EHRI-1_D19.3_Multilingual Search Interface

Joseba Kepa, Mike Priddy

European Holocaust Research Infrastructure
Theme [INFRA-2010-1.1.4]
GA no. 261873

Deliverable D19.3

Multilingual Search Interface

Kepa Joseba Rodriguez
Georg-August-Universitaet Göttingen Stiftung Oeffentlichen Rechts
Mike Priddy
Data Archiving and Networked Services, Koninklijke Nederlandse Akademie van Wetenschappen

Start: M15
Due: M30
Actual: M46

Note: The official starting date of EHRI is 1 October 2010. The Grant Agreement was signed on 17 March 2011. This means a delay of 6 months, which will be reflected in the submission dates of the deliverables.

Document Information

Project URL	www.ehri-project.eu
Document URL	n/a
Deliverable	19.3 Multilingual Search Interface
Work Package	19
Lead Beneficiary	1 NIOD-KNAW
Relevant Milestones	MS3
Nature	R
Type of Activity	RTD
Dissemination level	PU
Contact Person	Mike Priddy mike.priddy@dans.knaw.nl
Abstract (for dissemination)	This report describes the proposed implementation of the multilingual search interface of the EHRI portal. The main topics presented and discussed in the document are the use of the EHRI thesaurus and the geographical database GeoNames for multilingual and semantic query expansion, the indexing of the metadata registry with Solr and the metrics, datasets and results of the evaluation of the system.
Management Summary (required if the deliverable exceeds more than 25 pages)	

Table of Contents

1 Introduction

In the context of political, cultural and economic integration of European countries an important trend is the development of transnational data infrastructures for library, archival, administrative, heritage and scientific data[1]. The creation and use of these infrastructures involves the management of data in different languages, thus necessitating the implementation of multilingual information retrieval applications.

The management of retrieval of multilingual resources is an important challenge for the EHRI project, since EHRI integrates metadata from different institutions of different countries. These metadata records are written in different languages, and sometimes one can find single records written in more than one language, making multilingual information retrieval necessary to present to the user relevant and complete responses to their query.

In a European context several different approaches for multilingual information retrieval have been followed, such as automatic or manual translation of metadata, dictionary mapping, statistical similarity, etc. Most of these approaches have been evaluated by the European project CACAO (Cross-language Access to Catalogues And On-line libraries)[2].

The CACAO project was a EU funded initiative that ran between 2007 and 2009, with the aim to build a multilingual search infrastructure mainly targeting digital libraries. The project employed a number of technologies to aid query expansion, such as bilingual dictionaries, semantic disambiguation, multiple thesauri, WordNet similarity (semantic similarity between user query and concepts of WordNet - Pedersen and Patwardhan, 2004) and statistical natural language processing (CACAO 2009). The approach of CACAO has been adopted by European infrastructures, such as The European Library[3].

Another European project is the German BASE (Bielefeld Academic Search Engine)[4] search engine for academic open content resources. BASE uses the Eurovoc[5] thesaurus (Pieper 2008) of the European Commission to expand the queries into the 22 official languages of the European Union[6].

The methods implemented in the EHRI project are partially based on the approach used at the BASE project. Like the BASE project, EHRI uses a multilingual thesaurus for term expansion[7]. There are some differences between both projects. While BASE uses a broad purpose vocabulary, EHRI uses self-developed vocabularies with holocaust relevant knowledge. A further difference between both projects is that BASE expands only user queries. At the EHRI project we implement two optional kinds of term expansion, expansion of terms found in the user query, or expansion of terms found in the data when the data is being indexed.

---

1 Examples of well-known European infrastructures include: Europeana, The European Library, and the Archives Portal Europe.

2 http://www.cacaoproject.eu/deliverables/CACAO_presentation.pdf

3 http://www.theeuropeanlibrary.org

4 http://www.base-search.net/about/en/index.php

5 Thesaurus maintained by the Publications Office of the European Union with translations of the terms to 22 languages. Eurovoc is used by the European institutions. http://eurovoc.europa.eu/

6 Official & working languages at the time of the project.

7 The use of the thesaurus for search and retrieval purposes was proposed in the Description of Work.

This document describes a proposal for the implementation of the search interface with a special focus on the use of the controlled vocabularies. However the actual implementation may differ due to alterations in the operational and functional requirements.

In Chapter 2 we describe the controlled vocabularies used in EHRI, the EHRI thesaurus, and GeoNames[8], which is a database of geographic names accessible through web services. We use GeoNames to extend our locations authority lists to more languages.

Chapter 3 gives a short description of the index helper, an application that transform data from the EHRI's metadata registry to a format that is used to feed the search index.

Chapter 4 describes the implementation of multilingual term expansion in the search engine and presents two kinds of expansion types: query-time expansion (or expansion of terms of the user query), and index-time expansion (or expansion of terms found in the data during the indexing process).

In Chapter 5 we describe the term expansion service used for query-time semantic expansion. This service uses the hierarchy of the EHRI thesaurus terms.

Chapter 6 presents the evaluation of the multilingual expansion.

Finally, Chapter 7 presents the conclusions of this report.

At the end of the document there is a glossary of technical terms used in this document.

[8] http://www.geonames.org

2 Knowledge used for query expansion and indexing

The EHRI Thesaurus, created by Work Package 18, consists of a hierarchical list of terms, which will be in a total of ten languages and a set of lists of authorities. The Thesaurus has been described in detail in two documents, Deliverable 18.2 (Borut *et a*l., 2012) and Deliverable 18.3 (Gertner *et al.*, 2013). An overview of the terms and of the authority files used for the development of the multilingual search interface is given in sections 2.1 and 2.2**Erreur ! Source du renvoi introuvable.**.

Work Packages 18 and 19 worked together in the integration of the different languages of the thesaurus terms into a unique SKOS[9] representation. SKOS was chosen because is a standard format useful to integrate multilingual information and represent the relationships between concepts of the vocabularies. A further goal of the use of SKOS was to contribute, a knowledge resource in a standard format, to the research community in the form of reusable Linked Open Data. The SKOS representation of the terms and authority files has been imported into the Metadata Registry[10] and used for the multilingual search components.

Holocaust research involves the use of geographic information. This information is used in relation to persons (birth place, dead place, where they live), events like transports or deportations, detention facilities, etc. Places of names in holocaust relevant documentation are highly multilingual, not only because the Holocaust happened in a broad geographic area, but also because in the same area documentation was both produced by local institutions and occupation authorities, and therefore, post-Holocaust the document may have been stored and catalogued in different locations. That motivated us to extend the language coverage of the place authority lists developed in EHRI using an external source of knowledge, GeoNames[11].

GeoNames is a geographical database in which geographic terms are being translated into an increasing list of languages. It has a very complete coverage for the languages used in the EHRI project in the relevant geographic areas. The use of GeoNames to extend the language coverage of the place authority lists is discussed in section 2.4**Erreur ! Source du renvoi introuvable.**.

2.1 The EHRI thesaurus terms

The EHRI thesaurus is a hierarchically organized list of terms linked by different relationships. It aims to build a useful framework to represent information about the Holocaust.

The first version of the thesaurus was developed with exclusively English labels for terms. A further nine versions are based upon a translation of the English preferred labels[12] to the goal language. For each thesaurus term there is a preferred label in most of the languages[13].

The thesaurus contains 878 terms with their realisation in 10 different languages[14]. It contains 187 alternative labels for English and a different amount of alternative labels for

[9] http://www.w3.org/2004/02/skos/intro

[10] Which uses the graph database Neo4J as its main data store. Data is stored in a graph, which can model easily the concept structure and relationships represented in SKOS.

[11] http://www.geonames.org/about.html

[12] Here we use the SKOS definition of the labels as preferred, alternative and hidden. The actual version of the thesaurus doesn't contain any hidden labels.

[13] The German, French and Russian translations of the thesaurus provide more than one preferred labels for some terms, one for male and the other one for female in most of the cases.

other languages. Each term can have zero or more narrower terms, and multiple broader terms as well.

Terms are linked by three relationships:

1. Broader (BT): 1228 relationships
2. Narrower (NT): 1228 relationships
3. Related (RT): 352 relationships

2.2 Authority lists

We use two of the authority lists for the multilingual search component, the list of ghettos and the list of concentration camps:

1. Ghettos: The list of ghettos is based on the online Hebrew edition[15] from Yad Vashem (YV) (Miron and Shulhani, 2009). The list contains 1113 places.
2. Camps: The authority list of camps is based on a preliminary list of camps provided by the International Tracing Service (ITS). The list contains 2056 camps, several of them linked by 985 broader/narrower relationships.

The authority lists are described in detail in (Borut *et al.*, 2012).

2.3 SKOSification of the Thesaurus

Thesaurus terms and authority lists have been converted into a SKOS representation of the vocabularies. One of the goals of this work was to contribute with knowledge resource in a standard format to the research community in form of reusable Linked Open Data. However, the main aim was the use of the SKOS representation as a useful framework to integrate multilingual information, information and made it useful for multilingual and semantic term expansion as described in chapters 4 and 5.

2.3.1 Terms

The SKOSification[16] workflow begins with the export of the thesaurus from TemaTres[17] using the SKOS-core exporter, which provides a non-validated SKOS-like RDF[18] representation of the vocabulary.

The provided translation of the preferred labels and alternative labels have been imported and transformed to the RDF representation of the vocabulary. Finally the multilingual thesaurus has been converted into standard SKOS[19] using Skosify[20], an open source tool, which accepts RDF as input and conforms it to the SKOS standards.

[14] The thesaurus is being translated to the following languages: German, French, Russian, Ukrainian, Polish, Dutch, Czech, Greek, Hungarian and Serbo-Croatian

[15] http://www.yadvashem.org/yv/he/research/ghettos_encyclopedia/

[16] The term SKOSification implies the process of conversion or transformation of a terminology into SKOS (Athena, 2010).

[17] http://www.vocabularyserver.com/

[18] http://www.w3.org/RDF/

[19] http://www.w3.org/2004/02/skos/

[20] http://www.w3.org/2001/sw/wiki/Skosify

The import of the translated labels into a SKOS file is not a trivial process. SKOS allows only one preferred label in each language for each concept[21]. In languages like English this is not problematic, since terms can be expressed with a unique label in the plural form. In languages, which do not follow the English form, thesauri terms are represented with a number of labels in the singular form (ISO 2788[22], ISO 5964[23]). For instance for the English preferred label “Disabled” the thesaurus provides the preferred labels for German “Behinderter Mann”[24], “Behinderte Frau”[25] and “Behindertes Kind”[26].

This multiplicity of preferred labels in several languages makes it difficult to import several of the translations of the thesaurus into a SKOS representation. For the integration of the languages with more than one preferred label in the SKOS representation we created an extension of SKOS that includes the following gender specific preferred labels, which can occur simultaneously as one can see in Figure 1.

```
  <rdf:Description rdf:about="http://data.ehri-
project.eu/vocab#term584">
    <dct:created>2012-08-16 08:13:51</dct:created>
    <skos:prefLabel xml:lang="en">Emigrants</skos:prefLabel>
    <skos-ehri:prefFemaleLabel xml:lang="de">Emigrantin</skos-
ehri:prefFemaleLabel>
    <skos-ehri:prefMaleLabel xml:lang="de">Emigrant</skos-
ehri:prefMaleLabel>
    <skos:altLabel xml:lang="uk-Latn">Emìgrant</skos:altLabel>
    <skos:prefLabel xml:lang="uk-Cyrl">Емігранти</skos:prefLabel>
    <skos:altLabel xml:lang="uk-Cyrl">Емігрантка</skos:altLabel>
    <skos:altLabel xml:lang="hu">Emigránsok</skos:altLabel>
    <skos:inScheme rdf:resource="http://data.ehri-
project.eu/vocab#"/>
    <rdf:type
rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
    <skos:altLabel xml:lang="uk-Latn">Emìgrantka</skos:altLabel>
    <skos:prefLabel xml:lang="ru-Latn">èmigranty</skos:prefLabel>
    <skos:altLabel xml:lang="uk-Cyrl">Емігрант</skos:altLabel>
    <skos:prefLabel xml:lang="ru-Cyrl">эмигранты</skos:prefLabel>
    <skos:prefLabel xml:lang="nl">Emigranten</skos:prefLabel>
    <skos:prefLabel xml:lang="pl">emigranci</skos:prefLabel>
    <skos:prefLabel xml:lang="uk-Latn">Emìgranti</skos:prefLabel>
    <skos:prefLabel xml:lang="hu">Kivándorlók</skos:prefLabel>
    <skos:broader rdf:resource="http://data.ehri-
project.eu/vocab#term23"/>
    <skos:broader rdf:resource="http://data.ehri-
project.eu/vocab#term66"/>
  </rdf:Description>
```

Figure 1: Thesaurus term entry with gender specific preferred label

21 http://www.w3.org/TR/skos-reference/#L2831

22 "Guidelines for the establishment and development of monolingual thesauri". International Standards Organization, 1986

23 “Guidelines for the establishment and development of multilingual thesauri”. International Standards Organization, 1985

24 Disabled man

25 Disabled woman

26 Disabled child

2.3.2 Authority lists

Authority lists for Ghettos and Camps were created as Excel spreadsheets and converted into RDF format using the tool LODRefine[27]. As with the thesaurus terms, the resulting RDF file was converted into standard SKOS using Skosify.

2.4 Multilingual extension of the authority files

GeoNames[28] is a geographical database with a broad coverage of all countries. It contains around 10 million geographic names corresponding to over 8 million unique features, as for instance populated places, historic places, etc. Each feature has the name in different languages, alternative names, geospatial information, geopolitical information and postal code. The data of GeoNames is accessible though different Web services using available client libraries[29].

Each GeoNames feature is represented by a persistent URI, which provides access to a HTML wiki page or to a RDF description of the feature. The EHRI expansion service uses the RDF description to extract multilingual information. The Java Web service of GeoNames has been used to extend the language coverage of the authority lists, accessing it with a command line application[30].

In Figure 2 the authority entry of the ghetto Debrecen is presented as an example. In this authority entry the translation of the name of the Ghetto into Hebrew as RDF and SKOS labels, and geospatial information can be seen.

```
<rdf:Description rdf:about="http://data.ehri-
project.eu/ghettos#ehri-gh-216">
    <rdf:type
rdf:resource="http://www.w3.org/2003/01/geo/wgs84_pos#Point"/>
    <rdf:type
rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
    <foaf:isPrimaryTopicOf
rdf:resource="http://www.yadvashem.org/yv/he/research/ghettos_encycl
opedia/ghetto_details.asp?cid=216"/>
    <skos:notation>216</skos:notation>
    <rdfs:label xml:lang="en">Debrecen</rdfs:label>
    <rdfs:label xml:lang="he-Hebr">דברצן</rdfs:label>
    <skos:prefLabel xml:lang="en">Debrecen</skos:prefLabel>
    <skos:prefLabel xml:lang="he-Hebr">דברצן</skos:prefLabel>
    <geo:lat>47</geo:lat>
    <geo:long>21</geo:long>
    <skos:inScheme rdf:resource="http://data.ehri-
project.eu/ghettos#"/>
</rdf:Description>
```

Figure 2: Authority entry for Debrecen

[27] LODrefine is a based on Google Refine/Open Refine, which is a standalone desktop application for cleaning, transformation and parsing massive data sets. LODRefine offers extensions packages to application for linking data to linked open data sources, such as Dbpedia. http://code.zemanta.com/sparkica/.

[28] http://www.GeoNames.org

[29] An exhaustive list of client libraries can be found here: http://www.GeoNames.org/export/client-libraries.html

[30] https://github.com/KepaJRodriguez/geotranslation

The SKOS labels were extracted and expanded using the command line application mentioned above. A new vocabulary is produced that will be used by the search engine. The new expanded entry for the ghetto Debrence is presented in Figure 3, and includes translations of Debrecen to four other languages: Russian, Polish, Greek and Ukrainian. Since the expanded version of the authority list will be used only for multilingual term expansion purposes, all unnecessary tags, such as geo-coordinates or link to the Yad Vashem database of ghettos have been removed.

```
  <rdf:Description rdf:about="http://ehri01.dans.knaw.nl/ehri-gh-
216">
    <skos:topConceptOf
rdf:resource="http://ehri01.dans.knaw.nl/conceptscheme"/>
    <rdf:type
rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
    <skos:prefLabel xml:lang="ru">Дебрецен</skos:prefLabel>
    <skos:inScheme
rdf:resource="http://ehri01.dans.knaw.nl/conceptscheme"/>
    <skos:prefLabel xml:lang="en">Debrecen</skos:prefLabel>
    <skos:prefLabel xml:lang="pl">Debreczyn</skos:prefLabel>
    <skos:prefLabel xml:lang="el">Ντέμπρετσεν</skos:prefLabel>
    <skos:prefLabel xml:lang="uk">Дебрецен</skos:prefLabel>
    <skos:prefLabel xml:lang="he">דברצן</skos:prefLabel>
  </rdf:Description>
```

Figure 3: Expanded entry for Debrecen for the search engine

3 Index helper

The indexing and search service is based upon (the Lucene[31] based) popular indexing and search server Solr[32].

Solr provides a very flexible import mechanism that supports configuration-based data extraction from sources in different formats, and a set of RESTful[33] interfaces that makes it accessible for import, update of the indexes, and programmatic querying. Solr provides specific features for faceted search, returning facet-counts for the number of documents in the index that match given facets. This provides a breakdown of the search results based upon some simple criteria, like language of the material, institution, country or access points.

The configuration of Solr is made using a single XML “schema” file, which provides the means to define the field data types and name fields corresponding to a specific “document model”. Facetable fields are also specified in this “schema.xml”.

The index helper consists of a web application with a RESTfull API developed as part of the work of WP19. This application mediates between EHRI graph database and the Solr index, providing behaviours for either per-item updates and deletions, or batch operations.

The index helper transforms data from EHRI's internal data serialization format (which resembles a sub-graph consisting of an entity and other entities related to it in the form of nodes and edges) to a flat format that is used to feed the search index. It does this by matching JSON[34] paths[35] (potentially within multiple nested objects) in the source data, and mapping them to a single key/value pair in the output JSON. There are three main types of paths, handled in different ways:

- Paths that map to specific, known fields in the search index schema.
- Paths that serve to match all internal keys not handled specifically by dynamic fields in the output schema.
- Paths that have additional specific handling, such as transforming from ISO 3166 country codes to indexable text that responds to search queries.

An important characteristic of the index helper is that it operates on data streams. Since the EHRI web service provides JSON data as a streaming HTTP response, and the search index (Solr) is capable of operating on a streaming request body, it is possible to transform large amounts of data with a fairly constant memory footprint.

31 http://lucene.apache.org/

32 http://lucene.apache.org/solr/

33 http://en.wikipedia.org/wiki/Representational_state_transfer

34 http://www.json.org

35 A JSON path is the JSON equivalent of Xpath for XML, in that it provides a notation for specifying a particular node or attribute in a hierarchical document structure.

4 Multilingual indexing and querying

One of the main issues to be addressed in the implementation of the search engine is the multilinguality of the EHRI data. Metadata records are written in different languages, and in some cases different languages are used inside of a metadata record as one can see in Figure 4[36], where two different languages are present in the description (Dutch and German).

Figure 4: Example of multilingual metadata record

To allow retrieval of metadata, regardless of which language the descriptions are written in, EHRI implements solutions based on multilingual controlled vocabularies, which help to address the multilingual character of the metadata in the EHRI registry.

Indexing of multilingual data involves detection of language in which data is written and the application of language specific analysis tools, which is explained in section 4.1. Section 4.2 presents our extension of the Lucene-skos tool, a Solr/Lucene module used for term expansion.

Terms can also be extracted from the user query and expanded to several languages, and this approach is presented in section 4.3. The alternative to the query time term expansion is the extraction of terms from the data and expansion during the indexing process, as proposed in section 4.4.

4.1 Indexing of multilingual data

Efficient indexing involves the use of language specific analysis components like stemmers, lemmatizers, spellchecker and correctors, or stop-word lists. To take advantage of the language specific analysers the first step consists of identifying and tagging the language in which each metadata field is written. Language identification is made using the open source language detection module langdetect[37]. After the language of each field has been detected, a suffix with the language tag is added and language specific analysers are applied.

[36] Source: NIOD; http://www.beeldbankwo2.nl

[37] http://code.google.com/p/language-detection/

Several languages use diacritical marks and the use of diacritical marks is sometimes inconsistent, especially in user queries[38]. To solve this problem we remove all diacritic marks in querying and indexing with Unicode[39] normalisation[40] libraries.

4.2 EHRI extension of Lucene-skos

Lucene-skos[41], which is a SKOS analyzer module for Solr/Lucene, is utilised to support index-time and query-time expansion of a thesaurus term found in the indexed data or in the query, to all preferred, alternative, and hidden SKOS labels of a concept. Lucene-skos does not support any transitive relationship, and terms can be semantically expanded only to the immediate broader or narrower term. The tool is able to expand terms using the labels described in the SKOS-core. Modifications have been made to the application to be compatible with the EHRI extensions to the SKOS[42] schema.

4.3 Query-time multilingual expansion

An alternative is to perform only a multilingual query-time expansion. In this case the query will be expanded to all preferred and alternative terms in all the languages. This approach has been used by the BASE[43] project.

The chief advantage of the query-time expansion is the flexibility of the approach. First of all, the system does not need to re-index all of the metadata in the registry after each change to the control vocabularies. If vocabularies are updated at a restart of Solr it is enough to use the new vocabulary for query-time expansion. Secondly, the time needed to index data is not affected by this method.

4.4 Index-time multilingual expansion

In this approach the indexing service will create big indexes for the metadata, which might require a long time to perform. The metadata in the registry must be re-indexed after each update or modification to the EHRI thesaurus. However, updates are not expected to be frequent after all the translations of the thesaurus have been included.

Index-time expansion is compatible with expansion to a high number of terms. It allows for the combining of multilingual expansion with query-time semantic expansion as explained in section 5. Another advantage of this approach is that Solr expands with increased efficiency multi-word terms at the time of indexing.

38 http://wiki.apache.org/solr/LanguageAnalysis#Ignoring_Diacritics
39 http://www.unicode.org
40 http://site.icu-project.org
41 https://github.com/behas/lucene-skos
42 https://github.com/KepaJRodriguez/lucene-skos-ehri
43 http://www.ub.uni-bielefeld.de/~befehl/base/solr/eurovoc.html

5 Term expansion service

The ERHI thesaurus terms have a hierarchical structure in which terms in a lower position in the hierarchy describe more specific concepts. Although each thesaurus term can be used to query the portal and to index documents, studies in other areas, such as in the biomedical domain (Hersch et al. 2000), show that documentation specialists use the narrowest index terms to index document collections.

Semantic expansion of terms to synonyms and more specific terms is a partial solution implemented in different scenarios (Bhogal et al. 2007) that can help to bridge the vocabulary gap between user query and indexed data.

The search engine implements a term expansion service for user queries based on the terms of the EHRI thesaurus. This section describes the service for thesaurus based semantic term expansion for the search interface. After a short general introduction of the software, the methods used to interact with the EHRI thesaurus and expand the terms are described.

These methods will be used to find shingles in multiword queries as presented in 5.2. After each shingle is expanded individually the results are integrated as presented in section 5.3.

5.1 General description of the service

The expansion service is implemented as a Web application in Java using the Spring Framework[44].

The search engine accesses the EHRI thesaurus as a web service using the ReST-API of the Metadata Registry. The default representation language for queries and responses of the API is JSON[45], and other formats, as string or XML, are being implemented.

The registry database is queried using the Cypher Query Language[46], a declarative graph query language that allows for querying and updating the graph database. Cypher Query Language provides similar operators to SQL[47] and a similar pattern matching function that is found with SPARQL[48], that are used to implement regular expressions in our queries. Cypher queries are submitted to the registry database as a JSON map by a POST method.

The service is able to perform semantic expansion to preferred labels of narrower terms and broader terms. Interaction with the thesaurus is implemented via the following methods:

`inThesaurus`
Given a string the method checks whether it is a label of a term of the thesaurus.

Argument	String
Return	Boolean

Cypher query:

```
START n=node(*)
```

44 http://www.springsource.org
45 http://www.json.org
46 http://docs.neo4j.org/chunked/stable/cypher-query-lang.html
47 http://en.wikipedia.org/wiki/SQL
48 http://www.w3.org/TR/rdf-sparql-query

```
WHERE has(n.prefLabel)
AND (n.prefLabel =~ "input")
OR (has(n.altLabel) AND ANY(x in n.altLabel WHERE x =~ "input"))
RETURN n
```

getConceptNode
Given a string that corresponds to a label of the thesaurus, the method browses the graph to the node representing the concept / term.

Argument	String
Return	Node ID

Cypher query:
```
START n=node(*)
MATCH n -[:describes]-> n1
WHERE has(n.prefLabel)
AND (n.prefLabel =~ "input")
OR (has(n.altLabel) AND ANY(x in n.altLabel WHERE x =~ "input"))
RETURN n1
```

searchNarrower
It takes a node ID as argument and gets all the children nodes.

Argument	Node ID (as String)
Return	List of Node IDs

Cypher query:
```
START n=node("nodeID")
MATCH n -[:narrower]-> n1
RETURN n1
```

searchBroader
This method takes as an argument a node ID and returns the parent nodes in the hierarchy.

Argument	Node ID (as String)
Return	List of Node IDs

Cypher query:
```
START n=node("nodeID")
MATCH n1 -[:narrower]-> n
RETURN n1
```

getLabeledNodes
It takes as an argument a concept node and returns the nodes linked by a **DESCRIBE** relationship. Each of the returned nodes has as a property the preferred label for a language and an array with all its alternative labels.

Argument	Node ID (as String)
Return	String (JSONObject-like)

Cypher query:
```
START n=node("nodeID")
MATCH n <-[:describes]- n1
```

```
RETURN n1
```

`extractTerms`
The argument is the JSON-like string returned by the getLabeledNodes method. It returns a list of strings containing the preferred label and all alternative labels for a language.

Argument	String (JSONObject-like)
Return	List of Strings

`expandWithThesaurus`
It takes the node ID of a concept node and returns an expanded list of terms with all the preferred and alternative labels of the concept nodes at a 'lower position' in the hierarchy.

Argument	Node ID (String)
Return	List of Node IDs (List of Strings)

`expandWithThesaurusBroader`
The argument of the method is the node ID of a concept node. It returns an expanded list of terms with all the preferred and alternative labels of the concept nodes at a 'higher position' in the hierarchy.

Argument	Node ID (String)
Return	List of Node IDs (List of Strings)

5.2 Detection of shingles

If the user query consists of more than one word, the expansion engine has to find the groups of words that appear together corresponding to a search term. For instance, if the user enters the query *"France welfare institutions"* the system should be able to expand "*Welfare Institutions"* as a term and *"France"* as a different term, and avoid considering "*Welfare*", "*Institutions*" or "*Institutions France*" as search terms.

The detection of shingles is made using n-grams[49] of words as follows:

1. Length of the first n-gram is the length of the query. In the given example the length of the query is 3, and then we begin using a 3-gram.
2. The system searches for the 3-gram in the thesaurus terms. If it is found, the query is expanded and results stored in a list. In the given example the string doesn't correspond with any term in the EHRI thesaurus. Then the size of the n-gram is reduced to n-1.
3. n = n-1. The size of the n-gram is now 2.
4. The system searches in thesaurus to identify if the 2-grams each correspond to a term. The 2-grams are *"France welfare"* and *"welfare institutions".* "*Welfare institutions*" is a label of a term in the EHRI thesaurus and it is included in the list of terms to expand.
5. "*welfare institutions*" is removed from the input list in order to avoid over-generation of results.

[49] A n-gram is a contiguous sequence of n items from a given sequence of text.

6. If there are words that are not in the thesaurus (as France in the given example), they will be used for free text search.

5.3 Expansion of the shingles and individual words

Once the shingles have been detected we expand them using the methods described in section 5.1.

Finally the results of the expansion of the different shingles and individual words are integrated in a JSON object as in the following example[50]:

Query: "France welfare institutions",

```
{
"q": "France welfare institutions",
"result":["Welfare institutions", "Orphanages", "Children's homes",
"Old people's home", "France"]
}
```

5.4 Index time multilingual expansion and query time semantic expansion

Query-time semantic expansion can be combined only with the index-time multilingual expansion presented in section 4.4 because in this case the size of the query would increase too much to be handle by Solr.

For instance, if the term “Anti-Semitism” is expanded, 51 narrower terms are returned (Figure 5). If the system uses the query time multilingual expansion and expands each of them to all preferred and alternative terms in all (initial) languages, then a query of more than 500 terms is produced, which would lead Solr to crash, or at least will be very inefficient in the search.

[50] In this example the semantic expansion of “Welfare institutions” is shown. The term France has not been found in the EHRI thesaurus terms and it will be sent to free text search.

```
{
"q":"antisemitism"
"result":[ "Anti-Jewish economic measures", "Arierparagraph",
"Antisemitic organisations",  "Dismissal from work", "Anti-Jewish
legislation", "Antisemitic press", "Anti-Jewish orders and decrees",
"Violent attacks on Jews", "Propaganda - antisemitic", "Antisemitic
propaganda", "The Jewish Question", "Exclusion of Jews from the
military", "Endloesung der Judenfrage", "Blood accusation",
"Desecration of graves", "Riots", "Final Solution of the Jewish
Question", "Antisemitic speeches, threats and declarations", "Anti-
Jewish signs", "Shearing of beards", ,"Aryanization", "Religious
antisemitism", "Desecration of Jewish religious  texts and objects",
"Aryanisation", "Anti Judaism", "Antisemitic activities", "Anti-
Jewish boycotts", "Round ups (razzias, Aktionen)", "Desecration of
cemeteries", "Judenberater", "Marking of Jews"," Yellow badge",
"Arrests", "Murder of Jews", "Judenreferat", "Pogrom",
"antisemitism", "Humiliation of Jews", "Judenkartei", "Badge of
shame", "Religious discrimination", "Grave desecration",
"Discrimination of Jews" ,"Pogroms and riots", "Antisemitic
organizations", "Armband", "Treuhandstellen and
Treuhandgesellschaften", "Anti-Jewish measures", "Treuhaender",
,"Antisemitism", "Blood libel"]
}
```


Figure 5: Semantic expansion of the term "Antisemitism"

The size of the semantic query expansion is constrained, expanding only to preferred terms in English, then translated into other languages, and alternative terms are added in the index-time, The size of the query can be further reduced by adding the expansion to parent preferred and alternative labels in the index-time removing, in this way, the terms in the lowest level of the hierarchy from the expanded query.

6 Evaluation

Evaluation methods of search engines with controlled datasets, such as archives and digital libraries[51], are very dependent of the expectations and requirements of the user to the response of the search engine.

Often two assumptions are relied upon: the first is that the user is searching for one item from the library or small number of items, the second assumption is that the attention of the user focuses on the first objects of the search (Pera and Ng, 2010) and users often do not look for results beyond the second page of the search results. Metrics used to evaluate the performance of these systems includes analysis of the relevance of the answer, computing only precision of the first results[52] presented in the response and the accuracy of the ranking of results[53]. Further methods include analysis of logs to infer the behaviour of the user, such as clicks on the first items after the search or repetition of the search changing some parameters.

The necessities of EHRI users are different than those of the users of libraries. Scholarly researchers are mostly interested in finding new sources, which in a relevance-only based search engine would likely be ranked in a low position many pages in on a search result – the long tail. After the researcher discovers the sources, he/she will be able to establish relationships to other sources and material, creating new knowledge.

In this chapter the evaluation datasets (Section 6.1), a set of metrics that is planned for use for the first evaluation of the search engine (Section 6.3), and the results of the evaluation (Section 6.4) are presented.

6.1 Datasets and task

For the evaluation two datasets, a set of 28,030 documentary units indexed with Solr and a set of 48 queries are used.

Documentary units are written in English, Russian and Ukrainian. Most records are monolingual, but there are several documentary units written in both English and one of the Cyrillic languages.

Queries are 48 search terms in English, Russian and Ukrainian. The task consists of retrieving all possible correct responses for each query, regardless the language of the query or of the metadata record. Queries were sent to Solr in 3 different scenarios: without use of the multilingual expansion, query time expansion and index time expansion.

6.2 Time needed for indexing and querying

In the tests the indexing of the data for no-expansion and query-time expansion takes 1.5 minutes, and for index-time expansion almost 5 hours to complete.

[51] The evaluation of free text search engines, available on the Internet, follow different methodologies whose discussion is beyond the scope of this document.

[52] The most used metric for that is Mean Average Precision (Croft et al. 2010)

[53] Mean Reciprocal Rank (Croft et al. 2010)

The time needed to answer a query is on average 14 milliseconds for no-expansion, 23 milliseconds for index-time expansion and 40 milliseconds for query-time expansion. One can see that time needed to answer a query increases with the expansion. In the case of index-time expansion the reason is that more indexes are mapped with the user query; in cases where documents are in just one language, there is not difference between this scenario and the scenario in which no expansion was performed. In the case of query-time expansion, the reason of the increment in the time needed to answer the query is the increase in the size of the query. In queries that cannot be expanded the increment of the time needed is marginal.

6.3 Measures used for the evaluation

We evaluated the multilingual expansion implemented in the search engine using precision, recall and two F-measures.

6.3.1 Precision

Probably the historically first formally stated evaluation metric for information retrieval engines is precision, proposed in (Cleverdon and Keen, 1966). The idea is very simple; the precision of a system for a query is the proportion of relevant results in the response. Precision is computed as follows:

$$\text{Precision} = \frac{tp}{tp + fp}$$

Where:

- tp: true positives
- fp: false positives
- tp+fp: overall retrieved results

6.3.2 Recall

Recall is defined as the proportion of relevant results that have been retrieved, and it is computed as follows:

$$\text{Recall} = \frac{tp}{tp + fn}$$

Where:

- tp: true positives
- fn: false negatives
- tp+fp: relevant results in the database

Recall and precision are usually inversely proportional. If a retrieval system returns more results, the recall can increase, but precision is likely to achieve lower values.

6.3.3 F-measure

F-measure (Van Rijsbergen, 1979) combines precision and recall with the relative importance of each of the measures set by the weighting constant ß.

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}$$

If precision and recall have the same importance, then ß=1 (F1 score). In retrieval systems that need a high precision and where recall is not so important, ß assumes values closer to 0[54]. On the other hand, there are systems that need to capture all the information possible and where precision is not so important.

Two assumptions were made for the first evaluation of the system. The first one is that precision and recall have the same importance for the user (F1). The second one is that recall is more important than precision, with $ß^2$=2 (F2).

6.4 Results

Results of the evaluation are summarized in Table 1

Expansion type	Precision	Recall	F1	F2
No expansion	1	0.38	0.41	0.40
Query-time	0.99	0.69	0.73	0.72
Index-time	0.90	0.74	0.81	0.79

Table 1

As expected the recall in the experiment without term expansion is quite low. In most cases a query in a language can retrieve documentary units in the same language. There are some exceptions, when the documentary unit includes terms in more than one language (for instance documentary units written in English often include different 'foreign' language terms where there is no single English equivalent, e.g. the terms Terezín and Theresienstadt).

The difference in recall between query-time expansion and index-time expansion is explained mostly by the presence of multiword terms, directly in the query, metadata text or produced in the expansion process. In those cases Solr has a better performance if terms are expanded at the time of indexing.

[54] Search engines of digital libraries belong to this group.

7 Conclusions

This report describes the implementation of the multilingual search interface of the EHRI portal. A substantial part of the report is about the use of the EHRI thesaurus developed by WP18 and an external source of knowledge, the geographic database GeoNames, for the multilingual and semantic expansion of the queries.

The results of the evaluation show that the use of the EHRI thesaurus and GeoNames lead to a higher performance of the search engine using both, index-time and query-time expansion.

Query-time expansion is very flexible, which allows for the introduction of new vocabularies or expanding the actual thesaurus without the need of re-indexing all the data. Since query-time expansion produces long queries if thesaurus terms are met, this method is not compatible with semantic expansion of the query.

Index-time expansion offers a slightly better recall, which leads to a higher coverage of the query in the results. Moreover, since the size of the query doesn't increase, the query can be further expanded using the semantic expansion service. However the necessary time to index the data increases dramatically[55] and the entire registry has to be re-indexed after each update in the thesaurus. Since the indexing of new harvested data can be made as an update in Solr, which continues to allow the querying of the previously indexed material, the increment of the necessary time to index data is not a very relevant problem. Full indexing of the data will be needed only after update of the controlled vocabularies.

For the immediate future, and until the Metadata Registry is stable, it is recommend to keep both forms of expansion for the system. The use of index-time expansion offers a higher potential because it allows combining multilingual term expansion at the index-time with semantic expansion at the query-time. That helps to bridge vocabulary gaps between indexed data and user query, increasing completeness in the response. But until the metadata registry is stable and full re-indexing of the data is still often needed, the use of query-time expansion is more suitable. The final functional implementation may include one or both forms expansion query depending upon operational and functional requirements, which can only be determined when the majority of the metadata is present and the controlled vocabularies are stable.

[55] The index-time increases from 1.5 minutes to almost 5 hours as mentioned in section 6.1

Glossary

API – Application Programming Interface (Web): defines the method by which services and resources maybe requested from a web service along with the structure of the response.

Client library: API used for writing applications, which communicate with a server (client applications).

Command line application: computer program designed to be used via a text-only computer interface, such as a text terminal

Cypher Query Language: declarative graph query language for Neo4j that enables access to the graph

GeoNames: geographical database available and accessible through various web services, under a Creative Commons attribution license. GeoNames covers all countries and contains over eight million place names.

HTML - Hypertext Mark-up Language: is the mark-up language for web pages.

HTTP POST: method or verb defined for HTTP. POST is used when the client needs to send data to the server.

JSON - JavaScript Object Notation: open format used to transmit data between a server and an application. JSON data objects consist in attribute-value pairs. http://json.org/

JSON path: component that allows one to find and extract relevant portions out of JSON structures.

Lemmatizer: computer program that attempts to find the lemma that corresponds to an inflected word.

Lucene (Apache Lucene): open source information retrieval software library supported by the Apache Software Foundation.

N-gram: An n-gram is a contiguous sequence of n items from a given sequence of text.

Neo4j: An open-source graph database implemented in Java. The EHRI project uses Neo4j for the metadata registry.

Open Source Software: software with this source code made available and licensed with a license in which the copyright holder provides the rights to study, change and distribute the software to anyone and for any purpose. http://en.wikipedia.org/wiki/Open-source_software .

RDF - Resource Description Framework: a language for representing information about resources in the World Wide Web http://www.w3.org/RDF/ .

REST - Representational State Transfer: a style of software architecture for distributed hypermedia systems such as the World Wide Web.

RESTful: architecture built conforming to the REST constraints.

Shingle: group of words that appear together corresponding to a search term.

SKOS – Simple Knowledge Organization System (SKOS), a common data model recommended by W3C for sharing and linking knowledge organization systems via the Web. SKOS is built upon RDF and it has been designed to represent structured controlled vocabulary.

SKOSification: Conversion of transformation of a terminology into SKOS standard

Solr: Open source search platform from the Apache Lucene project. It includes full text search, faceted search and rich document (PDF, Word, etc.) handling.

SPARQL: query language and protocol for RDF.

SQL – Structured Query Language: programming language designed for the management of data in relational databases.

Stemmer: computer program based in heuristics which chop off end of words and in some cases removes derivational prefixes and suffixes.

Stop words: Words that are filtered before or after processing of natural language texts.

URI - Uniform Resource Identifier (URI): a string of characters used to identify or name a resource on the Internet.

Wordnet: WordNet is a lexical database in which words are grouped in sets of synonyms or synsets. Each synset is linked to other synsets through different semantic relationships.

Wordnet similarity: semantic similarity between user query and concepts of Wordnet.

Web Service: a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

XML - Extensible Markup Language: a set of rules for encoding documents electronically. It is defined in the XML 1.0 Specification produced by the World Wide Web Consortium, and several other related specifications; all are fee-free open standards.
http://www.w3.org/TR/REC-xml

References

Athena (2010): Skosification of the existing metadata standards and terminology sets used by the participating Museums

Bhogal, J.; Macfarlane, A. and Smith, P. (2007): A review of ontology based query expansion. Information Processing & Management, 42(4).

Borut, Y.; Halpern, N.; Links, P.; Horsman, P.; Dargenmond, K; Mork, E. and Haardt, M. (2012): Thesaurus in one language (English). Deliverable 18.2. EHRI project.

CACAO (2009): Final report.
http://www.cacaoproject.eu/fileadmin/media/Deliverables/CACAO_D0.7.pdf

Cleverdon, C. and Keen, M. (1996) Factors Determining the Performance of Indexing Systems, Volume 2, *The College of Aeronautics, Cranfield*, 1966

Croft, W.; Metzler, D. and Strohman, T. (2010). Seach Engines: Information Retrieval in Practice. Addison Wesley, 2010. Cited in (Pera and Ng, 2010)

Gertner, H.; Borut, Y; Frojimovics. K.; Gherman, Y.; Rodriguez, K.J.; Links, P.; Horsman, P; Dargenmond, K; Mork, E. and Haardt, M. (2012): Thesaurus translated in other languages. Deliverable 18.3. EHRI project.

Hersch, W.; Price, S. and Donohoe, L. (2000): Assesing thesaurus-based query expansion using the UMLS Metathesaurus. In Proceedings of the AMIA Symposium.

ISO 2788 (1986): Documentation - Guidelines for the establishment and development of monolingual thesauri. November 1986

ISO 5964 (1985): Documentation - Guidelines for the establishment and development of multilingual thesauri. February 1885

JCGM 200 (2008): International vocabulary of metrology — Basic and general concepts and associated terms (VIM)
http://www.bipm.org/utils/common/documents/jcgm/JCGM_200_2008.pdf

Miron, G. and Shulhani, Sh; eds (2009): The Yad Vashem Encyclopedia of the Ghettos During the Holocaust. NYU Press

Pedersen, T. and Patwardhan, S. (2004): Wordnet::similarity - measuring the relatedness of concepts (2004). In Demonstration Papers at HLT-NAACL (2004)
http://www.d.umn.edu/~tpederse/Pubs/AAAI04PedersenT.pdf

Pera, M.S and Ng, Y.K. (2010): A Performance Evaluation Framework for Library Search Engines. In Proceedings of the 2010 Sixth International Conference on Signal-Image Technology and Internet Based Systems

Pieper, Dirk (2008): Cross-Language Information Retrieval und automatische Sacherschließung in Suchmaschinen am Beispiel der "Bielefeld Academic Search Engine“(BASE). Presentation at the "97. Deutscher Bibliothekartag”, Mannheim, Germany.
http://www.opus-bayern.de/bib-info/volltexte/2008/546/pdf/base_bibtag%202008.pdf

Van Rijsbergen, C. J. (1979): Information Retrieval. Butterworths London.