	Yes	Yes	Yes	Yes
N. accesses (also called httpdownload)	ECLAP term (the so called httpdownload)	Yes	No	No	No	Yes
Date (different from DC.Date)	ECLAP term Last update of file	Yes	No	No	No	Yes
Duration	ECLAP term Duration time for audio and video	Yes	No	No	No	Yes
Partner (Publisher on the RootObjectInf	ECLAP term Affiliation among those in the user	Yes	No	No	Yes	Yes

o)	registration					
User (uid link)	ECLAP term: ECLAP portal nickname on the portal	Yes	No	No	No	Yes
Abstract	DCMI term	No	Yes	Yes	No	Yes
Publisher	DCMI term	No	Yes	Yes	No	Yes
axoid	ECLAP term, AXOID of the content	No	No	No	No	Yes
axoidref	ECLAP term, not used	No	No	No	No	Yes
cluster	ECLAP term	No	No	No	No	Yes
creationDate	ECLAP term	No	No	No	No	Yes
deflanguage	ECLAP term, coding of language of the digital resource	No	No	No	No	Yes
dx	ECLAP term, dimension on X for images and video	No	No	No	No	Yes
dy	ECLAP term, dimension on Y for images and video	No	No	No	No	Yes
filesecco	ECLAP term, coding of file type and extension	No	No	No	No	Yes
governedObject	ECLAP term, DRM protected or not	No	No	No	No	Yes
iphone	ECLAP term, 1 for yes, 0 for no (content valid for iPhone)	No	No	No	No	Yes
lastModificationDate	ECLAP term	No	No	No	No	Yes
latitude	ECLAP term	No	No	No	No	Yes
longitude	ECLAP term	No	No	No	No	Yes

mobile	ECLAP term, 1 for yes, 0 for no	No	No	No	No	Yes
modified	ECLAP term	No	No	No	No	Yes
nid	ECLAP term, internal NODE ID	No	No	No	No	Yes
nohttp	ECLAP term, deprecated	No	No	No	No	Yes
nop2p	ECLAP term, deprecated	No	No	No	No	Yes
nvoti	ECLAP term, deprecated	No	No	No	No	Yes
p2pdownload	ECLAP term	No	No	No	No	Yes
pcview	ECLAP term, 1 for yes, 0 for no	No	No	No	No	Yes
Pda	ECLAP term, 1 for yes, 0 for no (content valid for PDA)	No	No	No	No	Yes
playnumber	ECLAP term, deprecated	No	No	No	No	Yes
progressivepc	ECLAP term, deprecated	No	No	No	No	Yes
progressivertsp	ECLAP term, deprecated	No	No	No	No	Yes
published	ECLAP term, 1 for yes, 0 for no	No	No	No	No	Yes
ranking	ECLAP term, a copy of the vote	No	No	No	No	Yes
resolution	ECLAP term	No	No	No	No	Yes
sourcefile	ECLAP term, internal use only	No	No	No	No	Yes
sourceurl	ECLAP term, internal use only	No	No	No	No	Yes
stb	ECLAP term, 1 for yes, 0 for no	No	No	No	No	Yes
xmlObjectUri	ECLAP term,	No	No	No	No	Yes

	internal use only					
Drupal contents	Title, description, body of Drupal pages, forums, comments	No	Yes	Yes	No	Yes
Rich text documents (doc, docx, ppt, pptx, xls, xlsx, htm, html, pdf)	Documents attached to Drupal contents	No	Yes	Yes	No	Yes

5.3 Multilingual indexing & search

This section reports how metadata and texts in different languages are indexed using Solr and how the searches are performed in the frontal and in the advanced search.

The main features and functionalities of the indexing service are the following:

- Each Drupal content (page, forum comment) and Axmedis content is indexed in the Solr index;
- Rich text documents attached to the Axmedis object are parsed and indexed in the Solr index;
- The administrator is able to rebuild automatically from scratch the Solr index in case of index corruption;
- Digital contents are indexed in a single Multilanguage index;

The main features and functionalities of the searching service are the following:

- Each Drupal content (page, forum comment) and Axmedis content is searchable in the Solr index;
- The search service is divided in simple and advanced search;
- Fuzzy search and wildcards are possible both for the simple and the advanced search;
- Boosting of terms is possible for the administrator;
- Faceted search is allowed both from the simple and the advanced search;
- Search result are presented in relevance descending order (scoring) with pagination;

Metadata to be indexed are translated into different languages by a separate translation service, based on a Javascript rule. The boost and weighting of different query aspects are set up as shown in the following figure. Their values are better tuned when the portal is more populated by significant content. The following panel is accessible by the root administrator of the ECLAP portal.

Figure 5-1 Searching Settings

5.3.1 Digital Content Indexing

A JSP service, based on Tomcat, is available for content indexing. Drupal is allowed to call this external service with relevant parameters needed to index the content, retrieved by querying the MySQL database (metadata, documents attached etc.). The service indexes, updates and deletes existing contents.

Indexing service is implemented in Java using the SolrJ client API. Each time a user inserts, updates or deletes content from the Drupal portal, the service is called by a Drupal module, and then it starts assembling the object data to be indexed in Solr. Once completed the service returns a success/failure notification to the Drupal caller. At the time of indexing the indexing service extracts the content type and language by querying in the Drupal/Axmedis db tables, following this schema:

Content	Language
Drupal Page	Page language defined in Drupal db table
Drupal comment	Language of the page, group, object to which the comment is attached
Drupal Forum	Language of the group to which the forum is attached
Axmedis Object	Language of the object defined in Axmedis db table

Metadata are indexed in the Solr index using the following schema:

Solr Document Field	Type T = tokenized U = untokenized	Localization	Description
resource	String, U	NO	content resource (object, page, comment)

axoid	String, U	NO	Axmedis object id
axoidref	String, U	NO	Axmedis object ref
cluster	int	NO	Cluster
creationDate	Long	NO	Date of content creation (timestamp)
duration	String, U	NO	Content duration time
dx	int	NO	Content x resolution
dy	int	NO	Content y resolution
filesecco	String, U	NO	Resource extension (e.g. pdf)
governedObject	int	NO	1 if the object is protected, 0 otherwise
httpdownload	long	NO	Number of content downloads
iphone	int	NO	1 if the object is an iPhone content, 0 otherwise
lastModificationDate	long	NO	Last time content was modified (timestamp)
latitude	double	NO	Latitude
longitude	double	NO	Longitude
mobile	int	NO	1 if the object is a mobile content, 0 otherwise
modified	int	NO	1 if the content has been modified
nid	long	NO	Drupal id of the content
nohttp	int	NO	1 if http download link is available
nop2p	int	NO	1 if p2p download link is available
nvoti	long	NO	Number of votes for the content
p2pdownload	int	NO	Number of p2p downloads
pcview	int	NO	1 if the object if for pcview, 0 otherwise
pda	int	NO	1 if the object if for pda, 0 otherwise
playnumber	long	NO	Number of times the object was played
progressivertsp	int	NO	1 if the content is progressive rtsp, 0 otherwise
progressivepc	int	NO	1 if the content is progressivepc, 0 otherwise
published	int	NO	1 if the content has been published on portal
partner	String, U	NO	Name of the content owner partner
ranking	long	NO	Content raking
resolution	String, U	NO	Content resolution type (high = H, low = L)
sourcefile	String, U	NO	Content file path

sourceurl	String, U	NO	Content source url
stb	int	NO	1 if the content is for stb, 0 otherwise
xmlObjectUri	String, U	NO	Relative content path
deflanguage	String, U	NO	Original metadata language
videoQuality	String, U	NO	Video quality, M, H
workflow_type	String, T	NO	Workflow type
hidden	String, T	NO	Hidden object flag (1 = hidden, 0 public)
creator	String, U	NO	Creator name
taxonomy	String, U	NO	Taxonomies of the content
group	String, U	NO	Groups of the content
format	String, U	NO	Content format (ECLAP format)
date	String,U	NO	Content creation date (DC)
date_created	String, T	NO	
date_dateaccepted	String, T	NO	
relation_ispartof	String, T	NO	
relation_isreferencedby	String, T	NO	
relation_isreplacedby	String, T	NO	
relation_isrequiredby	String, T	NO	
format_extent	String, T	NO	
dcformat	String, T	NO	Content type (DC)
dctype	String, T	NO	Content type (DC)
type	String, U	NO	Content type (ECLAP type)
identifier	String, U	NO	Content identifier (DC)
language	String, U	NO	Content Metadata language (DC)
coverage	String, U	NO	Content coverage (DC)
relation	String, U	NO	Content metadata relation (DC)
contributor	String, U	NO	Content metadata contributor (DC)
taxonomyGenre	String, U	NO	Content taxonomy genre id if available
taxonomyHistoricalperiod	String, U	NO	Content taxonomy historical period id if available
taxonomyManagementandOrganisation	String, U	NO	Content taxonomy and organisation id if available
taxonomyPerformingArts	String, U	NO	Content taxonomy performing arts id if available
taxonomySubject	String, U	NO	Content taxonomy subject id if available
device	String, U	NO	Device

durationTime	String, U	NO	Duration time
uploadTime	date	NO	Date of upload
cid	long	NO	Comment drupal id
timestamp	String, U	NO	Timestamp
text_all	String, T	NO	Catchall field for document fields (e.g. doc, xls, pdf etc.)
title_all	String, T	NO	Catchall field for title field
body_all	String, T	NO	Catchall field for body field
description_all	String, T	NO	Catchall field for description field
contributor_all	String, T	NO	Catchall field for contributor field
subject_all	String, T	NO	Catchall field for subject field
taxonomy_all	String, T	NO	Catchall field for taxonomy field
All DC Fields (e.g. title, creator, format etc.)	Text, T	YES	DC metadata fields

5.3.1.1 Dependencies

Internal component	Type (drupal module, AXCP rule, Java Servlet, Other)	Notes
buildIndex.jsp service	Java Servlet	Password protected Java Web application
Axmedis module	Drupal module	Solr searching is coded in this module
luceneIndex module	Drupal module	Module that performs insert/update/delete of contents calling buildIndex.jsp service

Third-party software	Version	Notes
SolrJ API Client for Solr	Java API	Java client for Solr Index
Apache Tika	Java API	Library for document parsing

5.3.2 Rich Text Documents indexing

Documents attached to the content are sent to the Solr Index Instance through HTTP/XML for parsing. Solr Service uses the Apache Tika Library to perform the document parsing, supporting the main document formats (doc, docx, ppt, pptx, xls , xlsx, pdf, html, htm, txt etc.). The whole document structure is indexed by the Solr service and stored in a custom document field, based on the language of the document itself (e.g. doc_en, pdf_en, ppt_en etc.).

5.3.3 Rebuild Index Service

The rebuild service is a Java service written using the SolrJ API, and invoked as a Javascript rule by the AXCP Axmedis Platform. The service is allowed to retrieve needed data for each content item to be indexed as the standard indexing service interacting with Drupal. The rebuild service performs taxonomy/group extraction for each object, storing the result in a cache table in db for faster re-indexing.

5.3.4 Multilanguage Index

The Solr index is a single Multilanguage index, thus allowing fast access and easy management. Each of the above reported fields to be indexed must be tagged with a _locale suffix to allow language filtering for each field in the document structure at query time.

5.3.5 Digital Content Search

Each public document in the Drupal portal is searchable in the portal. This means that querying for a string contained in a Drupal page, forum, comment or an Axmedis object, results in matching the list of contents with that search term. Querying by taxonomy related to the content provides a match too.
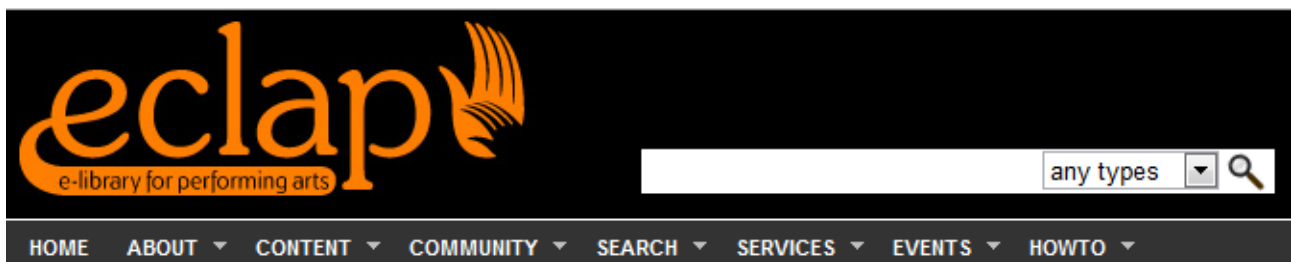

Figure 5-2 ECLAP Portal Top Menu

5.3.6 Simple and Advanced Search

Simple full-text search and advanced search are available for the user. Simple search is in the top center of the portal, and it allows the user to refine his search with a basic content type menu (archive, audio, cross-media etc.), while the advanced search provides extended functionalities (i.e. metadata search, or content search filtered by partner and language). User is allowed to compose an arbitrary number of Boolean clauses in the advanced search page, thus allowing the building of a rich metadata query; for example restricting the search to some metadata fields that only match any or all of them (OR/ALL).

Figure 5-3 ECLAP Advanced Search Page

5.3.7 Fuzzy search and wildcards

Searches can be performed with fuzzy techniques too. The querying string is compared to similar strings in the index to retrieve documents with a high degree of similarity (e.g. "documant" should match "document"), thus allowing an efficient search in case of mistyping. This fuzzy weight is customizable by the administrator in the portal. Each query string is automatically prefixed / suffixed with a special wildcard, in a transparent way to the user, to allow searching of substrings (e.g. query "test" matches "testing").

A checkbox in the full text frontal search lets the user enable/disable a deep search in the substrings.

5.3.8 Boosting of terms

Each field in the Solr document structure is boosted with a value, customizable by the administrator, so as to allow searching by giving more relevance to one field with respect to others. (e.g. title, subject, description etc.). For the search tuning, in the settings section of the portal, the administrator is able to change the boosting for these search fields:

- Title
- Body
- Description
- Subject
- Taxonomy

The administrator is able to change the fuzzy search similarity in the same section. (< 1 means fuzzy logic, $=$ 1 means Boolean).

5.4 Faceted search

Faceted search is allowed both from the simple and advanced search. Each faceted term is indexed untokenized in the Solr index, thus enabling a faceting count based on the whole facet. The user can select/remove any facet in any order to refine the search. Adding/removing a facet results in adding/deleting a search filter and performs again the search query with/without it. Relevant facets are:

- Resource Category
- Format
- Type
- Group
- Classification - Genre
- Classification – Historical Period
- Classification – Management & Organisation
- Classification – Performing Arts
- Classification – Subject
- Creator
- Content Language
- Duration
- Video Quality
- Device
- Publisher
- Original Metadata Language
- Upload Time

These facets can be subject to change. For instance, locations and dates (different from historical period) can be added.
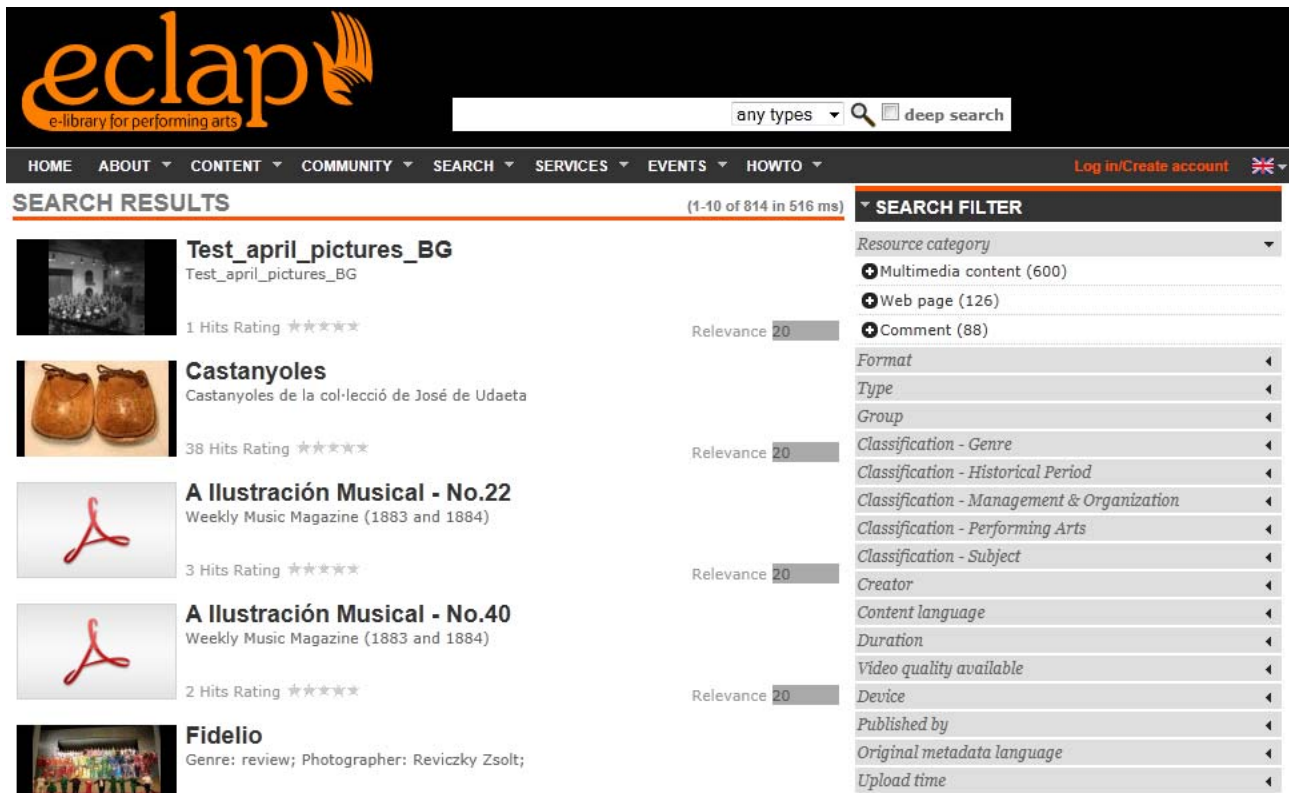
Figure 5-4 ECLAP Faceted Search

5.4.1 Result sorting and scoring

Matching documents are presented in relevance ascending order; this means that the first document is the one supposed to be more relevant to the query string, and the last one the less. Scoring is based on the occurrence of the query string in the indexed document fields: a higher number of occurrences of a string or similar strings produce a higher score for the document. Each document is presented as a result with a thumbnail and its relevant metadata (i.e. title, description, rating, creator, score) in the user language interface chosen by the user (or if not available in the English or the original metadata language). Results are paginated (typically 10 per page, but this setting can be changed by the administrator).

5.4.2 Queries/Views/Downloads Logging

Each query/view/download performed by the user on the portal (anonymous or authenticated) is logged in the MySQL database. Main details logged are:

- User id
- Country
- Region
- City
- Latitude
- Longitude
- Portal
- User agent
- Ip address
- Proxy (if used)
- Timestamp

- Query or view/download

The administrator can enable/disable logging from the settings in the administration section of the portal.

5.4.3 Indexing policy

Every public content item uploaded to the portal is indexed with full metadata and resources. Non public contents and their associated contents (e.g. private page and all comments related to it) are not indexed. The indexing service uses this strategy too.

6 Multilingual Portal interface and Automated Translation

The Portal Interface is now available in 21 different languages. Drupal provides functionalities in order to add other languages than the default English [1]. It is possible, to:

- Replace the text elements of the interface with customized ones.

- Enable more languages and translate the text.

When the system encounters text, it tries to translate it into the currently selected language; if the text has not been translated in the selected language, then the text appears in English. It is possible to change the language interface using the Language Switcher Dropdown (Figure 6-1).

Figure 6-1 Translate interface with available languages

The system allows two different methods to specify the text translations:

- Word by word: it is available as an integrated web interface through which it is possible to search and to change the text (Figure 6-2 Interface to translate a string).

- Importing other translations: it is possible to import some existing translations that are available as GNU gettext Portable Object files (.po files for short). Moreover, the export functionality also exists (Figure 6-3 Interface to import 'po' files with translations).

Figure 6-2 Interface to translate a string

Figure 6-3 Interface to import 'po' files with translations

For the automated translation of strings an AXCP rule has been implemented to find in the drupal table the strings that need to be translated and using the translation service they are translated.

7   Multilingual Portal Interface Validation and acceptance process

In the three years of ECLAP project the portal will go through several changes, which will affect the static texts in the interface. Static texts are the predefined textual elements of the ECLAP portal, such as menus, the header and footer of the website and the action buttons. As the ECLAP portal is multilingual, the changes of the portal will not only affect the texts in English, but also all of the 12 different translations of these texts.

Currently all static texts of the 12 languages are automatically translated from English, which is the default language. The reason for automatic translation is providing a starting point for translators to translate the 16.000 unique strings of the portal interface in order to make it easier to realise a full translation of the portal. As the ECLAP consortium has partners to represent each of the 12 defined languages, all of the translators are stationed within these partners' respective companies and should be native speakers of the language they are translating the English text into.

The next section describes how translators can provide corrections to the machine translated strings.

7.1   Collecting translation corrections

In order to communicate the corrections the translators make on the machine translated static texts in the portal a text file format has been defined describing the following elements:

Wrong string	A string that has been badly translated on the portal
New corrected string	The string being the correct translation of the wrong string
Language	The language code of the translation

Table 7-1. *Translation correction file format*

More specifically these corrections need to be provided in the way that each line in the file represents a correction in the following format:

<WRONG STRING> <NEW CORRECTED STRING> <LANGUAGE>

To give an example of a few lines from a file made by a Dutch translator:

<Uit te nodigen> <Meld je aan> <NL>
<Zie mijn profiel> <Mijn profiel> <NL>
<Nodig collega's> <Nodig collega's uit> <NL>

The files need to be saved in plain text and need to be sent to the developers of the ECLAP portal. These developers then will use the file to correct the related strings on the portal.

If preferred, translators can also use an Excel spreadsheet, which is displayed in the image below:

Incorrect string	New corrected string	language
Uit te nodigen	Meld je aan	NL
Zie mijn profiel	Mijn profiel	NL
Nodig collega's	Nodig collega's uit	NL

Figure 7-1. *Translation correction Excel template*

8   Multilingual SKOS Automated Translation

The translation of the Multilingual SKOS is performed automatically as for the other strings and as reported in the previous sections.

Moreover, in the same manner of the correction of the strings for the portal user interface, the Corrector/Translators are requested to provide correction with the same simple mechanism presented above. They can browse the taxonomy and providing the corrected strings.

ECLAP is going to provide as back office tools an instrument to maintain and/or perform the corrections and/or the validation of the translated strings. In this case, a Validator would have the possibility of accessing to a double table of terms with the English version and the corresponding translated string proposed in his/her mother tongue only. This model is strongly similar to the present ECLAP Metadata Editor tools already in place.

9 Multilingual Taxonomy Editor

This section describes the multilingual taxonomy editor. In D3.1: *Infrastructure: ingestion and processing content and metadata*, the functional requirements of this taxonomy were defined. In this section the current implementation is addressed in detail.

9.1 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.

9.2 Workflow

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).

9.3 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 9.4.2. This module was called the Multilingual Taxonomy Export module and was also implemented to be compliant with Drupal version 6.

9.4 Functionalities

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

9.4.1 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 9.3, 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 [2] . 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 9.1. Purpose & scope).

The initial screen of the editor is displayed in the figure below:
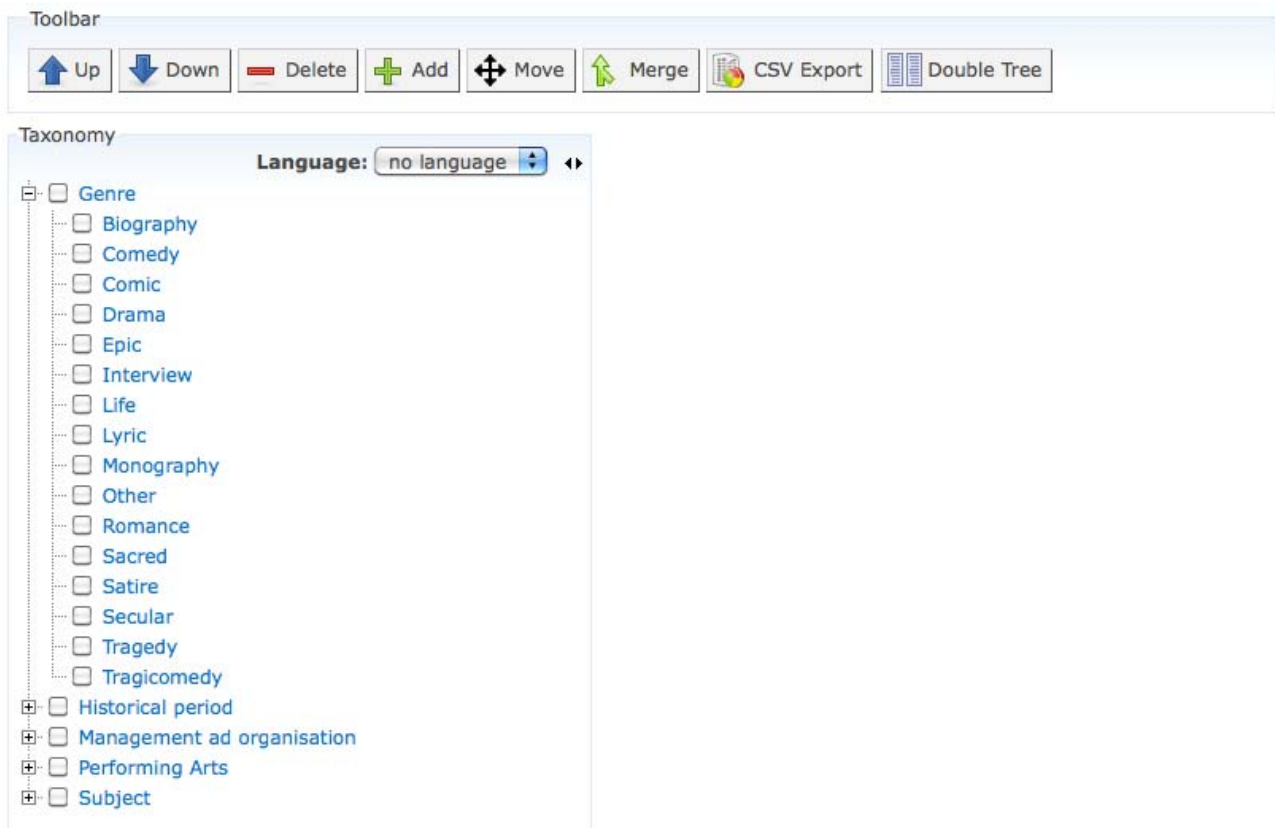
Figure 9-1 *Multilingual Taxonomy Editor: show taxonomy translations*

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.
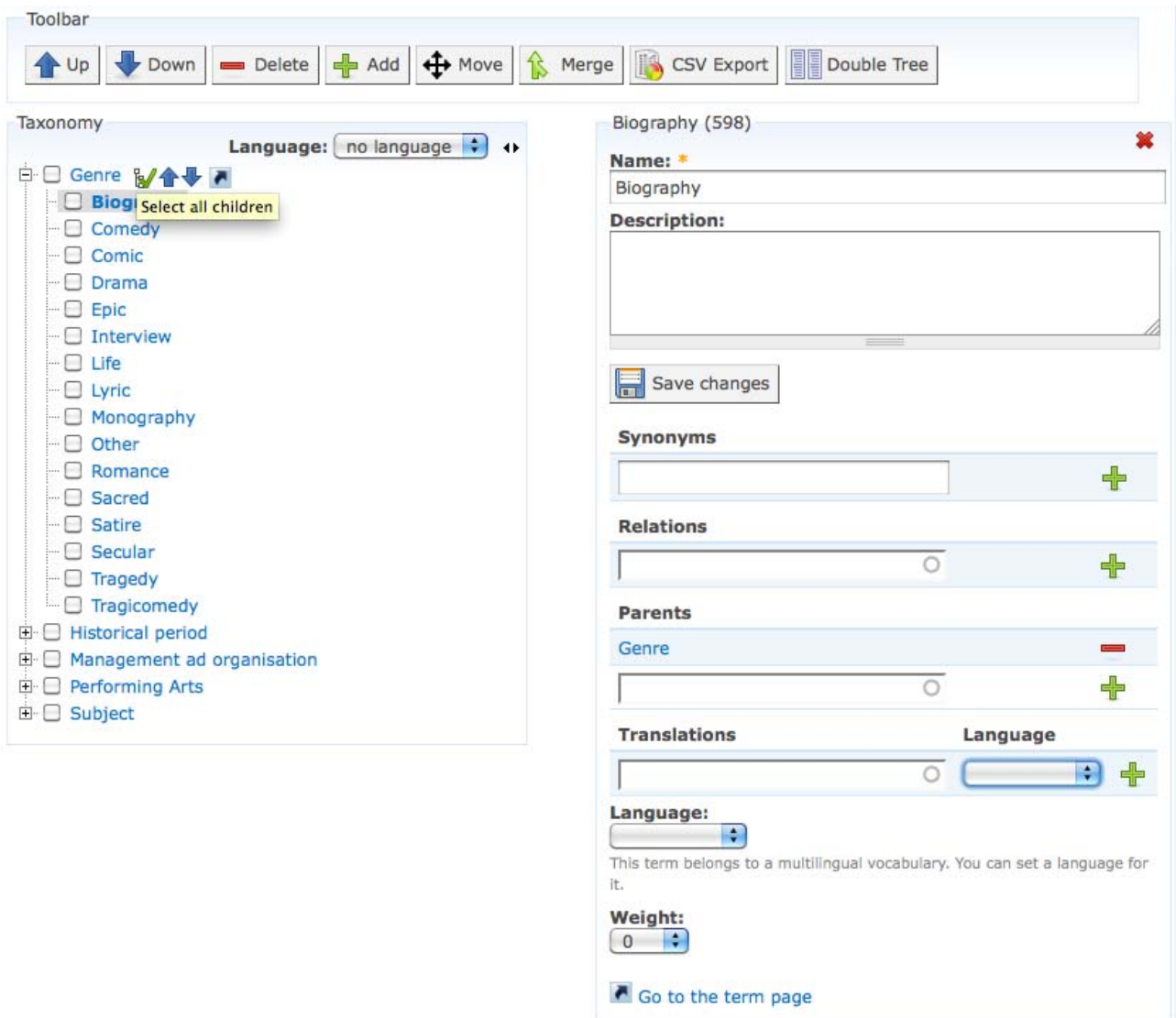
Figure 9-2 *Multilingual Taxonomy Editor: edit terms*

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:
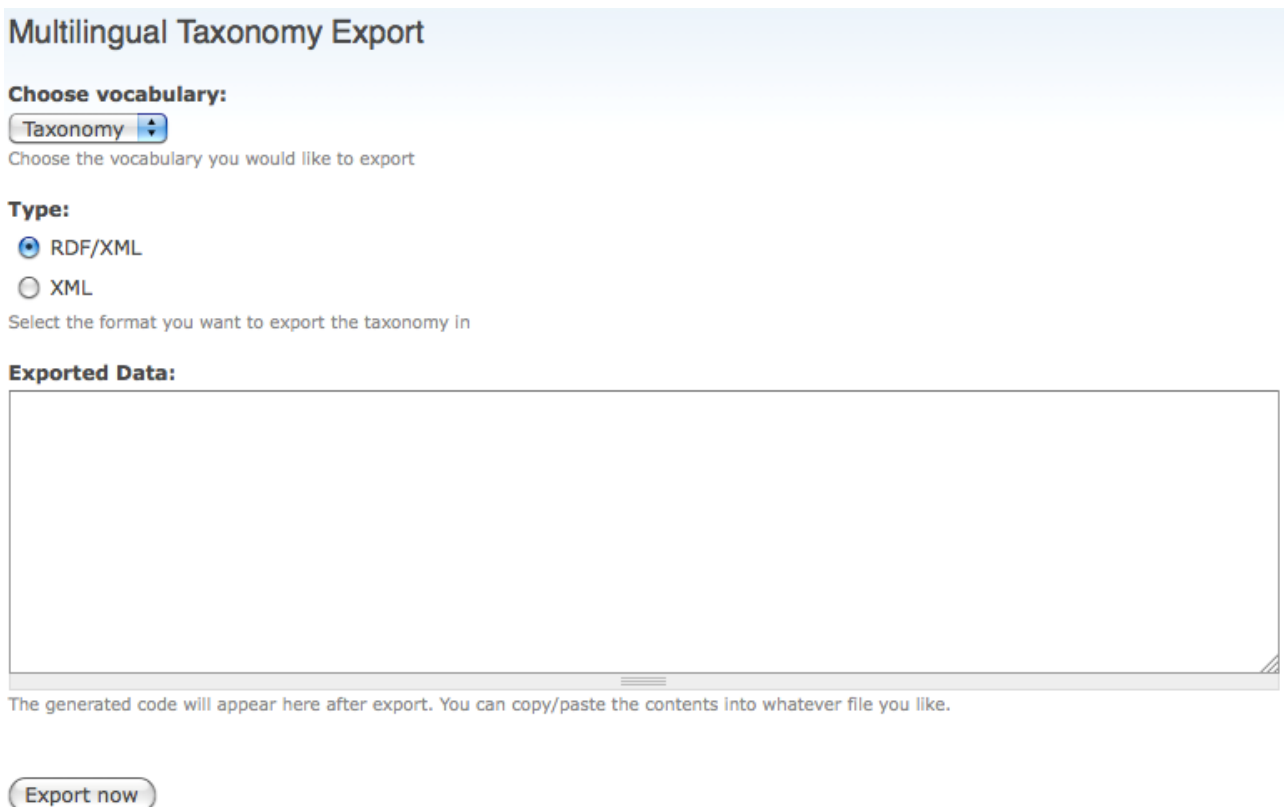
Figure 9-3 *Multilingual Taxonomy Editor: Adding translations*

9.4.2 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:

Figure 9-4 *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 9-5 *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 9-6 *Multilingual Taxonomy Export: exported XML data*

10 Multilingual Metadata and GUI on Mobile

Mobile applications for iPhone/iPad allow the download of content from ECLAP portal and use it locally.

The user with the mobile application can:

- Download content and metadata from the portal.
- Access to the metadata of the content downloaded.
- Open the content.
- Search for keywords on the metadata.
- Browse the downloaded content via taxonomy.

For the iPhone application the standard internationalization features of iOS applications has been used. It allows using the English and Italian user interfaces (the translations for other languages can be done quite easily), if the device language set is not English or Italian the English one is used.

For the metadata shown to the user it is tried to show the metadata in the current device language, if not present the English metadata is used. See Figure 11.1 where on the left is present the English user interface while on the right is present the Italian one. In Figure11.1 has been highlighted an object shown with English title when using the English language and the Italian title when using the Italian language. In figure 11.2 the metadata section of the same content is shown using English, Italian and French languages. It can be seen that the description is shown in the appropriate language.

When searching a text the match is done on all the metadata languages available not only the current language.
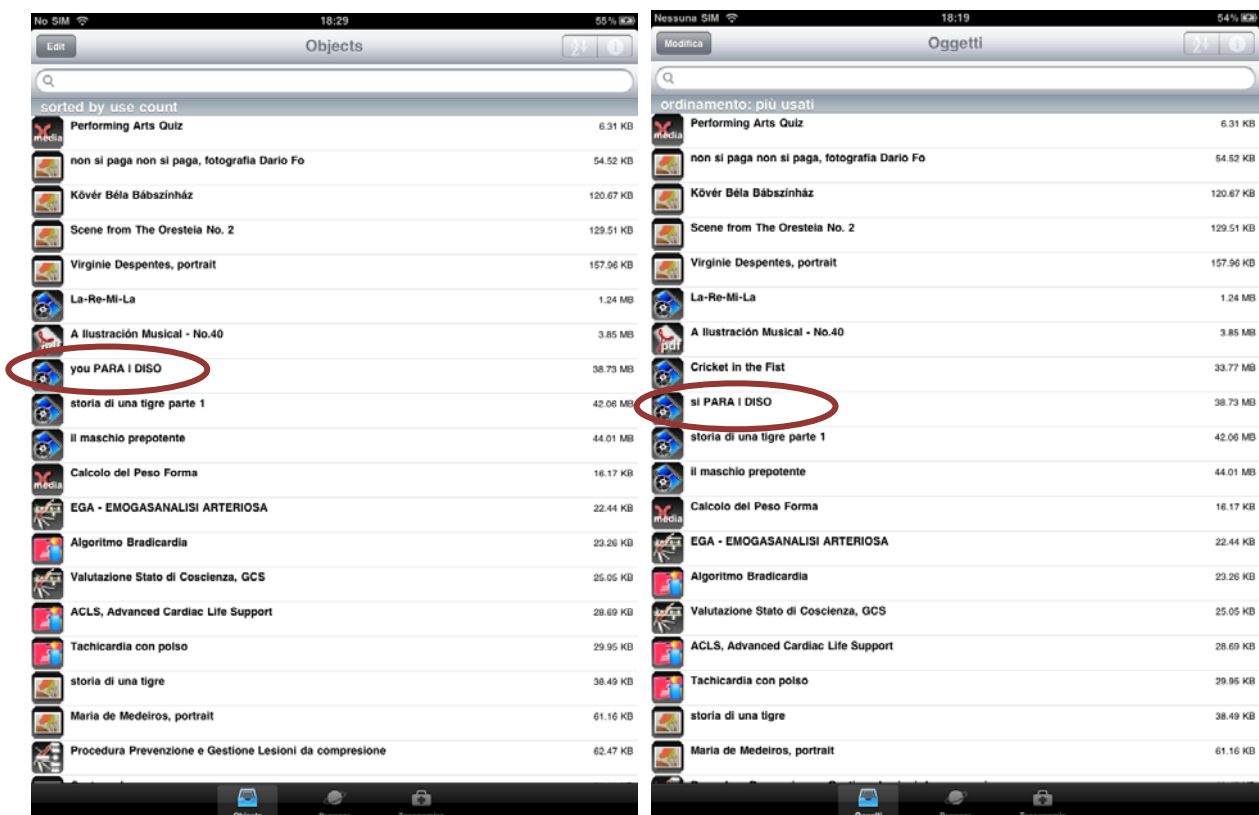
Figure 10-1 English and Italian user interface
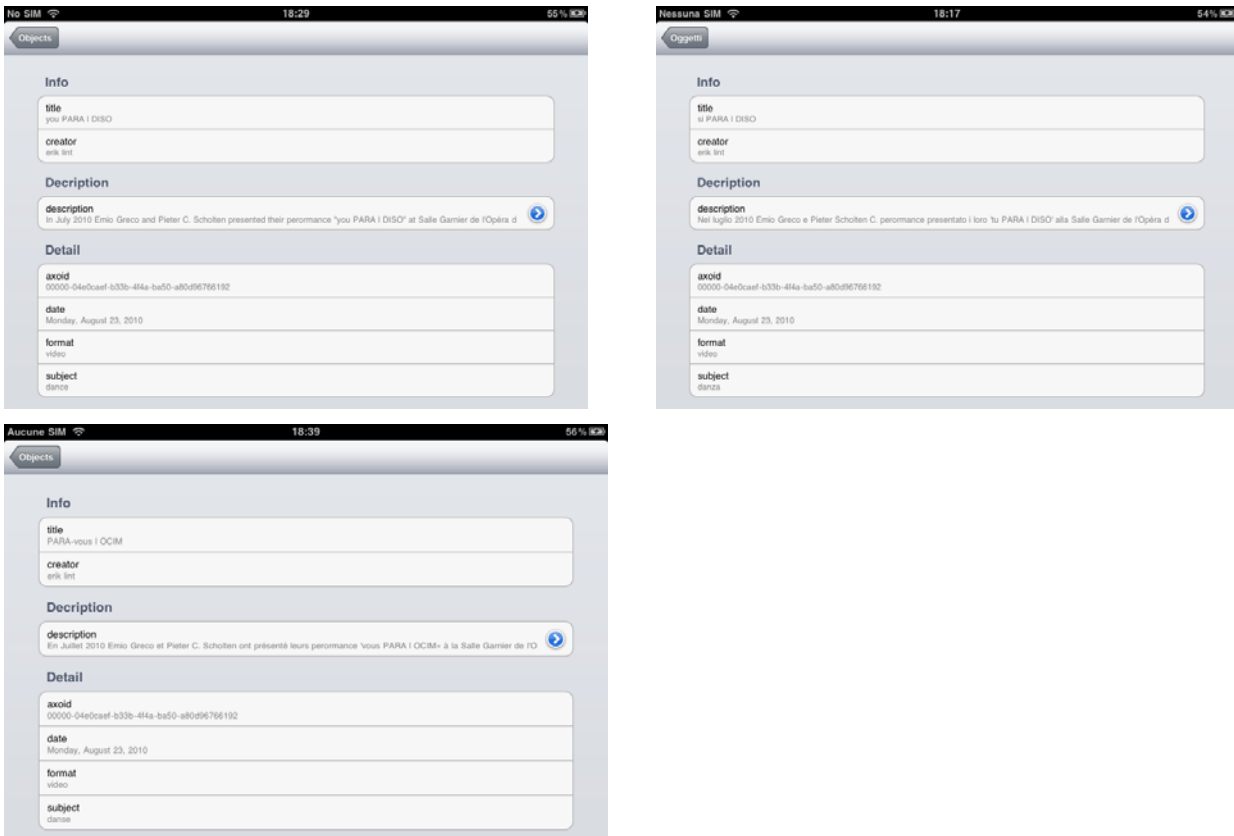
Figure 10-2 Metadata shown in English, Italian and French.

11 Accessibility Assessment Model

11.1 Web Accessibility Analysis

Focusing on Internet accessibility, one of the most relevant sets of guidelines are those developed by the Web Accessibility Initiative (WAI) of the World Wide [1] Web Consortium (W3C). W3C-WAI has established three sets of W3C recommendations to improve the accessibility of the Web. These are:

- WAG (Web Content Accessibility Guidelines), released in May 199, which concern how to make Web sites sufficiently accessible so that people with disabilities are able to use them alongside with today's [4] technologies.

- ATAG (Authoring Tool Accessibility Guidelines), released in February 2000, which provide guidance for software developers in designing authoring tools that produce accessible Web content and in creating accessible authoring interfaces [5].

- UAAG (User Agent Accessibility Guidelines), released in December 2002, which are concerned with how to make Web browsers and multimedia players more accessible, as well as being compatible with some of the assistive technology that people with disabilities use [6].

The strength of WAI is both their universal acceptance and the way they are produced (cooperatively, in a short period of time and in a clear way). However, these broadly accepted and used guidelines are far from being definitely established.

This remaining of this section provides general procedures and tips for evaluation in different situations, from evaluation during Web site development to ongoing monitoring of existing sites. The approaches in these pages are intended to supplement other content management and quality assurance procedures.

11.1.1 Preliminary review

A *preliminary review* can quickly identify some accessibility problems on a Web site. A preliminary review does not check every accessibility issue and will not catch all of the problems on a site. Thus the method described in this page is not sufficient to determine if a Web site conforms to Web accessibility guidelines.

A preliminary review combines some manual checking of representative pages on a Web site, along with the use of several semi-automatic accessibility evaluation tools. Reviewers do not need to know Web mark-up languages, but should be able to download software and familiarize themselves with some evaluation tools, and change certain settings on their browser. To conduct a preliminary review, the following tasks should be completed.

Select a representative page sample. From the Web site to be reviewed, select a *representative sampling* of pages that match the following criteria:

- Include all pages on which people are more likely to enter your site ("welcome page", etc.).
- Include a variety of pages with different layouts and functionality, for example:
  - o Web pages with tables, forms, or dynamically generated results;
  - o Web pages with informative images such as diagrams or graphs;
  - o Web pages with scripts or applications that perform functionality.

Examine pages using graphical browsers. Use a graphical user interface (GUI) browser (such *as Firefox, Internet Explorer, Netscape Navigator, Opera, Safari*, or others) and examine the selection of pages while adjusting some settings in your browser or operating system as follows (some of these manual checks may require additional software):

1. Turn off images, and check whether appropriate alternative text for the images is available.
2. Turn off the sound, and check whether audio content is still available through text equivalents.

3. Use browser controls to vary font-size: verify that the font size changes on the screen accordingly; and that the page is still usable at larger font sizes.

4. Test with different screen resolution, and/or by resizing the application window to less than maximum, to verify that horizontal scrolling is not required (caution: test with different browsers, or examine code for absolute sizing, to ensure that it is a content problem not a browser problem).

5. Change the display color to gray scale (or print out page in gray scale or black and white) and observe whether the color contrast is adequate.

6. Without using the mouse, use the keyboard to navigate through the links and form controls on a page (for example, using the "*Tab*" key), making sure that you can access all links and form controls, and that the links clearly indicate what they lead to.

Note: Browser extensions and other plug-in evaluation tools (such as *AIS Toolbar for Internet Explorer and Opera*, *WAVE Toolbar for Firefox, Internet Explorer, and Netscape Navigator,* or *Web Developer Extension for Firefox*) provide functionality to help perform many of these manual checks. For reviewers who have disabilities, certain of the following tasks may need to be done with another person who does not have the same disability.

Examine pages using specialized browsers. Use a voice browser (such as *Home Page Reader*) or a text browser (such as *Lynx*) and examine the selection of pages while answering these questions:

1. Is equivalent information available through the voice or text browser as is available through the GUI browser?

2. Is the information presented in a meaningful order if read serially?

Note: experienced users of screen readers may substitute a screen reader for a voice or text browser, but if blind, may need a sighted partner to compare information available visually; if sighted, listen to it with eyes closed, then open eyes and confirm whether the information is equivalent.

Use automated Web accessibility evaluation tools. Use at least two automated Web accessibility evaluation tools (see following sections) to analyze the selection of pages and note any problems indicated by the tools. **Note:** these tools will only check the accessibility aspects that can be tested automatically; the results from these tools should not be used to determine a conformance level without further manual testing.

Summarize obtained results. Summarize results obtained from previous four tasks:

1. Summarize the types of problems encountered, as well as positive aspects that should be continued or expanded on the site.

2. Indicate the method by which problems were identified, and clearly state that this was not a full conformance evaluation.

3. Recommend follow-up steps, including full conformance evaluation which includes validation of markup and other tests as described below, and ways to address any problems identified.

11.1.2 Conformance Evaluation

A *conformance evaluation* determines if a Web site meets accessibility standards, such as the Web Content Accessibility Guidelines (WCAG) [1]. This page describes a conformance evaluation method that combines automatic, semi-automatic, and manual testing of Web site accessibility. It can be used when developing a new site, or to evaluate an existing site.

It mainly focuses on technical assessment and does not include involving users with disabilities, which is addressed in [5]. Including users in evaluation helps ensure that technical accessibility solutions are applied effectively. Evaluations that combine technical assessment and usability testing of accessibility can be called *comprehensive evaluations*.

The conformance evaluation method described below requires:

1. Familiarity with Web mark-up languages (such as HTML),
2. Access to and skill with a variety of evaluation tools and approaches, and
3. Knowledge and experience in Web accessibility.

A conformance evaluation includes all of the tasks below except those that are explicitly identified as alternatives or optional. <u>Note</u>**:** while determining the scope of the evaluation is a key first task, and summarizing and reporting the results of evaluation is the logical conclusion, the order of the intervening tasks is not crucial.

<u>Determine the scope of the evaluation.</u> Determine and disclose scope of site to be evaluated and the targeted conformance level for the evaluation. <u>Note:</u> disclosure should be internal to the organization during evaluation; if conformance is claimed publicly, disclose externally (for example, on the Web site).

1. Determine and disclose the target conformance level of WCAG 1.0.
2. Select a *representative sampling* of pages for manual evaluation that match the following criteria:
    o Include all pages on which people are more likely to enter your site ("welcome page", etc.)
    o Include a variety of pages with different layouts and functionality, for example:
        ▪ Web pages with tables, forms, or dynamically generated results;
        ▪ Web pages with informative images such as diagrams or graphs;
        ▪ Web pages with scripts or applications that perform functionality.

    Note: there are special considerations for Web sites with database driven dynamically generated Web content.
3. Identify and disclose the *entire Web site* including all pages at a base URL for automatic and semi-automatic evaluation; <u>Note:</u> if testing of the entire site is not feasible (for example, because of its unusual size or dynamic nature) identify an *expanded page selection*, to be clearly explained and disclosed on the Web site. Suggestions for inclusions in this expanded page selection: pages from different sections of the Web site; pages representing different "look & feel"; pages representing different development tools and processes including those generated from databases; pages produced under different guidelines; "contact us" pages; pages critical to your business; etc. If any area of a site is excluded from evaluation, be sure to disclose this information.

<u>Use Web accessibility evaluation tools.</u> Validate markup including syntax and style sheets, using all applicable validators, on the selected sample of pages. Run at least one validation tool across entire Web site (or *expanded page selection*):
    o HTML Validation service [8] ;
    o HTML Tidy [9];
    o CSS Validation service [10];
    o MathML Validator [11].
2. Use at least two Web accessibility evaluation tools on the selected sample of pages and run at least one tool across entire Web site (or *expanded page selection*). Note any problems indicated by the tools. <u>Note:</u> Using at least two evaluation tools helps catch potential misidentification of accessibility problems that might result from using a single evaluation tool.

<u>Manually evaluate representative page sample.</u>

Apply accessibility checklist to page sample. Examine the selected sample of pages using checkpoints from the Checklist of Checkpoints for Web Content Accessibility Guidelines 1.0 [12] that are applicable to your site. <u>Note:</u> Applicable can mean checkpoints that cannot be evaluated by automatic or semi-automatic tools; checkpoints that actually apply to the site (for example, if site contains no audio content, skip those); and, as a minimum, those checkpoints that apply to the level of conformance you are evaluating.

Examine pages using graphical browsers. Examine the selected sample of pages with graphical user interface (GUI) browsers: select at least three different configurations from among the following variables: different graphical user interface browsers (such as *Internet Explorer*, *Mozilla Firefox*, *Netscape Navigator*, *Opera, Safari*, or others), in different versions (latest, older), running on different platforms (Windows, Linux, Mac) and making the following adjustments:

1. Turn off images, and check whether appropriate alternative text is available.

2. Turn off the sound, and make sure audio content is still available through text equivalents.

3. Use browser controls to vary font-size: verify that the font size changes on the screen accordingly; and that the page is still usable at larger font sizes.

4. Test with different screen resolution, and/or by resizing the application window to less than maximum, to verify that horizontal scrolling is not required (caution: test with different browsers, or examine code for absolute sizing, to ensure that it is a content problem not a browser problem).

5. Change the display color to gray scale (or print out page in gray scale or black and white) and observe whether the color contrast is adequate.

6. Without using the mouse, use the keyboard to navigate through the links and form controls on a page (for example, using the "*Tab*" key), making sure that you can access all links and form controls, and that the links clearly indicate what they lead to.

7. Also examine page with scripts, style sheets, applets, and other embedded objects not loaded.

Note: Browser extensions and other plug-in evaluation tools (such as *AIS Toolbar for Internet Explorer*, *WAVE Toolbar for Firefox*, *Internet Explorer*, and *Netscape Navigator*, or *Web Developer Extension for Firefox*) provide functionality to help perform many of these manual checks. For reviewers who have disabilities, certain of the following tasks may need to be done with another person who does not have the same disability.

Examine pages using specialized browsers. Examine the selected sample of pages with one text browser (such as *Lynx*) and one voice browser (such as *Home Page Reader*), and answer the following questions:

With text browser:

- Is equivalent information and function (for example, links and scripted events) available through the text browser as is available through the GUI browser?

- Is the information presented in a meaningful order when read serially?

With voice browser:

- Is equivalent information available through the voice browser as is available through the GUI browser?

- Is the information presented in a meaningful order when spoken serially?

Note: *for settings where there is limited choice of assistive technologies*, also perform a manual evaluation of the Web site with those assistive technologies; for instance, JAWS is the only screen reader translated into Danish, and therefore in Denmark a trained evaluator should evaluate the Web site using JAWS.

Read and evaluate page content. Read over the selected sample of pages: is the text clear and simple to the extent appropriate for the purpose of the Web site? (For English sites, consider using Clear and Appropriate Language and Design (CLAD) [13] test.)

Summarize and report findings. Summarize any problems and best practices identified for each page type and a representative URL, and method by which they were identified. Recommend follow-up steps, potentially including:

- Repair of accessibility barriers identified through conformance evaluation process.

- Expanding positive aspects on site.

- Ongoing maintenance and monitoring of site.

Note: for evaluations using an expanded page selection instead of the entire Web site, apply what you've learned to other pages.

11.1.3 Specific Contexts

Evaluations approaches for specific contexts refers to considerations for evaluation of large and complex Web sites, during the development process, ongoing monitoring, evaluation of legacy sites, and evaluation of dynamically generated Web pages.

Evaluation during the development process. Evaluation during the development process is essential. It can sometimes be difficult, as both in-house and subcontracted Web developers sometimes prefer to establish the site design and demonstrate their progress before getting feedback. However, accessibility issues identified early are easier to correct and avoid. Effective evaluation during the design period can include:

- Establishing clear requirements for the expected accessibility conformance level.
- Involvement in initial planning meetings for the site.
- Agreeing on a review schedule during the development process.
- Providing information on evaluation approaches so that the developers can at least do preliminary reviews on their own.

Ongoing monitoring. To maximize likelihood that a Web site will maintain a given conformance level in the future, the following provisions should be in place:

- Clear statement of expected conformance level and scope of Web site it applies to.
- Clearly identified individuals responsible for monitoring the site, and follow-up procedures they can use to rapidly bring non-conformant pages into conformance.
- Clear expectations with regard to frequency, method, and scope of evaluations.
- Processes for validating and evaluating all changed pages and new types of pages before they are added to the site.
- Software to facilitate evaluation.
- Incorporation in Web site of address for feedback on accessibility of site.
- Automated or semi-automated tests to identify problems identified in the comprehensive evaluation.

Note: A full conformance evaluation is not necessarily required at each milestone in an ongoing review process. Steps like repeated user testing may only be required after major template or content changes.

Evaluation of legacy sites. Occasionally Web sites that are "frozen" (legacy; no longer actively maintained) are found to have substantial accessibility problems. It can be difficult to determine how to address these. It is helpful to:

- identify who the current owner is;
- determine whether they have any obligation or interest in making the site accessible;
- after evaluating the site, outline for the owner the changes that would be required to retrofit the site for accessibility;
- identify and propose resources and a timeline for retrofitting the site;
- disclose, publicly, accessibility problems on the site.

Evaluation of dynamically generated Web pages. Dynamically generated pages are usually assembled from one or more templates that provide common layout and navigation features, and content provided automatically from a database or other content management system. To achieve full conformance the accessibility of both templates and generated content must be evaluated. It is not sufficient to evaluate only templates: content may also contain markup, or be required to contain markup in order to be accessible. The following are things to consider:

Templates. Evaluate all templates as follows:

1. Evaluate static templates using the Web Content Accessibility Guidelines 1.0.
2. Add a minimal amount of plain text content to the templates and evaluate again.

Note: if templates are generated by authoring software evaluate the capability of the authoring tool to include accessible features, for example:

- Does the generated tab order from the template allow getting to the generated text content effectively?

Content. Evaluate the capability of the content management system to store and generate accessibility. Consider the following questions:

- Are images supplied with alt texts, and if needed, the longdesc?
- Do generated data tables have accessibility aids (for example: captions, id on th header cells, etc.)?
- If generated video, is it captioned?
- If generated audio narrative, is textual equivalent available?
- Is the generated markup valid?

Templates and content combined. For pages that are generated as a result of a query to a database, the source generated as the page is rendered should be captured and evaluated using the Web Content Accessibility Guidelines 1.0 -- [may require operator intervention].

Note: if all dynamic content cannot be evaluated, generate broadly representative samples, capture content, and test the output. Do the generated pages retain the accessibility features evaluated under templates and content when combined?

11.1.4 Involving Users in Evaluating Web Accessibility

Web accessibility evaluation often focuses on conformance to accessibility standards such as WCAG [14]. While conformance is important, there are many benefits to evaluating with real people to learn how your website or web tool really works for users and to better understand accessibility issues. Evaluating with users with disabilities and with older users identifies usability issues that are not discovered by conformance evaluation alone.

Initial Review. A first step in evaluating web accessibility is reviewing the website to check for any obvious accessibility problems. The Preliminary and Conformance Evaluation sections provide guidance.

Even web developers with little accessibility knowledge can find some accessibility issues through a preliminary review. An accessibility expert with first-hand experience of how people with different disabilities interact with the web can:

- evaluate accessibility issues for a broad range of users, which might not be found by individual users;
- help fix any known barriers before bringing in users; and
- focus the evaluation with users on potential areas of concern.

The initial review identifies any significant accessibility barriers to fix before evaluating with users. It also helps define what to focus on for evaluation with users.

Range of User Evaluation . Users with disabilities and older users can be included in a wide range of evaluation activities, from brief consultations to large-scale usability studies. There are many options in between these extremes**:**

- Informal evaluation of a specific accessibility issue can be as simple as asking someone you know who uses a screen reader, someone with other disabilities, or even your grandmother, to find some data in an early draft of a data table that you are developing, observing her interaction, and discussing issues.
- Formal usability testing of a website follows established protocols to gather quantitative and qualitative data from representative users performing specific tasks. Formal usability tests can be optimized to focus on accessibility issues.

What type of evaluation you do depends on factors such as the stage in your project, for example, initial investigation of design ideas, testing specific areas of prototypes, or reviewing final designs. Conducting informal evaluations throughout development is more effective than only formal usability testing at the end of a project.

Basics. In most cases, including users in evaluation involves:

- getting a few people with disabilities - and depending on your target audience, older users,
- including them throughout the development process to complete sample tasks on prototypes so you can see how the different aspects of the design and coding could be improved, and
- discussing accessibility issues with them.

Just as with any evaluation with users, whether you include novice, average, or advanced users depends on your target users. For example, if you are developing a web application to be used by accountants inside a company, you probably want advanced assistive technology users; for a public website to apply for disability benefits, you want novice assistive technology users.

Caution: Carefully consider all feedback and avoid assuming that feedback from one person with a disability applies to all people with disabilities**.** A person with a disability does not necessarily know how other people with the same disability interact with the web, nor know enough about other disabilities to provide valid guidance on other accessibility issues.

Note: In addition to finding accessibility problems, evaluating with users with disabilities usually reveals general usability problems that impact all users, including those *without* disabilities.

Analyzing Accessibility Issues

Web accessibility depends on several components of web development and interaction working together, including web browsers, assistive technologies (AT), and web content.-Accessibility problems can be caused by one or more different components. For example, if a user who cannot use a mouse has trouble with keyboard access, it could be because:

- the developer did not markup/code the web page properly, or
- the browser or media player isn't handling the markup properly, or
- the user's AT isn't handling the markup properly, or
- the user doesn't know how to use the browser, media player, or AT's keyboard access features, or
- the page is poorly designed and it is a general usability problem for all users, including those *without* disabilities.

Combine User Evaluation with Standards. Involving users with disabilities in evaluation has many benefits; however, it alone cannot determine if a website is accessible. Combine user involvement with evaluating conformance to WCAG to ensure that accessibility is provided to users with a range of disabilities and situations.

Drawing Conclusions and Reporting.

Be careful drawing conclusions from limited evaluations or studies. Results from only a couple of people with disabilities cannot be generalized to apply to all people with similar disabilities or people with other disabilities, as mentioned above.

Reports should include the scope of the study and the evaluation parameters, such as the testing methods and the user characteristics. For example, if a study included only usability testing with participants who are blind, its report should clarify that it did not evaluate conformance to accessibility guidelines and that it does not apply to all people with disabilities. Thus the report can help readers draw appropriate conclusions.

While small studies often provide useful information, they are not robust enough to provide statistical significance.

Note for usability professionals. When specifically exploring accessibility barriers, the protocol is usually different from a general usability test; for example:

- you would likely use a think-aloud technique with high facilitator interaction;
- data collection would focus on understanding errors related to accessibility barriers, rather than on time-on-task or user satisfaction; and
- tasks would concentrate on specific areas of concern for potential accessibility problems, rather than general site usage.

Note that is also important to evaluate general usability, user satisfaction, and other such criteria for users with disabilities.

The More Information section below includes additional guidance specifically for usability professionals.

More Information and Guidance . This section document briefly addresses a few points of a very complex topic. Many resources on other aspects of involving users in evaluation are available on the Web.

- Involving Users in Web Projects for Better, Easier Accessibility is a prerequisite for this document that covers broader issues of including users *early* in website design, tool development, standards, and other web projects.
- Just Ask: Integrating Accessibility throughout Design provides guidance on incorporating accessibility throughout design of websites and other products. The chapter on Usability Testing for Accessibility includes:
  - o Planning for usability testing for accessibility – determining participant characteristics, recruiting participants, providing compensation.
  - o Preparing for usability testing for accessibility – preparing test materials, ensuring the facility is accessible, setting up and testing the assistive technology, conducting a pilot test, using screening techniques [15].
  - o Conducting a usability test for accessibility – interacting with people with disabilities, setting up the room.
  - o Reporting usability testing for accessibility – distinguishing between accessibility and usability issues, drawing conclusions, writing about people with disabilities.
- White paper: conducting user evaluations with people with disabilities [17]
- Many books, articles, conference presentations, and other resources cover usability evaluation techniques, including different types of usability testing; test design; developing test protocol including questionnaires, tasks, data collection; conducting pilot tests; and how many participants to include in usability testing. For example: [18], [19], and [20] with a sample usability test, scripts, and forms online.
- There are organizations around the world that specialize in helping recruit people with disabilities and conduct evaluations with users with disabilities.

11.1.5 Using Combined Expertise to Evaluate Web Accessibility

Evaluating the accessibility of Web content for people with disabilities requires diverse kinds of expertise and perspectives. While it is possible for individuals to evaluate Web accessibility effectively if they have training and experience across a broad range of disciplines, it is less likely that one individual will have all the expertise that a collaborative approach can bring.

Recommended Expertise. Effective evaluation of Web accessibility requires more than simply running an evaluation tool on a Web site. Comprehensive and effective evaluations require evaluators with an understanding of Web technologies, evaluation tools, barriers that people with disabilities experience,

assistive technologies and approaches that people with disabilities use, and accessibility guidelines and techniques.

Approaches for Collaborative Evaluation. When first conducting a Web accessibility evaluation, the initial approach in many organizations is to assign the task to an individual within the organization, or to outsource it. However, many organizations use a collaborative evaluation process involving the skills and perspectives of multiple evaluators. This approach allows an organization to use in-house expertise as well as outside experts where needed.

Collaborative evaluation processes can involve:

- a group of colleagues distributed within a larger organization; for instance, Web developers from different units of a large corporation working together
- a Web development or quality assurance team within a larger organization, which brings in outside experts to help them conduct evaluations in the short term and helps them build improved capability for in-house evaluation over the long term
- a small business whose mission is to provide accessibility evaluation services, and which does so with a multi-disciplinary team
- disability advocates from different organizations who collaborate in online fora to monitor accessibility of Web sites
- a group of individuals distributed across related organizations such as government agencies, each with the obligation to monitor accessibility of their own Web site, who combine their diverse expertise & perspectives for higher quality evaluations

Considerations in Combining Expertise.

Centralized versus distributed evaluation capability. Organizations with in-house evaluation capacity sometimes use a centralized group of evaluators, and sometimes use evaluators who are distributed across the organization. A centralized team can serve as a resource for the rest of the organization. Evaluation capability that is distributed across an organization may offer more possibilities for integrating accessibility work into Web development processes throughout the organization. It may help in identifying more diverse expertise since one can look beyond the boundaries of a centralized team. In addition, it can help in developing a shared organizational mission for continual improvement of accessibility, rather than leaving oversight of accessibility as the responsibility of a single office.

Identification of external expertise. Once gaps in internal expertise are clear, an organization can prioritize its needs for external expertise. The internal gaps are often in areas of knowledge specific to disability and/or accessibility; for instance, Web accessibility guidelines, cross-disability accessibility barriers, or use of assistive technologies. In addition, even organizations with established user testing processes may need guidance on how to get feedback from users with disabilities. It can be valuable in some cases to bring in more than one outside expert to cover this range of issues effectively, or to look for feedback in online communities focusing on Web accessibility.

Involving users in evaluation. Inclusion of people with disabilities in a collaborative group can contribute to a better understanding of accessibility issues within the organization, and/or to maintaining awareness of the urgency of addressing accessibility barriers on a site, in addition to their individual technical contributions to the evaluation.

Regardless of the collective expertise of a collaborative group of evaluators in conducting conformance evaluations, an organization may want to ensure periodic review by users with a variety of disabilities. There are many factors to consider in effectively involving users in Web accessibility evaluations, including ensuring diversity in disabilities represented, types of assistive technology used, and experience with the Web.

Facilitating collaboration through shared tools and templates. A group of evaluators may want to arrange for shared access to certain evaluation tools, or to ensure that they have access to a broad range of evaluation tools across the group as a whole.

Using an agreed-upon template for reporting the results of evaluations can greatly facilitate coordination between different evaluators

Communicating results .Collaborative teams may want to give particular attention to communicating the results of their evaluations to their customers clearly, since their reports represent the combined perspectives of different evaluators.

Getting and giving feedback. Providing a mechanism for feedback within an organization on the usefulness of the evaluation process and resulting report may assist collaborative evaluators in ongoing identification of gaps in expertise, and contribute to long-term improvement in the quality of evaluations.

Feedback from experienced groups of evaluators on evaluation resources such as W3C/WAI's resource suite Evaluating Web Sites for Accessibility can, over time, also help improve the quality of evaluation support resources available to the broader Web community.

11.2 Accessibility Assessment Tools

Web accessibility evaluation tools are software programs or online services that help determine if a Web site is accessible. This document highlights different features of evaluation tools which can assist during evaluation reviews such as the methodologies described in the WAI resource Evaluating Web Sites for Accessibility.

What evaluation tools can do. Web accessibility evaluation tools can significantly reduce the time and effort required to carry out evaluations. When used carefully throughout the design, implementation, and maintenance phases of Web development, these tools can assist their users in preventing accessibility barriers, repairing encountered barriers, and improving the overall quality of Web sites. The following are ways in which tools can assist users in evaluating Web sites for accessibility; some tools can perform both:

- Determine the conformance of Web sites to accessibility checks which can be executed automatically;
- Effectively assist reviewers in performing accessibility checks which need to be evaluated manually.

What evaluation tools can *not* do. Many accessibility checks require human judgement and must be evaluated manually using different techniques. Also, in some cases evaluation tools are prone to producing false or misleading results such as not identifying or signal incorrect code. The results from evaluation tools should not be used to determine conformance levels unless they are operated by experienced evaluators who understand the capabilities and limitations of the tools in order to achieve accurate results. Web accessibility evaluation tools cannot *determine* the accessibility of Web sites, they can only *assist* in doing so.

11.2.1 Considerations for selecting evaluation tools.

Web accessibility evaluation tools can be used throughout all stages of Web site development. For example at the early design stage, Web designers may be interested in using tools that help them understand how the site structure, navigation, or look-and-feel perform with respect to accessibility requirements. Later at the implementation stage, developers may be more interested in tools that help them assess the accessibility of the underlying code which is generated by the Web authoring tools (such as editors or content management systems). Web content authors, project managers, and other types of Web site developers have further requirements for evaluation tools that help them fulfill their respective tasks.

According to the specific organization and Web site for which evaluation tools will be used, different characteristics and features of Web accessibility evaluation tools may be more or less important to the tool users. For example, an organization may choose to use fully automated evaluation tools which can examine the whole Web site, and additionally evaluate samples of pages using other types of tools in order to compensate the limitations of fully automated checking. The following are some of the factors which may be considered during the selection of Web accessibility evaluation tools:

Organizational structure and development process. For larger organizations or when several types of Web developers (such as designers, programmers, content authors, quality assurance reviewers, or others) participate in the development of the site, it may be beneficial to use a combination of evaluation tools in order to balance the capabilities of the tools, and to address the needs of the different user roles throughout the development stages.

Complexity and size of the Web site. Examples of complex Web sites are sites that make heavy use of scripting to generate Web pages or to provide functionality in them; employ multimedia content such as audio or video files; incorporate advanced technologies such as SMIL, SVG, or MathML; or are very large and difficult to maintain. In such cases, more specialized evaluation tools might be more useful, even though they may have other limitations.

Skills and knowledge of the Web developers. Some evaluation tools require users to have more knowledge of accessibility requirements or markup code (such as HTML, CSS, ...) than others. Also, some evaluation tools can support Web developers in learning such skills differently than others. It is important to identify the intended tool users and their requirements when selecting appropriate evaluation tools for a specific organization.

Pre-existing Web development environment. It is often beneficial to deploy evaluation tools that work well with the existing operating systems and other development infrastructure. Also, sometimes evaluation tools are plug-in extensions for Web authoring tools (such as editors, content management system, or save-as utilities) or browsers; or they can export evaluation reports in different formats (for example, export results to a database).

11.2.2 Usages of evaluation tools

Web accessibility evaluation tools can be used for different purposes depending on the expertise of the users and what checkpoints they want to evaluate. The following are some of the common characteristics of evaluation tools to support users fulfill different tasks during an evaluation process; some tools provide more than one mode of operation:

Generating reports. Report generating evaluation tools are usually designed to evaluate multiple pages or whole Web sites with little or no user interaction. The results of the accessibility checks that the tools execute are summarized in reports which can often be customized according to the needs of the users. Report generating tools are very useful in quickly determining the conformance of Web sites to checkpoints which can be evaluated automatically, and for identifying remaining checkpoints that need to be evaluated manually.

Step-by-step evaluations. *Wizard-based* evaluation tools guide users through sequences of checks step by step. Sometimes these tools are able to execute some of the accessibility checks automatically and prompt the users to manually evaluate the remaining checks. For example, an evaluation tool with a wizard interface may be able to automatically check if the images on a Web site have text descriptions, then display the images with their corresponding descriptions to the users to evaluate how appropriate these descriptions are.

In-page feedback. In-page feedback evaluation tools insert (temporary) icons and markup into the code of the Web pages to display the results of automated accessibility checks and their corresponding location within the pages. Sometimes, other types of icons are also inserted into the Web pages to assist the manual evaluation of checkpoints. For example, some tools may insert icons to indicate the hierarchy of the page headings or lists, or the reading sequence of table cells that are be perceived by some Web site users.

Page transformations. Transformation tools modify the appearance of the Web sites to help identify conceptual design issues with regard to Web accessibility. For example, transformation tools may present the content of Web sites in text only, without color, or read the content aloud. These types of evaluation tools are usually especially useful to compensate the limitations of automated accessibility checking and to support the users in evaluating checkpoints that need to be evaluated manually.

11.2.3 Features of evaluation tools

The following is a (non-exhaustive) collection of features which may help users to compare and assess Web accessibility evaluation tools for their specific needs. Some evaluation tools provide some of these features with varying adequacy so that careful analysis needs to be made when assessing them; sometimes the tool vendors can provide additional information about how their tools support these or other features.

Accessibility: *How accessible is the evaluation tool for people with disabilities?*

It is equally important to ensure that people with disabilities can effectively contribute to the Web, as it is for them to be able to effectively use the Web. Evaluation tool developers and vendors can provide accessibility in all parts of a tool (user interface, documentation, or generated reports) by following the Authoring Tool Accessibility Guidelines 1.0 [16].

Checkpoint coverage: *Which checkpoints is the evaluation tool able to adequately address?*

While most Web accessibility evaluation tools support a broad variety of accessibility checks, some tools focus on specific checkpoints that are usually not automatable or require more sophisticated methods to evaluate. Also, evaluation tools may have varying degrees of accuracy or support the evaluation of the same checkpoints in different ways (such as automatically or manually for example).

Configuration: *How well does the evaluation tool adapt to the requirements of the users?*

Accessibility checks. For specific Web sites, evaluation tool users may want to customize the built-in accessibility checks in order to achieve better performance. For example, tool users may want to suppress certain automated checks, or to modify the parameters which trigger dialogs and prompts that assist users with manual checks.

Reporting. Evaluation tools that generate reports sometimes provide capabilities to customize the format of these reports to different degrees. Customizing reports according to the roles of the developers (for example content author, programmer, project manager, etc.) is especially useful for larger development teams.

Web site coverage. In some cases, evaluation tools can be configured to examine entire groups of related pages (such as department sub-sites, pages required to fulfil a specific task on a Web site, etc.) rather than single Web pages. This feature may also be useful for Web site monitoring purposes.

Integration: *How well does the evaluation tool integrate into the Web development environment of the users?*

Platform support. Even though some evaluation tools may be available on more than one platform (hardware, operating system, and system configuration), they may sometimes not support the same features or perform equally on all platforms. It is important to ensure that the required tool features are supported on the platform where it will be deployed.

Software extension. Some evaluation tools integrate into existing development environments by providing plug-in interfaces for Web browsers or authoring tools (such as editors, content management systems, or save-as utilities). This feature may be important to some tool users even though such evaluation tools may sometimes be constrained by the application they are plugged into.

Data export. Some evaluation tools can export evaluation results to databases or other types of data processing tools such as analysis or reporting tools. Some of the commonly supported data formats which facilitate such data exchange are XML or Evaluation and Report Language (EARL).

Policy requirements: *Which guidelines and policy requirements does the evaluation tool support?*

Some evaluation tools provide support for several accessibility guidelines and national policy requirements. For organizations that are obliged to adhere to one or more national policy requirements, it is important that the selected evaluation tools adequately support these.

Reliability: *How reliable are the results delivered by the evaluation tool?*

Inaccurate results such as not detecting accessibility barriers or reporting non-existent barriers (that is, false positives or false negatives) decrease the reliability of the evaluation tool and hence decrease the efficiency of the evaluation. Currently there is no widely accepted single method to measure reliability so that careful

assessments of the performance of evaluation tools with respect to the specific type of Web site need to be made.

Repair: *How well does the evaluation tool assist developers in repairing inaccessible Web sites?*

Even though repair is not part of the evaluation process, it is often the next logical step. Evaluation tools can assist developers in repairing accessibility barriers and raising the overall quality of Web sites by providing in-line repair options, or by providing additional information for possible repair measures.

Web technology support: *How well does the evaluation tool support the relevant Web technologies?*

There several types of Web technologies (such as HTML, XHTML, CSS, etc.) and often several versions of each. Even though some of the more advanced Web technologies such as SMIL, SVG, or MathML are currently not widely supported by evaluation tools, it is important to select evaluation tools that best address the specific implementation of Web sites.

11.3 Strategies for Assessing the Accessibility of ECLAP Portals

This section focuses on the metrics related to the assessment and evaluation of the accessibility of the ECLAP Portals. It is complementary to DE1.1 "Assessment and Evaluation Manual" and its purpose is to maintain the accessibility quality under control via a specific set of metrics. The scope of this section is to identify the assessment criteria regarding the accessibility of the implemented portals and tools . The requirements on the content are directly linked to the user needs and expectations regarding the portal. More specifically, the aim has been to evaluate ease of access to information and ease of use of the search engine, and to identify problems arising during navigation on the site.

The ECLAP system is comprised of a back end and a front end tool. The back end tool provides the tools for ingesting and preparing the metadata to be presented on the ECLAP portal as well as making them available to Europeana. The front end tool incorporates the construction of the ECLAP portal's functionalities including transcoding and play-out of the digital content.

This is determined by the underlying technology and tools in the back-end (workflow for putting content online) of the system as well as the front-end (interaction with the portal) which is comprised of the user interface, content and functionalities. The focus and aim of the chosen methodology has therefore been on accessibility in regard to design, functionality, search, navigation, and playing of content. More specifically the available components that have to be assessed are:

- ECLAP Metadata Ingestion Servers;
- ECLAP Social Service Portal;
- ECLAP Mobile tools.

11.3.1 Methodology

The metrics proposed are based on the Web Content Accessibility. The Web Content Accessibility Guidelines (WCAG) 2.0 [21] covers a wide range of recommendations for making Web content more accessible. Following these guidelines will make content accessible to a wider range of people with disabilities, including blindness and low vision, deafness and hearing loss, learning disabilities, cognitive limitations, limited movement, speech disabilities, photosensitivity and combinations of these. Following these guidelines will also often make your Web content more usable to users in general.

The proposed metrics are organized around the following four principles, which lay the foundation necessary for anyone to access and use Web content. Anyone who wants to use the Web must have content that is:

- Perceivable - Information and user interface components must be presentable to users in ways they can perceive.
- Operable - User interface components and navigation must be operable.
- Understandable - Information and the operation of user interface must be understandable.
- Robust - Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.

The checkpoints related to the ECLAP portal accessibility are divided to three levels of conformance: A (lowest), AA, and AAA (highest). The checkpoints are the following:

A (lowest priority checkpoints)	Yes	No	N/A
Provide a text equivalent for every non-text element (e.g., via "alt", "longdesc", or in element content). *This includes*: images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ascii art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video.			
Ensure that all information conveyed with color is also available without color, for example from context or markup.			
Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions).			
Organize documents so they may be read without style sheets. For example, when an HTML document is rendered without associated style sheets, it must still be possible to read the document.			
Ensure that equivalents for dynamic content are updated when the dynamic content changes.			
Until user agents allow users to control flickering, avoid causing the screen to flicker.			
Use the clearest and simplest language appropriate for a site's content.			
And if you use images and image maps (Priority 1)	**Yes**	**No**	**N/A**
Provide redundant text links for each active region of a server-side image map.			
Provide client-side image maps instead of server-side image maps except where the regions cannot be defined with an available geometric shape.			
And if you use tables (Priority 1)	**Yes**	**No**	**N/A**
For data tables, identify row and column headers.			
For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells.			
And if you use frames (Priority 1)	**Yes**	**No**	**N/A**
Title each frame to facilitate frame identification and navigation.			
And if you use applets and scripts (Priority 1)	**Yes**	**No**	**N/A**
Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not possible, provide equivalent information on an alternative accessible page.			
And if you use multimedia (Priority 1)	**Yes**	**No**	**N/A**
Until user agents can automatically read aloud the text equivalent of a visual track, provide an auditory description of the important information of the visual track of a multimedia presentation.			
For any time-based multimedia presentation (e.g., a movie or animation), synchronize equivalent alternatives (e.g., captions or auditory descriptions of the visual track) with the presentation.			
And if all else fails (Priority 1)	**Yes**	**No**	**N/A**

If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page.			

Table 11-1 *Lowest Priority Checkpoints*

AA (medium priority checkpoints)	Yes	No	N/A
Ensure that foreground and background color combinations provide sufficient contrast when viewed by someone having color deficits or when viewed on a black and white screen. [Priority 2 for images, Priority 3 for text].			
When an appropriate markup language exists, use markup rather than images to convey information.			
Create documents that validate to published formal grammars.			
Use style sheets to control layout and presentation.			
Use relative rather than absolute units in markup language attribute values and style sheet property values.			
Use header elements to convey document structure and use them according to specification.			
Mark up lists and list items properly.			
Mark up quotations. Do not use quotation markup for formatting effects such as indentation.			
Ensure that dynamic content is accessible or provide an alternative presentation or page.			
Until user agents allow users to control blinking, avoid causing content to blink (i.e., change presentation at a regular rate, such as turning on and off).			
Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages.			
Until user agents provide the ability to stop auto-redirect, do not use markup to redirect pages automatically. Instead, configure the server to perform redirects.			
Until user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the user.			
Use W3C technologies when they are available and appropriate for a task and use the latest versions when supported.			
Avoid deprecated features of W3C technologies.			
Divide large blocks of information into more manageable groups where natural and appropriate.			
Clearly identify the target of each link.			
Provide metadata to add semantic information to pages and sites.			
Provide information about the general layout of a site (e.g., a site map or table of contents).			
Use navigation mechanisms in a consistent manner.			
And if you use tables (Priority 2)	Yes	No	N/A
Do not use tables for layout unless the table makes sense when linearized. Otherwise, if the table does not make sense, provide an alternative equivalent (which may be a linearized			

version).			
If a table is used for layout, do not use any structural markup for the purpose of visual formatting.			
And if you use frames (Priority 2)	**Yes**	**No**	**N/A**
Describe the purpose of frames and how frames relate to each other if it is not obvious by frame titles alone.			
And if you use forms (Priority 2)	**Yes**	**No**	**N/A**
Until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned.			
Associate labels explicitly with their controls.			
And if you use applets and scripts (Priority 2)	**Yes**	**No**	**N/A**
For scripts and applets, ensure that event handlers are input device-independent.			
Until user agents allow users to freeze moving content, avoid movement in pages.			
Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies [Priority 1 if functionality is important and not presented elsewhere, otherwise Priority 2.]			
Ensure that any element that has its own interface can be operated in a device-independent manner.			
For scripts, specify logical event handlers rather than device-dependent event handlers.			

Table 11-2 *Medium Priority Checkpoints*

In General (Priority 3)	**Yes**	**No**	**N/A**
Specify the expansion of each abbreviation or acronym in a document where it first occurs.			
Identify the primary natural language of a document.			
Create a logical tab order through links, form controls, and objects.			
Provide keyboard shortcuts to important links (including those in client-side image maps), form controls, and groups of form controls.			
Until user agents (including assistive technologies) render adjacent links distinctly, include non-link, printable characters (surrounded by spaces) between adjacent links.			
Provide information so that users may receive documents according to their preferences (e.g., language, content type, etc.)			
Provide navigation bars to highlight and give access to the navigation mechanism.			
Group related links, identify the group (for user agents), and, until user agents do so, provide a way to bypass the group.			
If search functions are provided, enable different types of searches for different skill levels and preferences.			
Place distinguishing information at the beginning of headings, paragraphs, lists, etc.			
Provide information about document collections (i.e., documents comprising multiple			

	Yes	No	N/A
pages.).			
Provide a means to skip over multi-line ASCII art.			
Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page.			
Create a style of presentation that is consistent across pages.			
And if you use images and image maps (Priority 3)	**Yes**	**No**	**N/A**
Until user agents render text equivalents for client-side image map links, provide redundant text links for each active region of a client-side image map.			
And if you use tables (Priority 3)	**Yes**	**No**	**N/A**
Provide summaries for tables.			
Provide abbreviations for header labels.			
Until user agents (including assistive technologies) render side-by-side text correctly, provide a linear text alternative (on the current page or some other) for *all* tables that lay out text in parallel, word-wrapped columns.			
And if you use forms (Priority 3)	**Yes**	**No**	**N/A**
Until user agents handle empty controls correctly, include default, place-holding characters in edit boxes and text areas.			

Table 11-3 *General*

11.3.2 Metadata Ingestion Servers

11.3.2.1 Overview

The "Metadata Ingestion Servers" are responsible for transferring the metadata & content in the content partners' archives to the ECLAP Social portal and how these are adapted to be accessible to users on the portal as well as via Europeana.

Content partners can provide content & metadata using a manual web upload, providing a content file together with the metadata entered via a web form, but in many cases they have a great number of content items to be provided and the manual approach can be used only in few cases. For this reason the basic workflow for massive content and metadata ingestion will be made of the following steps:

- Content partners will provide metadata using the ECLAP Metadata Ingestion Service (EMIS) portal, metadata should be provided as XML, it will be directly uploaded as a file or harvested them via a OAI-PMH access;
- Each content partners will map their own metadata XML structure to the ECLAP metadata XML format , this will be done using the EMIS portal to define a XSLT that will be used in the mapping phase;
- In case of a OAI-PMH access the EMIS will crawl the content partners archives and it acquires the original metadata;
- When the original metadata is acquired it is mapped to the ECLAP metadata format and stored and it will be available to the ECLAP Social Service Portal (ESSP) via OAI-PMH access;
- The ESSP regularly crawls the EMIS to acquire metadata represented using the ECLAP metadata format and in the original metadata format;

- For each metadata record acquired the ESSP will download the content files associated with it, ESP will use the content urls defined in the ECLAP metadata (content may be also acquired using Hard Disks, DVDs in case content size is too high for a rapid download);

- Each piece of content acquired will be formatted, adapted and a complete media content will be produced in the ESP portal with the associated metadata;

- Each produced content will be available only internally in the ESSP portal for:
  - metadata enrichment,
  - metadata translation,
  - IPR definition,
  - …

  These activities will be orchestrated using an internal workflow management tool integrated in the portal;

- When the content and the associated metadata will be ready they will be available (on the basis of the IPR defined) for access to the final users;

- Metadata will be published on Europeana and Europeana users will reach the ESSP portal to see the content.

11.3.2.2 *Assessment and Evaluation*

The assessment and evaluation of the "Metadata Ingestion Servers" is performed based on the following set of metrics. Each metric is presented along with an indicator to provide a different name to any single metric.

- *PLPCCMI:* Percentage of "lowest priority checkpoints" checkpoints (Table 11-1) to which the Social Service Portal is compliant.

- *PMPCCMI:* Percentage of "medium priority checkpoints" checkpoints (Table 11-2) to which the Social Service Portal is compliant.

- *PHPCCMI:* Percentage of "highest priority checkpoints" checkpoints (Table 11-3) to which the Social Service Portal is compliant.

- *PUCMIS1:* Percentage of the Use Cases described in WP2 and related to the "Metadata Ingestion Server", which can be carried out with images turned off.

- *PUCMIS2:* Percentage of the Use Cases described in WP2 and related to the "Metadata Ingestion Server", which can be carried out with images sound off.

- *PUCMIS3:* Percentage of the Use Cases described in WP2 and related to the "Metadata Ingestion Server", which can be carried out in the lowest screen resolution.

- *PUCMIS4:* Percentage of the Use Cases described in WP2 and related to the "Metadata Ingestion Server", which can be carried with the display color set to gray scale.

- *PUCMIS5:* Percentage of the Use Cases described in WP2 and related to the "Metadata Ingestion Server", which can be carried without using the mouse.

- *PUCMIS6:* Percentage of the Use Cases described in WP2 and related to the "Metadata Ingestion Server", which can be carried out in a voice browser.

- *PUCMIS7:* Percentage of the Use Cases described in WP2 and related to the "Metadata Ingestion Server", which can be carried out in a text browser.

In the following Table each metric is related to the unit of measure of the metric; the minimum, the typical and the optimum target value to be reached:

Indicator	Unit of Measure	Minimum Target Value	Typical Target Value	Optimum Target Value
PLPCCMI	Percentage			

PMPCCMI	Percentage			
PHPCCMI	Percentage			
PUCMIS1	Percentage			
PUCMIS2	Percentage			
PUCMIS3	Percentage			
PUCMIS4	Percentage			
PUCMIS5	Percentage			
PUCMIS6	Percentage			
PUCMIS7	Percentage			

Table 11-4 *Metadata Ingestion Servers Accessibility Metrics*

11.3.3 Social Service Portal

11.3.3.1 Overview

The "ECLAP Social Service Portal" is a socially enabled portal that is the main front end for the networked user to upload, enrich and work on content. It provides support for:

- Access to the content, make queries via PC and Mobile;
- Create communities and groups for the ECLAP Networking, discussions on content, on group topics, etc.;
- Augment content and metadata with additional information and free tagging;
- Create aggregated content for leisure and entertainment, such as compilation, collections, slide shows of images, etc.;
- Ingest and Upload content on ECLAP providing metadata and classifications;
- Define IPR issues on the content, with the support of a guiding IPR Wizard;
- User registration and networking: single user and groups;
- Develop social relationships with other ECLAP users, defining friends and getting and producing recommendations, clustering of content and users;
- Creation of discussion groups and for each group a forum of discussion on topics, a list of users, a mailing list, etc.;
- Provide multilingual capabilities interface, multilingual metadata in several languages such as: Danish, Polish, Slovenian, Greek, English, Italian, French, Dutch, Spanish, Hungarian, German, Portuguese, Catalan;
- Search and retrieval of content on the basis of semantic information associated with digital items (e.g., web2.0 technologies), taking and integrating solutions based on metadata, classifications, ontologies, dictionaries, synonymous, etc., Content will include digital resources, web pages of groups, comments on content, forum discussion, etc.;
- Play content: rendering visually content as video, audio, web pages, images, document, etc. Over multiple players and languages;

- Voting and ranking content; Also used for producing recommendations to users to see high quality content when they work on some specific enrichment activity;

- Comments and annotations of Content: The users will be entitled to leave comments and annotations on digital objects and content in general, to add semantic and knowledge to the collected content;

- Search for similar users; Also used for producing recommendations to users to identify similar Users that may be of help when they work on some specific enrichment activity;

- Search for similar objects (digital resources); also used for producing recommendations to users to see similar objects when they work on some specific enrichment activity.

The term "Mobile Tools" refers to the subset of "Social Service Portal Tools" responsible for remote access to the ECLAP portal via mobile devices for collecting and organizing ECLAP content on the mobile device.

11.3.3.2 Assessment and Evaluation

The assessment and evaluation of the "Social Service Portal Tools" is performed based on the following set of metrics. Each metric is presented along with an indicator to provide a different name to any single metric.

- *PLPCCSSP:* Percentage of "lowest priority checkpoints" checkpoints (Table 11-1) to which the Social Service Portal is compliant.

- *PMPCCSSP:* Percentage of "medium priority checkpoints" checkpoints (Table 11-2) to which the Social Service Portal is compliant.

- *PHPCCSSP:* Percentage of "highest priority checkpoints" checkpoints (Table 11-3) to which the Social Service Portal is compliant.

- *PUCSSPS1:* Percentage of the Use Cases described in WP2 and related to the "Social Service Portal", which can be carried out with images turned off.

- *PUCSSPS2:* Percentage of the Use Cases described in WP2 and related to the "Social Service Portal", which can be carried out with images sound off.

- *PUCSSPS3:* Percentage of the Use Cases described in WP2 and related to the "Social Service Portal", which can be carried out in the lowest screen resolution.

- *PUCSSPS4:* Percentage of the Use Cases described in WP2 and related to the "Social Service Portal", which can be carried with the display color set to gray scale.

- *PUCSSPS5:* Percentage of the Use Cases described in WP2 and related to the "Social Service Portal", which can be carried without using the mouse.

- *PUCSSPS6:* Percentage of the Use Cases described in WP2 and related to the "Social Service Portal", which can be carried out in a voice browser.

- *PUCSSPS7:* Percentage of the Use Cases described in WP2 and related to the "Social Service Portal", which can be carried out in a text browser.

In the following Table each metric is related to the unit of measure of the metric; the minimum, the typical and the optimum target value to be reached:

Indicator	Unit of Measure	Minimum Target Value	Typical Target Value	Optimum Target Value
PLPCCSSP	Percentage			
PMPCCSSP	Percentage			
PHPCCSSP	Percentage			
PUCSSPS1	Percentage			

PUCSSPS2	Percentage			
PUCSSPS3	Percentage			
PUCSSPS4	Percentage			
PUCSSPS5	Percentage			
PUCSSPS6	Percentage			
PUCSSPS7	Percentage			

Table 11-5 Social Service Portal Accessibility Metrics

11.3.4 Mobile Tools Portal

11.3.4.1 Overview

The "Mobile Tools Portal" is an implementation of the "ECLAP Social Service Portal" intended for use with mobile devices.

11.3.4.2 Assessment and Evaluation

The assessment and evaluation of the "Mobile Tools Portal" is performed based on the following set of metrics. Each metric is presented along with an indicator to provide a different name to any single metric.

- *PLPCCMTP:* Percentage of "lowest priority checkpoints" checkpoints (Table 11-1) to which the Mobile Tools Portal is compliant.
- *PMPCCMTP:* Percentage of "medium priority checkpoints" checkpoints (Table 11-2) to which the Mobile Tools Portal is compliant.
- *PHPCCMTP:* Percentage of "highest priority checkpoints" checkpoints (Table 11-3) to which the Mobile Tools Portal is compliant.
- *PUCMTPS1:* Percentage of the Use Cases described in WP2 and related to the "Mobile Tools Portal", which can be carried out with images turned off.
- *PUCMTPS2:* Percentage of the Use Cases described in WP2 and related to the "Mobile Tools Portal", which can be carried out with images sound off.
- *PUCMTPS3:* Percentage of the Use Cases described in WP2 and related to the "Mobile Tools Portal", which can be carried out in the lowest screen resolution.
- *PUCMTPS4:* Percentage of the Use Cases described in WP2 and related to the "Mobile Tools Portal", which can be carried with the display color set to gray scale.
- *PUCMTPS5:* Percentage of the Use Cases described in WP2 and related to the "Mobile Tools Portal", which can be carried without using the mouse.
- *PUCMTPS6:* Percentage of the Use Cases described in WP2 and related to the "Mobile Tools Portal", which can be carried out in a voice browser.
- *PUCMTPS7:* Percentage of the Use Cases described in WP2 and related to the "Mobile Tools Portal", which can be carried out in a text browser.

In the following Table each metric is related to the unit of measure of the metric; the minimum, the typical and the optimum target value to be reached:

Indicator	Unit of Measure	Minimum Target Value	Typical Target Value	Optimum Target Value

PLPCCMTP	Percentage			
PMPCCMTP	Percentage			
PHPCCMTP	Percentage			
PUCMTPS1	Percentage			
PUCMTPS2	Percentage			
PUCMTPS3	Percentage			
PUCMTPS4	Percentage			
PUCMTPS5	Percentage			
PUCMTPS6	Percentage			
PUCMTPS7	Percentage			

Table 11-6 Mobile tools Portal Accessibility Metrics

12 References

[1]   http://drupal.org/documentation/modules/locale

[2]   http://drupal.org/node/22272

[3]   W3C Web Accessibility Initiative (WAI) - http://www.w3.org/WAI/intro/wcag.php

[4]   http://www.w3.org/TR/WAG

[5]   http://www.w3.org/TR/ATAG

[6]   http://www.w3.org/TR/UAAG

[7]   http://www.w3.org/WAI/eval/users.html

[8]   http://validator.w3.org/

[9]   http://www.w3.org/People/Raggett/tidy/

[10]   http://jigsaw.w3.org/css-validator/

[11]   http://www.w3.org/Math/validator/

[12]   http://www.w3.org/TR/WCAG10/full-checklist.html

[13]   http://www.eastendliteracy.on.ca/ClearLanguageAndDesign/readingeffectivenesstool/

[14]   Web Accessibility Guidelines - http://www.w3.org/WAI/intro/wcag.php

[15]   http://www.uiaccess.com/accessucd/screening.html

[16]   http://www.w3.org/TR/ATAG10/

[17]   http://www-03.ibm.com/able/resources/userevaluations.html

[18]   http://www.alistapart.com/articles/usability-testing-demystified/

[19]   http://www.sensible.com/rocketsurgery/index.html

[20]   http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470185481,descCd-DOWNLOAD.html

[21]   http://www.w3.org/TR/WCAG20