europeana
creative

D2.4 – Enrichment and Linking Service

This deliverable is software.

Co-funded by the European Union

The project is co-funded by the European Union, through the **ICT Policy Support Programme** as part of the **Competitiveness and Innovation Framework Programme (CIP)**.

http://ec.europa.eu/information_society/activities/ict_psp/

Österreichische
Nationalbibliothek

Europeana Creative is coordinated
by the Austrian National Library

Deliverable

Project Acronym: Europeana Creative

Grant Agreement Number: 325120

Project Title: Europeana Creative

D2.4 – Enrichment and Linking Service

Revision: Final

Authors: Alexandros Chortaras (NTUA)
Vladimir Alexiev (Ontotext)
Despoina Trivela (NTUA)

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

Österreichische
Nationalbibliothek

Europeana Creative is coordinated
by the Austrian National Library

Revisions

Version	Status	Author	Date	Changes
0.1	Draft	Despoina Trivela, NTUA	December 18, 2014	Template
0.2	Draft	Despoina Trivela, NTUA	January 9, 2015	Section 2
0.3	Draft	Alexandros Chortaras, NTUA	January 20, 2015	Section 2
0.4	Draft	Vladimir Alexiev, ONTO	January 27, 2015	Section 3
0.5	Final draft	Despoina Trivela, NTUA	January 27, 2015	finalize
0.6	Final draft	Vladimir Alexiev, ONTO	January 27, 2015	Final editing and changes
1.0	Final	Susanne Tremml, ONB	February 4, 2015	Minor changes

Distribution

Version	Date of sending	Name	Role in project
0.1	January 29, 2015	Hugo Manguinhas, EF	Reviewer
0.2	January 29, 2015	Antoine Isaac, EF	Reviewer
0.6	February 4, 2015	Susanne Tremml, ONB	Project Manager
1.0	February 27, 2015	Marcel Watelet, EC	Project Officer

Approval

Version	Date of approval	Name	Role in project
1.0	February 26, 2015	Max Kaiser, ONB	Project Coordinator

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This deliverable contains original unpublished work except where clearly indicated otherwise.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.

This deliverable reflects only the author's/authors' views and the European Union is not liable for any use that might be made of information contained therein.

Table of Contents

1. Introduction

The purpose of this deliverable “D2.4 Enrichment and Linking Services” is to describe the linking and enrichment services that were developed in order to link the Europeana content with external web resources. The software that is presented here serves the purposes of Task 2.4- Linking to external web resources. The linking services aim to exploit the semantic web technologies in order to link Europeana metadata to the emerging web of data.

The enrichment and linking service allows for entity extraction from text, and provides links to relevant information deriving from external data sources. The knowledge sources are also used to identify semantic relationships between the identified concepts, and to interlink the metadata in Europeana.

This deliverable gives a report on the enrichment and linking services of the Europeana metadata. It provides helpful information for the Pilot applications that wish to exploit these services to enrich and link the metadata with external related web resources such as such as Freebase, DBpedia, Wikipedia, VIAF, Getty, Geonames.

The document is divided into two parts. In the first part, functionalities of the enrichment tool developed by NTUA are presented and details explaining its usage are provided. The second part presents a study performed by Ontotext concerning the vocabularies that could be used within enrichment and a potential co-referencing process. In particular, the semantic data and name sources integrated by Ontotext into a name data service are described.

2. Enrichment and Linking process

In this section the enrichment and linking process framework developed by NTUA is presented. The process was implemented using the FReS platform which has been developed by NTUA and is a workflow editor that allows the graphical definition of complex data processing workflows.

2.1 Overview of FReS

Before describing in more detail the actual implementation of the enrichment and linking process, this section provides an overview of the FReS functionalities. As mentioned above, FReS may be used for the definition of a custom, reusable data processing *workflow*. The building block of each workflow is a *node*, which performs some data import, processing, or export operation. A node inserted on the workflow editor is shown in 1.

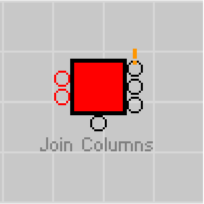
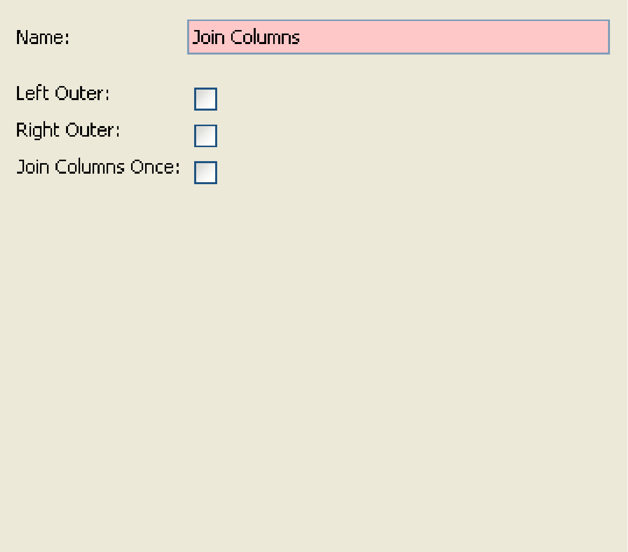


Fig. 1 A FReS node

A node may have some *input ports*, some *parameters*, and some *output ports*. E.g. the node shown in 1 has two inputs and three outputs. The input ports which are not optional and on which it is mandatory to connect an input for the operation of the node to be possible are coloured in red. The parameters of the node as well as some configurations related with the input and output ports are controlled through the parameters panel of each node, discussed in the following sections. The port below the node is an extra input port that can optionally be used for passing the node parameters as tabular data, instead through the parameter panel interface.

Most nodes work on *tables*, which is the basic internal working model of FReS. Thus, the data expected by a node to be found in its input ports and that are produced for its output ports are usually tables; each table column bears a distinct name, which is used for referencing it. Each table column has also a type, which can be either *numerical*, *boolean*, *string, annotated text* or a *collection*. The annotated text type is a special structure that allows the representation of a text along with some string annotations. Each annotation, which always refers to a particular segment of the underlying text (having a *start* and *end* location) has a *class* and a *value.* The type and the value of an annotation are both user-provided strings that are used to describe the intended meaning of the annotation. E.g. a part-of-speech text may have for each word an annotation of class “pos” and value the particular part-of-speech detected for the respective word. Finally, the collection type is a set or an array of values.

The parameters of a node are modifiers to the node operation (e.g. a node that replaces some text by some other text may have a case sensitive parameter) and may be of numerical, boolean or string type. They are provided to the node through the appropriate interface, as shown in 2, for the node of 1.



Name: Join Columns

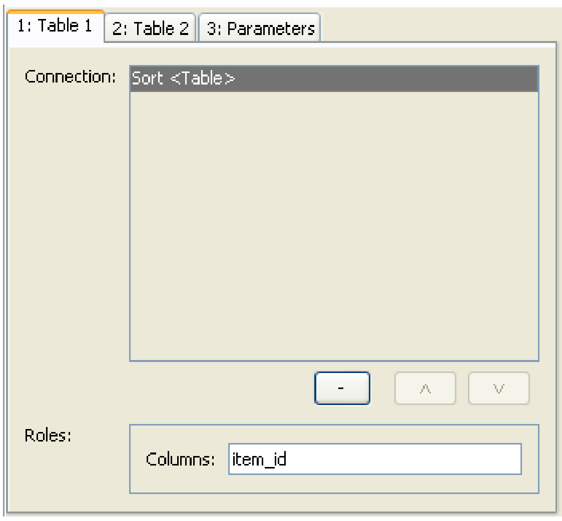
Left Outer: □

Right Outer: □

Join Columns Once: □

Fig. 2 Node parameters

Since the input of a node is a table which may have several columns, depending on the operation of the node, it is necessary to define which column of an input table should be used for the node operation. Hence, for each node taking as input node it is necessary to define the input *roles*, i.e. the columns of the input tables which will provide the data required for the node operation. Fig. 2 shows the interface for defining the input roles for the node of 1. Since the node has two input ports (representing the two tables to be joined) the column on which the join should be performed has to be defined for each input.



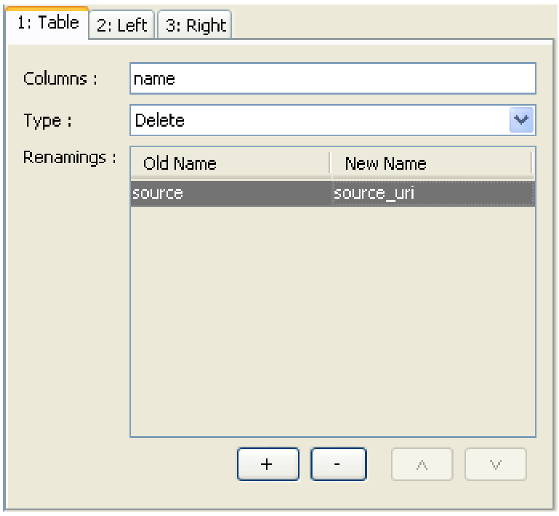
Connection: Sort <Table>

Roles:

Columns: item_id

Fig. 3 Definition of input roles

The output tables of a node can also be configured, so that some columns are deleted, or renamed. This is done through the outputs interface shown in Fig. 4. As shown in the figure, a renaming table can be constructed so that some columns of the output tables change name after the execution of the node.



Columns : name

Type : Delete

Renamings :

Old Name	New Name
source	source_uri

.

Fig. 4 Definition of changes in the output table columns

Although the data that is present in the input or output ports of a node is usually a table, in some cases it can be a *model*, which represents a more complex data structure whose tabular representation is not meaningful. Examples of models used in FReS are data source models such as an OWL Ontology or RDF.

When executed, a node, according to its functionality, reads the data present in its input ports, and parameters, applies its business logic and writes out the results in its outputs. In the case of table inputs and outputs, usually the node will iterate over the rows of the input tables, do some processing (e.g. replace some text by some other text) and write out a modified table in the output.

FReS provides a wide range of nodes that offer the most commonly needed functionalities, and they are divided into several groups, according to their functionality. Additional custom nodes can easily be developed by using the FReS node development API.

The relevant groups and nodes used for the generation of the enrichment and linking service are the following:

Data Import Nodes

File Import: It takes no inputs, put a file path. It reads the contents of the file and converts them into a table in the output. The column delimiters for the generated table as well as the encoding of the file path to be read are provided as parameters.

Table Editor: It takes no inputs, but provides an interface through which a table can be specified by the user as a parameter. The user specifies the number of desired rows and columns, and then the names of the columns and the contents of each cell. When executed, the tabular data inserted by the user are converted into a FReS table in the output.

Structured Data Nodes

RDF Store: It takes no inputs, but a parameter with an SPARQL Endpoint or an RDF file path. In the output it produces a RDF Source model.

OWL Ontology: It takes no inputs, but a parameter with an OWL file path or URI. In the output it produces an OWL Ontology model.

Merge OWL Ontologies: It takes as input one or more OWL Ontology models, and produces an OWL Ontology model that results by merging all the input ontologies.

OWL Reasoner: It takes as input an OWL Ontology model and executes a reasoning task over it, as defined by the parameters. The parameter is a query expression that asks e. g. for the computation of the superclasses, subclasses, equivalent classes of a certain class, or for the retrieval of all objects that are instances of a certain class. The node takes an optional second input that can provide, if needed, values for any parametrical expressions used in the query expression. A parametrical expression is a construct of the form @[col_name], which stands for the content of the column with name col_name of the input table. E.g. the pattern <@[col1] > will produce the value <http://www.europeanafashion.eu/> if the contents of the current row of the column with name col1 of the input table is http://www.europeanafashion.eu/. The node can also compute the classification tree of the input ontology model, in which case the output is a Tree Model.

Classification Match: Takes as input a Tree Model, and a table containing some values which are matched against the nodes of the tree. In the output the values of the parent or children nodes (as determined be a parameter) of the tree for each match are written out in a tabular format.

Define Query: It takes as optional input a RDF Source model, and is used for producing a SPARQL query over that model. The query is provided as a parameter, and when executed, the node writes it out in the output as a single column and row table.

RDF Query: It takes as input an RDF Source model, a table containing a SPARQL query and optionally a table providing actual data for any parametrical expressions contained in the SPARQL query. It executes the query over the data source and writes out the retrieved data as a table.

RDF Describe: It takes as input an RDF Source model and as parameters a SPARQL describe query. It executes the describe query and writes out the answers as an RDF Source model in the output.

RDF Update: It takes as input an RDF Source model and as parameters a SPARQL update query. It executes the update query and writes out the result as an RDF Source model in the output.

RDF Add Prefixes: It takes as input an RDF Source model and a table containing a list of prefixes to full URI mappings. It enriches the model of the input with the prefix mappings and writes out the result in the output.

Processing Nodes

Search: It takes as input a table with a text column on which the search is going to be applied, and a table with the strings to be searched for. If the search strings are but a single string it can be passed as a parameter. It supports both regular expression and simple string matching. The output table consists of the rows of the input table that contained at least one match. In case of a regular expression search, grouping is supported, and the output table contains in separate columns the matched groups that where extracted.

Replace: It takes as input a table with a text column on which the replacements are going to be applied, and a table with the strings to be searched for and the respective replacements. If the search-replacement strings are a single pair they can be passed as a parameter. It supports both regular expression and simple string matching.

Process Columns: It takes as input a table and produces an output table whose columns have resulted by concatenating the contents of the input table and any arbitrary strings. For each output column a parametrical expression must be defined by the user which is provided to the node as a parameter.

Keep First: It takes as input a table, and removes any duplicate rows.

Join Columns: It takes as input two tables, and computes their natural join. It has three output ports. The first contains the joined table, the second the row of the first table that are not part of the joined table, and the third the rows of the second table that are not part of the joined table. By using the parameters it can be specified whether a left/outer join is desired or not.

Split: It takes as input a table and splits the contents of one of its columns on a string that is specified as a parameter. Each part of the splitting operation is included in the output table as a new row.

Filter: It takes as input a table and one or more conditional parametrical expressions. For each expression an output table is generated which contains the rows of the input table that matched the corresponding expression.

Group Collect: It takes as input a table for which a base and a collect role are defined. When executed, it collects all different values appearing in the collect columns for the rows that have the same base values and writes them out in the output table as a collection.

Add Columns: It takes as input two tables. The first is the data table and the second is a table containing some rows each of one which will be added as a column to each row of the first input table.

Text Processing Nodes

Regex Annotate: It takes as input a table containing some string column, and another table containing three columns, one containing a string or a regular expression, one containing an annotation class string, and one containing the corresponding annotation values. The node tries to match the strings or regular expressions of the second input on the first table, and each time a match is detected, the respective string part is annotated by the respective annotation class and values. The parameters specify if the matching will be regular expression or keyword based and whether it will be case sensitive. The output is a table containing an annotated text column.

Extract Annotations: It takes as input a table containing an annotated text column, and as parameter a search expression that defines which annotations should be extracted. The output is a table containing a row for each annotation matching the search expression. The search expression defines the classes and values of the annotations that should be matched.

Data Output

Export Table: It takes as input a table, and writes it out in a file on the file system as a csv file.

Export RDF Model: It takes as input an RDF model, and writes it out in a text file on the file system.

A workflow is built by adding nodes and connecting an output port of a node to an input port of another node. This defines the desirable flow of data at execution time. Each node can be executed, but in order for this to be feasible all the nodes connecting to its input ports have to be executed first. A table input port can accept more than one inputs, in which case at execution time the several input tables are concatenated into one according the their column names.

Apart from the above described nodes, FReS provides two special *structures*: The *loop* structure and the *subroutine* structure. These are two-node structures, i.e. pairs of special nodes, the first one of which is the input part of the structure and the second one the output part of the structure. Both the input and output parts are simple nodes that have the same number of input and output ports, and simply relay the data present in their inputs to their outputs. The output ports of the input part are expected to be connected to some nodes which make up a workflow, whose outputs are then connected to the inputs of the output part of the two-node structure. Thus, the special two-node structures act as delimiters for an internal workflow that should be executed in a particular way. In the case of the loop structure the internal workflow is executed in a controlled way by concurrently iterating over the rows of the several inputs. In the case of the subroutine structure, the internal workflow acts as a reusable sub-workflow that can be used in other parts of the main workflow, like a programming language procedure. In fact, whenever a subroutine structure is added in the main workflow, the available list of nodes is enriched by a new, *custom* node representing the new structure, which can be used as a normal node (performing a user-defined complex operation).

A workflow constructed using the FReS platform can be executing within the FReS platform so as to overview the execution results. In particular, the interface of each node contains a results panel, which provides access to the output tables of the respective node after its execution. A FReS workflow can be exported for reuse as an xml file. Moreover, a FReS workflow containing a single subroutine structure can be deployed as a REST service. In this case, a REST wrapper should be included in the workflow in order to convert the REST parameters to appropriate inputs of the workflow, and similarly for the outputs.

2.2 Overview of the Enrichment and Linking Process

This section describes the enrichment and linking processes as they were implemented in the form of a workflow using the above-described FReS platform.

The enrichment and linking processes consist conceptually of several self-contained sub-processes that permit a more general and adjustable implementation of the entire process. The input for the entire process are the sets of metadata properties values associated with each data item, while the final output are the original metadata properties values extended with some additional URI's which are the products of the enrichment and linking process and represent relevant to the processed metadata vocabulary values.

The following section describes the enrichment and linking process in terms of the sub-processes of which it consists, and provide a running example to illustrate better the whole process. The running example was taken from the EDM-fp dataset that was used in the Europeana Fashion project. The data are described in terms of the EDM-fp schema, which is a general vocabulary that provides several metadata properties. EDM-fp schema uses several properties such as dc:type, dc:title, dc:description, dcterms:medium, edmfp:technique, gr:color, dcterms:spatial, etc., that correspond respectively to the type of an item, its title, a brief description of it, the materials from which it has been made, the techniques that have been applied for its production, its colours, its place of origin, etc. As values for the metadata properties URI's defined in the Fashion Thesaurus, which has been developed for Europeana Fashion project, can be used. For example, the categories of the Fashion Thesaurus that describe object types (e.g. Fashion Object > Costume > Main Garment > Coat > Tailcoat) can be used to assign values to the property dc:type, the categories for techniques (Technique > Weaving Technique > Weave > Taffeta > Batiste) can assign values to edmfp:technique, etc. In addition to the URIs, several metadata properties of the EDM-fp schema allow for free text values.

While a controlled vocabulary value is a resource with a certain URI that has a well-defined meaning (which is defined in the context of the vocabulary or ontology in which it is included) and can be used in conjunction with semantic technologies to allow e.g. semantic query answering and other tasks that involve reasoning, a free text value lacks any semantics and can be used in a limited way only in conjunction with traditional keyword search (i.e. string matching). The challenge in the case of free text values is to add some kind of semantic characterization on them so that the relevant information can be described by formal semantics. Then, the data related to text descriptions will have a specific meaning as in the case of vocabulary URIs, and will be defined by formalisms thatcan be understood by semantic web technologies. Towards this direction, the process described in this section aims initially at extracting some significant concepts from the free text descriptions. In particular, within the enrichment process a set of concepts described by URIs of the Fashion Thesaurus or of common vocabularies is mapped to a free text description. The enrichment process is thus at the same time a linking process of the EDM-fp content to external resources of the Linked Data cloud, if the enrichment is done using external vocabularies.

The enrichment and linking process consists of 5 subroutines. Within the data retrieval process, data coming from different data content providers are gathered from an RDF repository. Next, depending on the area of interest, appropriate SPARQL queries are posed in order to gather the metadata properties values that will subsequently be described in terms of an ontology as part of the enrichment/linking process. The concept label generation process serves to describe some concepts included in an ontology of preference that can be associated with the data properties. Then, the thesaurus-based enrichment process maps the data to the aforementioned concepts. The data are linked to DBpedia entities within the DBpedia based linking process. Finally, the enriched data generation process produces the enriched/linked RDF file.

The entire enrichment/linking process for a certain provider as an implemented FReS workflow is shown in Fig. 5, where each one of the five above-mentioned subroutines corresponds to a custom node (Retrieve Data, Ontology Labels, Annotate, DBpedia Lookup and Generate RDF). In the following the operations performed by each one of these custom nodes are described.

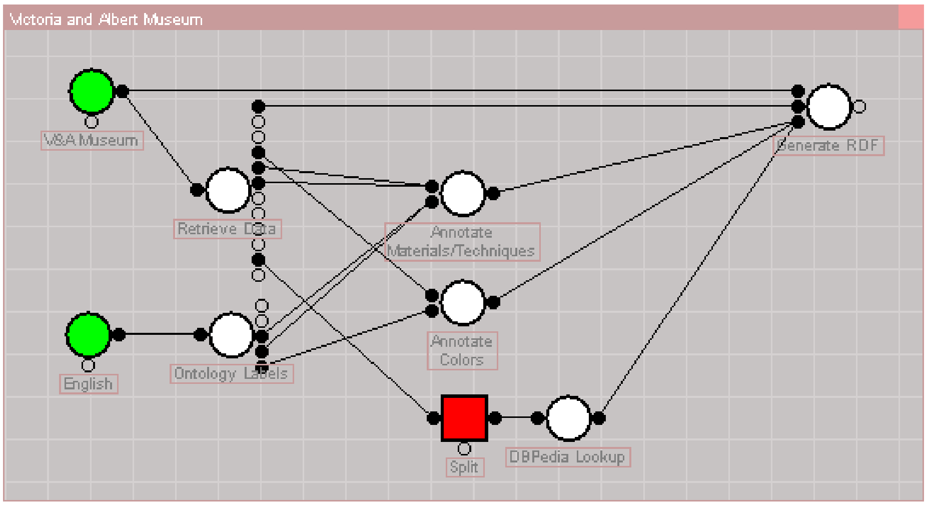


Fig. 5 Enrichment/Linking workflow

2.2.1 Data Retrieval Process

In this section the sub-process that is responsible for retrieving data is described. considerate considers data that were entered into the Fashion project's portal through the MINT system (http://mint.image.ece.ntua.gr/redmine/projects/mint/). The data are initially encoded in RDF/XML format and each item description is included in one file.

The following example presents an extract of the file that corresponds to the item with identifier `http://mint-projects.image.ntua.gr/europeana-fashion/1023_0bfe0b0844b9067a57fe6b86ffc06b3264c051f4a14a59ab7bd9f1848fcf157c_2716399_2607_555-1893` (for brevity Item X) provided by the Victoria and Albert Museum.

```
<rdf:RDF>
<edm:ProvidedCHO rdf:about="http://mint-projects.image.ntua.gr/europeana-
fashion/1023_0bfe0b0844b9067a57fe6b86ffc06b3264c051f4a14a59ab7bd9f1848fcf157c_2716399_2607_5
55-1893">
        <dc:subject
xml:lang="en">grapes;borage;honeysuckle;carnation;pansy;columbine</dc:subject>
        <edm:type>IMAGE</edm:type>
        <edmfp:technique>hand sewing</ edmfp:technique>
        <dc:description xml:lang="en">Purse, embroidered canvas with silk on silver ground,
plaited silk strings, 1600-1650, English. Linen, silk, silver and silver-gilt threads, silk
thread; hand sewn, hand embroidered, hand plaited.</dc:description>
        <dcterms:created>1600/1650</ dcterms:created>
        <dcterms:medium>silk taffeta</dcterms:medium>
        <dc:title>Purse</dc:title>
        <dcterms:extent>Length 11.7cm, Width 11.3cm (approx., bag only)</dcterms:extent>
        <dcterms:medium>linen</dcterms:medium>
        <dc:date>1600/1650</dc:date>
        <edmfp:technique>plaiting</edmfp:technique>
        <dc:identifier>555-1893</dc:identifier>
        <edmfp:localType>Purse</edmfp:localType>
        <dcterms:medium>silver gilt thread</dcterms:medium>
        <dcterms:medium>silk thread</dcterms:medium>
        <dcterms:spatial>
                <edm:Place><skos:prefLabel>Great Britain</skos:prefLabel></edm:Place>
        </dcterms:spatial>
        <edmfp:technique>hand embroidery</edmfp:technique>
        <dc:type rdf:resource="http://thesaurus.europeanafashion.eu/thesaurus/10140"/>
        <dcterms:medium>silver thread</ dcterms:medium>
</edm:ProvidedCHO>
</rdf:RDF>
```

The metadata files for all items were gathered into an RDF store in order to enhance the data retrieval process. Then, SPARQL queries can be posed over the repository in order to extract information from each item. Thus, depending on the area of interest one can select the properties on which one would like to apply the enrichment/linking process and construct the appropriate SPARQL queries. For example, depending on the use of the data one could ask for properties such as dc:type, or dc:title within the SPARQL query and retrieve the relevant values. It is to notice that the exact set of retrieved properties is a parameter to the data retrieval process.

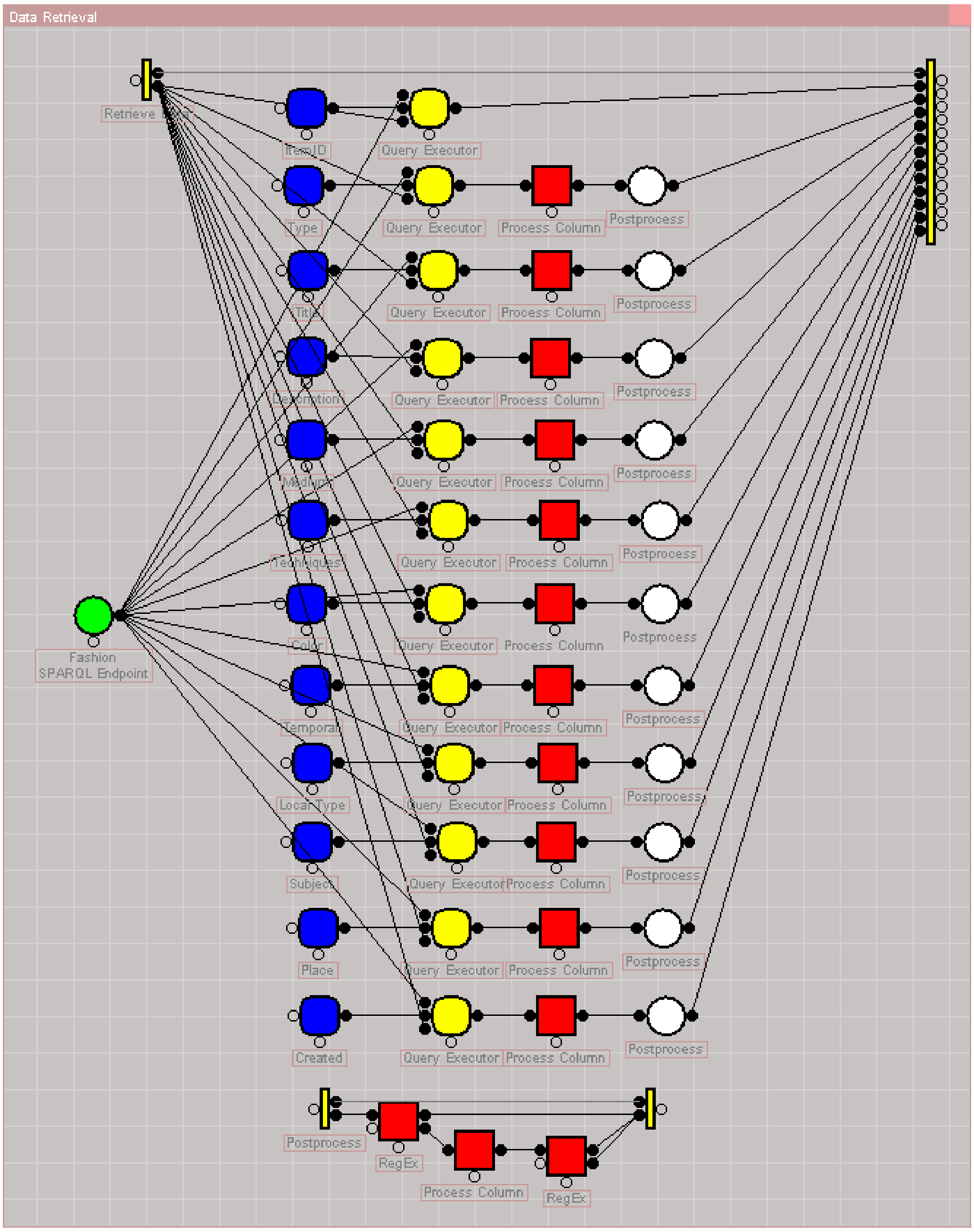
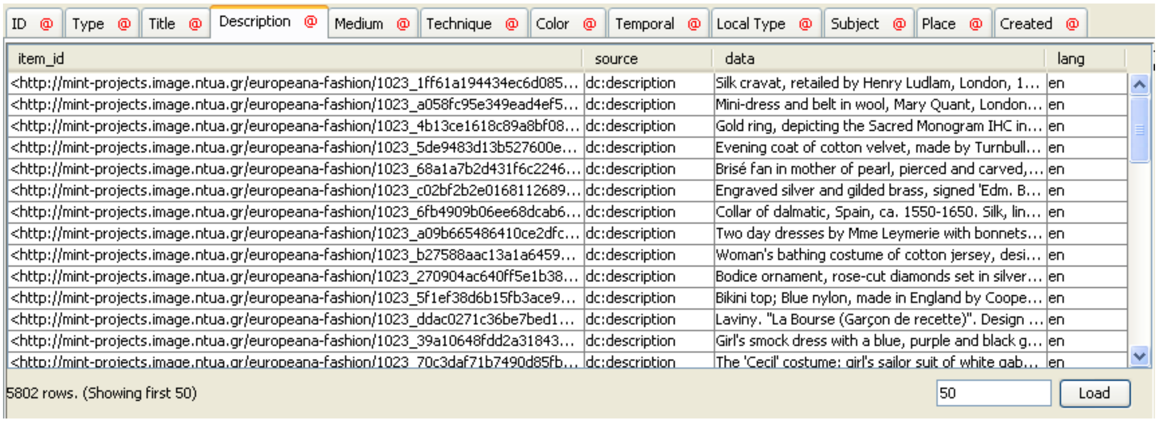


Fig. 6 Data retrieval process workflow

Fig. 6 shows the data retrieval workflow that is hidden behind the corresponding node. It consists of several SPARQL queries executed over the RDF store that holds the data. Each SPARQL query gives rise to a distinct result table, which after some normalization preprocessing (removal of quotes, separation of actual content from accompanying language identifier done by the postprocess custom node) makes up the output of the data retrieval sub-process. Each SPARQL query is so formed that it retrieves one metadata property value for each date item in the RDF Store (in particular the edm:ProvidedCHO, dc:type, dc:title, dc:description, dcterms:medium, edmfp:technique, gr:color, dcterms:temporal, edm:localType, dc:subject dcterms:spatial and dcterms:created metadata). Each output of the data retrieval subprocess is a table that contains one row for each property value; the column tables are the item identifier, the metadata property name, the property content, and the language identifier for the content, if available. As an example Fig. 7 shows the output table for the 4th output which corresponds the dc:description field.



item_id	source	data	lang
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_1ff61a194434ec6d085…	dc:description	Silk cravat, retailed by Henry Ludlam, London, 1…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_a058fc95e349ead4ef5…	dc:description	Mini-dress and belt in wool, Mary Quant, London…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_4b13ce1618c89a8bf08…	dc:description	Gold ring, depicting the Sacred Monogram IHC in…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_5de9483d13b527600e…	dc:description	Evening coat of cotton velvet, made by Turnbull…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_68a1a7b2d431f6c2246…	dc:description	Brisé fan in mother of pearl, pierced and carved,…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_c02bf2b2e0168112689…	dc:description	Engraved silver and gilded brass, signed 'Edm. B…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_6fb4909b06ee68dcab6…	dc:description	Collar of dalmatic, Spain, ca. 1550-1650. Silk, lin…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_a09b665486410ce2dfc…	dc:description	Two day dresses by Mme Leymerie with bonnets…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_b27588aac13a1a6459…	dc:description	Woman's bathing costume of cotton jersey, desi…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_270904ac640ff5e1b38…	dc:description	Bodice ornament, rose-cut diamonds set in silver…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_5f1ef38d6b15fb3ace9…	dc:description	Bikini top; Blue nylon, made in England by Coope…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_ddac0271c36be7bed1…	dc:description	Laviny. "La Bourse (Garçon de recette)". Design …	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_39a10648fdd2a31843…	dc:description	Girl's smock dress with a blue, purple and black g…	en
<http://mint-projects.image.ntua.gr/europeana-fashion/1023_70c3daf71b7490d85fb…	dc:description	The 'Cecil' costume: girl's sailor suit of white gab…	en

5802 rows. (Showing first 50)

50 Load

Fig. 7 Data retrieval process output.

For the running example, the joined table is provided below (for the properties dc:title, dc:description, dcterms:medium, edmfp:technique, dcterms:spatial) for Item X (described by the above RDF/XML file):

Table 1

item_id	Property	data	lang
…1023_0bfe0b0844…	dc:title	Purse	
…1023_0bfe0b0844…	dc:description	Purse, embroidered canvas with silk on silver ground, plaited silk strings, 1600-1650, English. Linen, silk, silver and silver-gilt threads, silk thread; hand sewn, hand embroidered, hand plaited.	en
…1023_0bfe0b0844…	dcterms:medium	silk taffeta	
…1023_0bfe0b0844…	dcterms:medium	Linen	
…1023_0bfe0b0844…	dcterms:medium	silver gilt thread	
…1023_0bfe0b0844…	dcterms:medium	silk thread	
…1023_0bfe0b0844…	dcterms:medium	silver gilt thread	
…1023_0bfe0b0844…	edmfp:technique	hand embroidery	
…1023_0bfe0b0844…	dcterms:spatial	Great Britain	

Clearly, the values for the properties dcterms:medium and edmfp:technique are amenable to enrichment by using the Fashion Thesaurus controlled vocabulary, while the value for the dcterms:spatial property is amenable to linking to an external resource, in particular to a DBpedia resource.

2.2.2 Concept Label Generation Process

This sub-process serves to associate metadata properties values with concepts included in an ontology. To begin with, the Fashion Thesaurus, which has been developed as a SKOS based taxonomy, was converted into an OWL Fashion Ontology, so that it could be enriched with more complex axioms and can be exploited within reasoning tasks. By taking into account the top-level categorization incorporated into the Fashion Thesaurus taxonomy five main concept categories that can be of interest for the enrichment process were distinguished, namely the objects, events, material, techniques and colours categories. Each one of them has a corresponding EDM-fp property, namely dc:type (which can capture both objects and events depending of the nature of the item), dcterms:medium, edmfp-technique and gr:color.

Fig. 8 shows the workflow that performs the concept label generation process which is hidden behind the Ontology Labels node of Fig. 5. Its purpose is to construct a table for each one of the aforementioned five ontology categories, that will be used for performing the enrichment process, i.e. a table that will associate an ontology concept (a uri) with a surface label form that will be looked for within the data. This operation should be performed for all languages supported by the data.

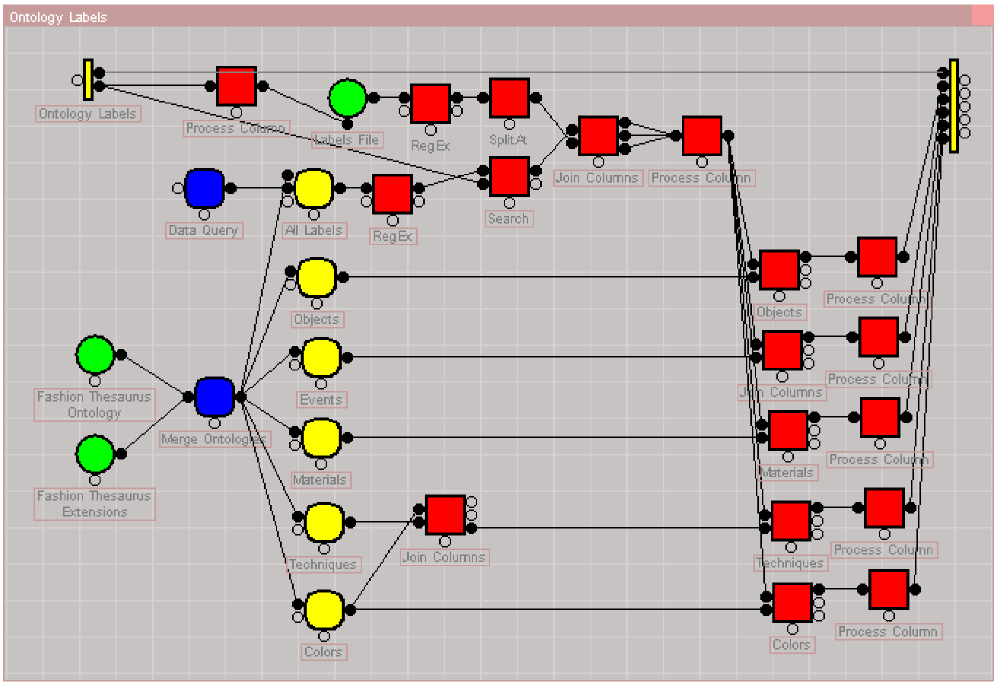
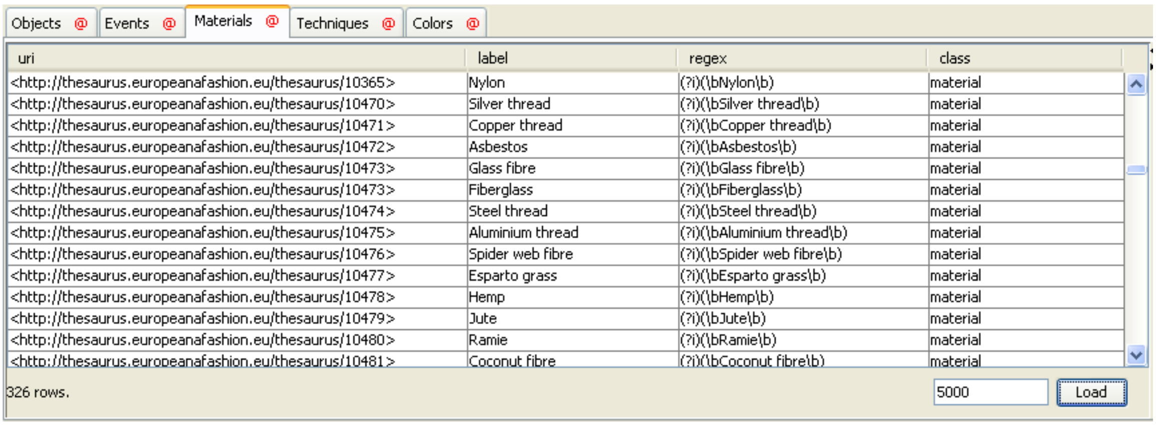


Fig. 8 Label generation process workflow

For this reason the workflow of Fig. 8 takes a single input that is a one cell table with the code of the language for which the labels will be created and produces five tables, one for each of the selected categories. In order to achieve this, the workflow first classifies the Fashion Ontology, then retrieves all sub-concepts for each of the five categories and associates them with the respective EDM-fp property names. For the enrichment process to be more flexible, the surface label form generated by this sub-process is in fact a regular expression, which may generated either automatically from the labels provided by the ontology for each concept, or manually by an external file. The resulting table, for the materials category is shown in Fig. 9. The first column is the ontology concept URI, the second one is the label provided for the concept by the ontology, the third is the generated regular expression, and the last one is the category that reflects the respective EDM-fp metadata property.



uri	label	regex	class
<http://thesaurus.europeanafashion.eu/thesaurus/10365>	Nylon	(?i)(\bNylon\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10470>	Silver thread	(?i)(\bSilver thread\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10471>	Copper thread	(?i)(\bCopper thread\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10472>	Asbestos	(?i)(\bAsbestos\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10473>	Glass fibre	(?i)(\bGlass fibre\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10473>	Fiberglass	(?i)(\bFiberglass\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10474>	Steel thread	(?i)(\bSteel thread\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10475>	Aluminium thread	(?i)(\bAluminium thread\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10476>	Spider web fibre	(?i)(\bSpider web fibre\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10477>	Esparto grass	(?i)(\bEsparto grass\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10478>	Hemp	(?i)(\bHemp\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10479>	Jute	(?i)(\bJute\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10480>	Ramie	(?i)(\bRamie\b)	material
<http://thesaurus.europeanafashion.eu/thesaurus/10481>	Coconut fibre	(?i)(\bCoconut fibre\b)	material

326 rows.

Fig. 9 Generated labels for the materials category

2.2.3 Thesaurus-based Enrichment Process

In this section the core component of the enrichment process is presented. The thesaurus-based enrichment process takes as input the union of one or more tables generated by the data retrieval sub-process (Table 1) and one or more tables generated by the Concept Label Generation Process (Fig. 9) and applies regular expression string matching techniques in order to detect instances of the provided labels in the content data. The goal is to detect occurrences of the values in the label column of Fig. 9, within the textual values of the data column of Table 1. When such a match is detected, a new row is added in the output table, which consists of a copy of the relevant row of Table 1 extended with the URI and category columns of the row of Fig. 9 that gave rise to the match. Fig. 10 shows the workflow that does this operation and is hidden behind the Annotate nodes Fig. 5.

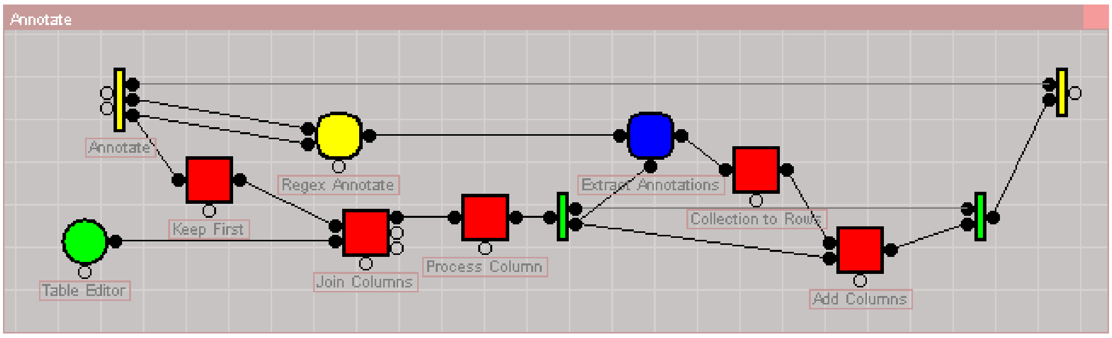
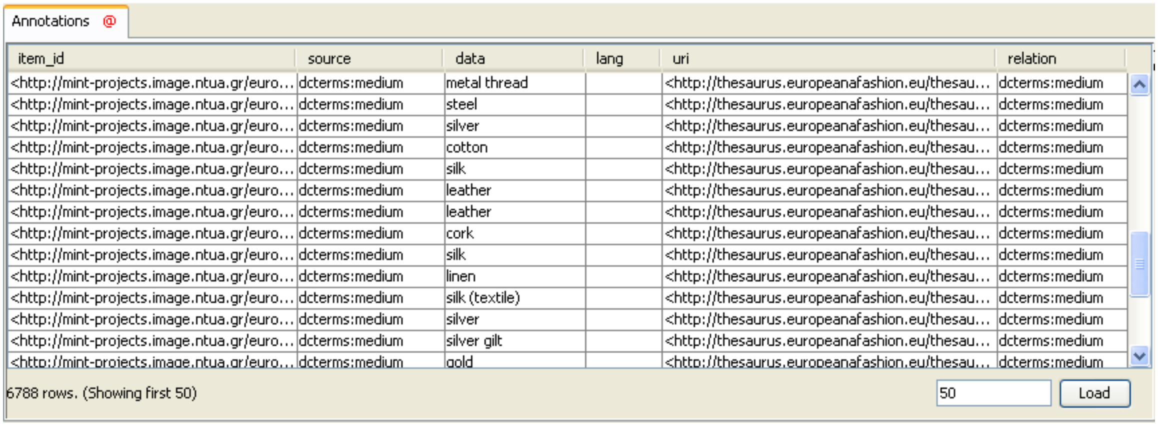


Fig. 10 Thesaurus-based annotation process

Fig. 11 shows the resulting table for the Annotation node that tries to map the contents of the original dcterms:medium and edmfp:techniques to the materials and techniques URI provided by the ontology.



item_id	source	data	lang	uri	relation
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	metal thread		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	steel		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	silver		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	cotton		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	silk		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	leather		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	leather		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	cork		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	silk		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	linen		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	silk (textile)		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	silver		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	silver gilt		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium
<http://mint-projects.image.ntua.gr/euro…	dcterms:medium	gold		<http://thesaurus.europeanafashion.eu/thesau…	dcterms:medium

6788 rows. (Showing first 50)

50

Fig. 11 Annotation results table.

The table below shows the result of applying the enrichment process on Item X. As shown in Fig. 10, the full enrichment process uses twice the annotate node, once for detecting occurrences of material and techniques instances in the dcterms:medium and edmfp:techniques properties, and another for detecting instances of colours in the dc:description field.

Table 2

item_id	source	data	URI	relation
…1023_0bfe0b0844…	dc:description	Purse, … *silver*	thesaurus:10408	gr:color
…1023_0bfe0b0844…	dcterms:medium	*linen*	thesaurus:10355	dcterms:medium
…1023_0bfe0b0844…	dcterms:medium	*silver* thread	thesaurus:10470	dcterms:medium
…1023_0bfe0b0844…	dcterms:medium	*silk* thread	thesaurus:10352	dcterms:medium
…1023_0bfe0b0844…	dcterms:medium	*silk* taffeta	thesaurus:10352	dcterms:medium
…1023_0bfe0b0844…	dcterms:medium	*silver* gilt thread	thesaurus:10547	dcterms:medium
…1023_0bfe0b0844…	dcterms:medium	silk *taffeta*	thesaurus:10372	edmfp:technique
…1023_0bfe0b0844…	dcterms:medium	hand *embroidery*	thesaurus:10428	edmfp:technique

As illustrated above, each property of an item that is accompanied by a free text is now associated also with a set of Fashion Thesaurus URI's. The output table contains an extra category column which determines the type of the association, more in particular the EDM-fp property that should be used to bind the item with the particular URI value. In the data column, the exact words of the original textual value that gave rise to the particular row are shown in italics.

2.2.4 DBPedia-Based Linking Process

The purpose of this subprocess is to link metadata properties to the external source of DBpedia. This is the core process that inks the local metadata to the web of data. By using entity extraction techniques, the entities are obtained from the free text describing the metadata properties. Once the entities are obtained, links are provided to related information deriving from external data sources, in particular DBpedia. The workflow that does the linking process is shown in Fig. 12. This process takes again as input the union of one or more tables generated by the Data Retrieval Process and tries to determine matching URI's in the external ontology for the named entities that appear in the textual values. More precisely, the process transforms each named entity occurrence to an acceptable DBpedia URI's and by querying the DBpedia SPARQL endpoint checks if the particular URI is indeed an existing DBpedia URI and is an instance of a desired top-level concept of the DBpedia ontology.

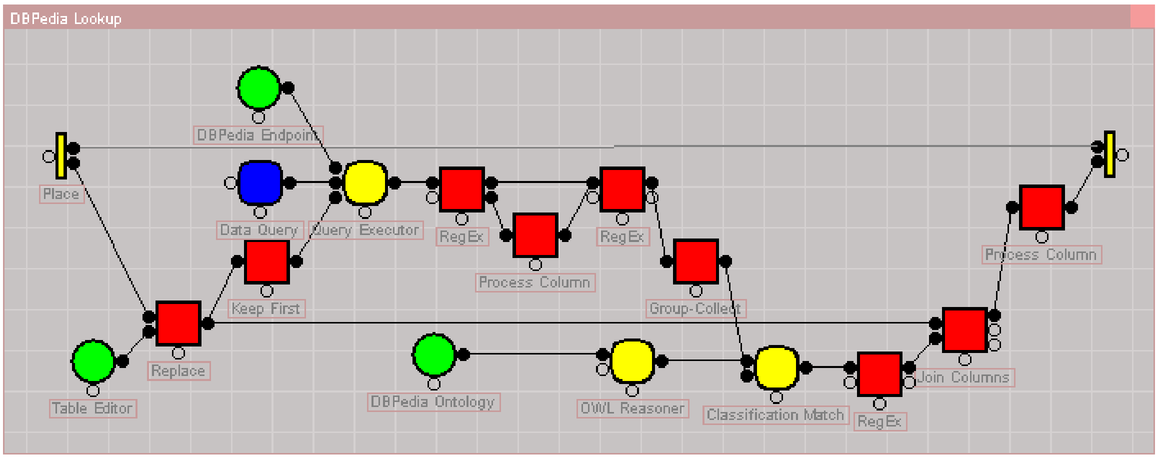
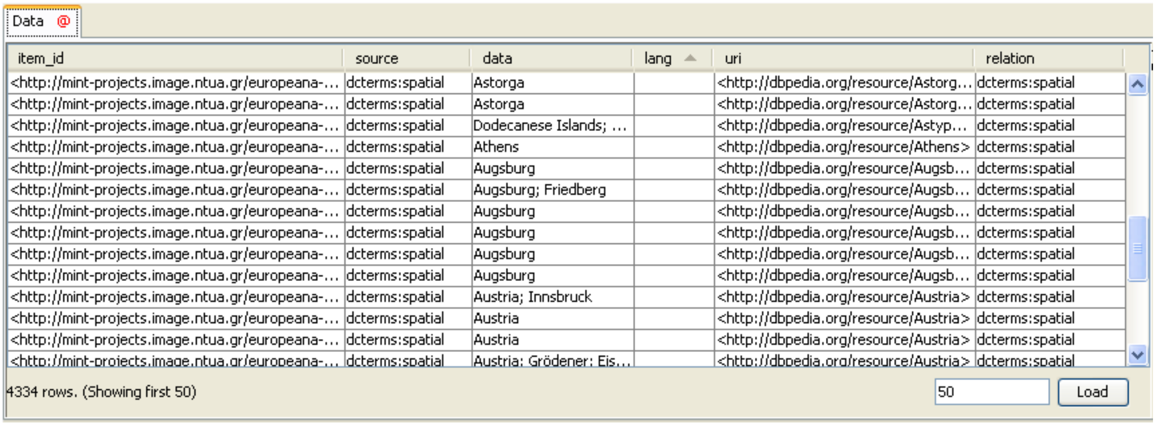


Fig. 12 DBPedia-base linking process

The output table of the process is shown in Fig. 13, which is similar to the output of the thesaurus-based enrichment processes, the difference being that the detected URI's are external links to the DBpedia database, instead of Fashion Thesaurus links.



Data

item_id	source	data	lang	uri	relation
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Astorga		<http://dbpedia.org/resource/Astorg…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Astorga		<http://dbpedia.org/resource/Astorg…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Dodecanese Islands; …		<http://dbpedia.org/resource/Astyp…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Athens		<http://dbpedia.org/resource/Athens>	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Augsburg		<http://dbpedia.org/resource/Augsb…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Augsburg; Friedberg		<http://dbpedia.org/resource/Augsb…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Augsburg		<http://dbpedia.org/resource/Augsb…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Augsburg		<http://dbpedia.org/resource/Augsb…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Augsburg		<http://dbpedia.org/resource/Augsb…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Augsburg		<http://dbpedia.org/resource/Augsb…	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Austria; Innsbruck		<http://dbpedia.org/resource/Austria>	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Austria		<http://dbpedia.org/resource/Austria>	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Austria		<http://dbpedia.org/resource/Austria>	dcterms:spatial
<http://mint-projects.image.ntua.gr/europeana-…	dcterms:spatial	Austria; Grödener; Eis…		<http://dbpedia.org/resource/Austria>	dcterms:spatial

4334 rows. (Showing first 50)

50 Load

Fig. 13 DBpedia linking results.

E.g. the results for detecting instances of the dbpedia:Place top-label concept in the data for Item X are shown in the following table:

item_id	property	data	lang	URI	category
…1023_0bfe0b0844…	dcterms:spatial	Great Britain		dbpedia:Great_Britain	dcterms:spatial

2.2.5 Enriched/Linked Data Generation Process

This is the final subprocess that constructs the output enriched and linked file item. By taking into account the previous subprocesses, all the information concerning the enriched metadata and their links to external web resources has been encoded into tables. It remains to gather and save this information into an output file in RDF format.

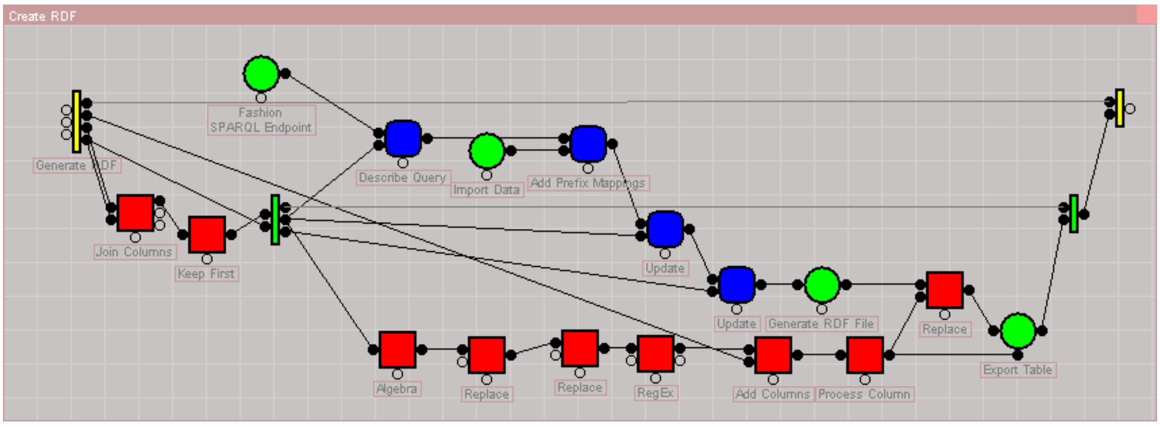


Fig. 14 RDF generation process workflow

More precisely, the process takes as input the tables produced by the two above-described annotation processes for each data item, retrieves from the original RDF store all the RDF triples making up the original description of the particular item, and updates it by adding a new EDM-fp property value for each row of the input table. The result is the enriched/linked RDF/XML file. E.g.

the final output for item `T.127-1933` is the following (the results of the enrichment are shown in italics):

```
<rdf:RDF>
<edm:ProvidedCHO rdf:about="http://mint-projects.image.ntua.gr/europeana-
fashion/1023_0bfe0b0844b9067a57fe6b86ffc06b3264c051f4a14a59ab7bd9f1848fcf157c_2716399_2607_5
55-1893">
        <dc:subject
xml:lang="en">grapes;borage;honeysuckle;carnation;pansy;columbine</dc:subject>
        <edm:type>IMAGE</edm:type>
        <edmfp:technique>hand sewing</ edmfp:technique>
        <dc:description xml:lang="en">Purse, embroidered canvas with silk on silver ground,
plaited silk strings, 1600-1650, English. Linen, silk, silver and silver-gilt threads, silk
thread; hand sewn, hand embroidered, hand plaited.</dc:description>
        <dcterms:created>1600/1650</ dcterms:created>
        <dcterms:medium>silk taffeta</dcterms:medium>
        <dc:title>Purse</dc:title>
        <dcterms:extent>Length 11.7cm, Width 11.3cm (approx., bag only)</dcterms:extent>
        <dcterms:medium>linen</dcterms:medium>
        <dc:date>1600/1650</dc:date>
        <edmfp:technique>plaiting</edmfp:technique>
        <dc:identifier>555-1893</dc:identifier>
        <edmfp:localType>Purse</edmfp:localType>
        <dcterms:medium>silver gilt thread</dcterms:medium>
        <dcterms:medium>silk thread</dcterms:medium>
        <dcterms:spatial>
                <edm:Place><skos:prefLabel>Great Britain</skos:prefLabel></edm:Place>
        </dcterms:spatial>
        <edmfp:technique>hand embroidery</edmfp:technique>
        <dc:type rdf:resource="http://thesaurus.europeanafashion.eu/thesaurus/10140"/>
        <dcterms:medium>silver thread</ dcterms:medium>
        <dcterms:medium rdf:resource=”http://thesaurus.europeanafashion.eu/thesaurus/10372”/>
        <dcterms:medium rdf:resource=”http://thesaurus.europeanafashion.eu/thesaurus/10470”/>
        <dcterms:medium rdf:resource=”http://thesaurus.europeanafashion.eu/thesaurus/10352”/>
        <dcterms:medium rdf:resource=”http://thesaurus.europeanafashion.eu/thesaurus/10547”/>
        <edmfp:technique
rdf:resource=”http://thesaurus.europeanafashion.eu/thesaurus/10372”/>
        <edmfp:technique
rdf:resource=”http://thesaurus.europeanafashion.eu/thesaurus/10428”/>
        <dcterms:spatial>
                <edm:Place rdf:resource=”http://dbpedia.org/resource/Great_Britain"/>
        </dcterms:spatial>
</edm:ProvidedCHO>
</rdf:RDF>
```

3. Name Data Service for Semantic Enrichment

Semantic enrichment in Europeana is a very difficult task due to several factors:

- Varying metadata quality across different collections, sometimes including misallocation of metadata fields
- Varying metadata formatting practices across different collections, e.g. some collections indicate the role of a creator in brackets after the creator name
- Lack of accurate language information

For these reasons, it was decided to focus on Person and Institution enrichment (person Named Entity Recognition), which in itself is an ambitious task.

3.1 Introduction

Historic people are often referred to by many names. For successful semantic enrichment it is important to integrate high-quality and high-coverage datasets that provide name info. There is a great number of Name Authority files maintained at libraries, museums and other heritage institutions world-wide, e.g. VIAF, ISNI, Getty ULAN, British Museum. Linked Open Data (LOD) datasets also have a plethora of names, e.g. in DBpedia, Wikidata and FreeBase. Some of the available datasets in terms of person coverage, name coverage, language tags, extra features that can be useful for enrichment, quality were analysed.

The important topic of **co-referencing** is analysed as well, i.e. how connected the sources are to each other.

The investigation follows a hands-on approach, with detailed descriptions of access methods and tools that were used. A number of accompanying files are provided at http://vladimiralexiev.github.io/CH-names and referenced below.

3.1.1 RDF-based Gazetteers and Data Integration

Enrichment is most often based on large and efficient lookup structures called **gazetteers**. Ontotext's experience with commercial semantic enrichment is that the best way to make gazetteers is:

- Acquire a number of data sources, converting to RDF if needed
- Load the data sources to a single repository, thus integrate them in semantic format. Many of the sources have SPARQL endpoints or proprietary APIs that are very useful for investigation, but usually unworkable for production (see "LOD Cache" below)
- Assemble gazetteers "dynamically" from the RDF sources.
    - In the simplest form this can be done with custom SPARQL queries that extract the data needed for a particular enrichment application. The gazetteer needs to be refreshed periodically when the RDF data changes

    - In more sophisticated forms this could involve dynamic synchronization of gazetteers with the RDF repository upon change

3.1.2 Data Source Dimensions

The sources along several dimensions are investigated:

- Dataset description: where it came from, past evolution, future outlook. All these non-technical aspects are important factors in deciding whether and how to use the source
- Agent coverage: number of people/institutions. For some sources published reports are used and summaries, for others recent counts are used and analyses performed
- Name coverage: number of name forms for a well-known person (found in all sources), then analyse set intersections and unique contributions
- Access: description how the source can be accessed: web and API access for exploration, bulk download for RDF integration, update rate, etc.
- Language coverage: provides overall info about the number of languages covered, but have not performed precise counts
- Extra features that can be useful for enrichment (see next section)
- Quality and accuracy of the various data elements

3.1.3 Extra Features

In addition to person names, the following features can be very useful for disambiguation:

- **Description:** In addition to person names, most data sources have a person description. It can be useful for contextual disambiguation, e.g. to distinguish a painter from a sculptor by the type of object. It can vary between:
    - Short standardized description as provided by ULAN, e.g. "German painter, draftsman, and printmaker, 1472-1553"
    - Short unstandardized description as provided by Wikidata, e.g. "German Renaissance painter and printmaker in woodcut and engraving"
    - Short or long abstract as provided by DBpedia. The long abstract is the beginning of the article until the first heading. The short abstract is the first couple of sentences.
    - Quite long biographies as provided by the British Museum (only for well-known artists)
- **Life years:** These are useful to filter out by date range. e.g. a 20-century painter cannot be the author of a 16-century painting. But life years in CH are often subject to uncertainty, usually expressed with qualifiers like "circa" (c), "early/mid/late Nth century", "floruit" (fl). In such cases rule-of-thumb defaults may be employed, e.g. "a

person cannot live for more than 100 years", "a person is not creatively active before 15 and after 90 years of age", etc.

  - Imprecision: the exact year is not known, e.g. "early 16th century". This can be translated to a range of years
  - Ambiguity: different opinions about the year coming from different sources. In such cases, all opinions are recorded, together with source
  - "Floruit": the birth and/or death date is not known, only a range when the person was active

- **Language tags:** Whether the source has reliable language tags for the names. They can help restrict the candidate names, but only if the text to be enriched also has reliable language info. Unfortunately there is no such info in Europeana
- **Popularity:** Often the "popularity" of an entity has good correlation to the probability of its appearance in text, thus is a good way of ordering candidate matches. The question is how to compute "popularity". For cities, a good approximation is the population. For agents, one could use the connectedness and centrality of the agent in an RDF graph, e.g.:
  - Number of paintings on Wikimedia Commons
  - Number of "influences" connections on DBpedia
  - Number of relations in ULAN
  - Ontotext GraphDB provides a simple measure called RDFRank, which is an implementation of PageRank for RDF graphs.

3.1.4 Running Example: Lucas Cranach

An example-driven approach, analysing name coverage for Lucas Cranach the Elder:

Lucas Cranach is one of the most important painters of the German Renaissance. His name evolved over time. He was born "Lucas Maler": this surname means "painter" and denoted the profession of his father, not his ancestry. Later his surname was "changed" to Cranach, after the name of his birthplace (Kronach in upper Franconia), another custom of the times. (Some Italians called him "Lucas Tedesco", another geographically-derived name). When his son was born he also became a painter (though a less prominent one): then art historians started referring to him as Lucas Cranach the Elder (I), and to his son as Lucas Cranach the Younger (II) to avoid confusion. German Wikipedia even refers to Lucas Cranach III, though there are no known works by this painter.

Such name evolution is quite typical of historic persons, leading to a large number of names.

3.2 Name Sources

3.2.1 Wikidata

Wikidata is an open crowd-sourced database of facts. Wikidata is intended to provide a central **data** store for all Wikipedias, similar to Wikimedia Commons providing a central **media** store. The hope is that such central fact store will take care of an important problem in Wikipedia language editions: that articles about the same entity in different languages may include different claims about the same property, in terms of value, sources, timeliness ("as of"), etc.

Wikidata started with mass-import of information from Wikipedia: inter-language links (corresponding to owl:sameAs statements between DBpedia language editions), labels, basic data such as birth/death years, coordinates, etc. New data is added all the time, both by human editors and automated processes (bots). The quality of data is higher since it is a centralized database ("single source of truth" for each claim) and has stricter editorial process (while each Wikipedia language edition uses its own properties and editorial policies).

3.2.1.1 Wikidata Access

One can access the info about Cranach in various ways:

- Per-entity web page: http://www.wikidata.org/entity/Q191748
- Per-entity semantic format. One can request the entity with content negotiation (Accept header), or with the corresponding file extension. Turtle is the easiest to read (the names use real Unicode chars), NTriples may be easier to compare (it is line-oriented), JSONLD may be easiest to process in applications.

```
curl -L -Haccept:application/rdf+xml
http://www.wikidata.org/entity/Q191748 > cranach-wikidata.rdf
curl -L -Haccept:text/turtle
http://www.wikidata.org/entity/Q191748 > cranach-wikidata.ttl
curl -L -Haccept:text/plain
http://www.wikidata.org/entity/Q191748 > cranach-wikidata.nt
curl -L -Happlication/ld+json
http://www.wikidata.org/entity/Q191748 > cranach-wikidata.jsonld
```

  - This includes only labels and Wikipedia (inter-language) links. Other statements are not yet available by Wikidata entity access
  - To get the best of both worlds (line-oriented and real Unicode chars), reprocess the Turtle with Jena rdfcat. (Note: rdfcat does not produce proper Unicode from the NTriples file):

```
rdfcat -out ntriple cranach-wikidata.ttl | sort > cranach-
wikidata1.nt
```

- Reasonator application, which collates a lot of useful info in a pretty way: http://tools.wmflabs.org/reasonator/?&q=191748
- Wikidata API (Reasonator is built using it)
- DBpedia SPARQL endpoint: http://dbpedia.org/sparql.
    - In DBpedia the entity URL is rewritten to http://wikidata.dbpedia.org/resource/Q191748
    - You can get the info with a query like this, since the rewritten URL does not resolve

      ```
      describe <http://wikidata.dbpedia.org/resource/Q191748>
      ```

    - Compared to Wikidata there may be some more info (especially for less popular items), but it is less precise/accurate
    - It is not clear where this data came from, or how often it is updated

Wikidata also provides a number of powerful tools that are described next.

3.2.1.2 Wikidata Query

Wikidata Query (WDQ) is a very peculiar but very powerful query language. The implementation caches large amounts of key data, so query answering is very fast.

- WDQ API: http://wdq.wmflabs.org/
- WDQ Documentation: http://wdq.wmflabs.org/api_documentation.html, with executable examples
- WDQ editor UI: http://wdq.wmflabs.org/wdq/, with editable examples

WDQ is multifunctional. The links above can be used for testing. Queries can be loaded into the WDQ editor.

- number of Humans: P31 "instance of" is Q5 "human": 2690452

  ```
  http://wdq.wmflabs.org/api?q=CLAIM[31:5]&noitems=1
  ```

- number of subclasses of Human: start from Q5, go backward along P279 "subclass of": 121 (some of them quite ad-hoc and weird)

  ```
  http://wdq.wmflabs.org/api?q=TREE[5][][279]&noitems=1
  ```

- number of instances of Human or subclasses thereof: 2690504 (it is good that there are almost no instances of the ad-hoc classes)

  ```
  http://wdq.wmflabs.org/api?q=CLAIM[31:(TREE[5][][279])]&noitems=1
  ```

- number of items with VIAF id (P214): 504912

  ```
  http://wdq.wmflabs.org/api?q=CLAIM[214]&noitems=1
  ```

- number of Humans (Q5) with VIAF id (P214): 489705 (97% VIAF items, but only 18.2% of all Humans)

```
http://wdq.wmflabs.org/api?q=CLAIM[31:5]+and+CLAIM[214]&noitems=1
```

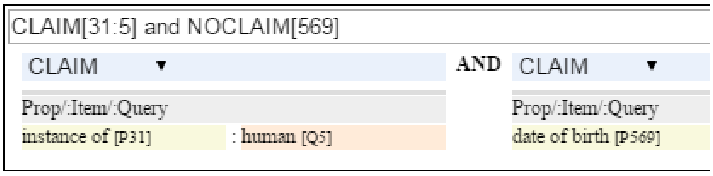
- non-Humans with VIAF id: returns nothing, which is strange/inconsistent

```
http://wdq.wmflabs.org/api?q=NOCLAIM[31:5]+and+CLAIM[214]
```

- number of Humans with missing birth date (P569): 1049808 (39%)

```
http://wdq.wmflabs.org/api?q=CLAIM[31:5]+and+NOCLAIM[569]&noitems=1
```

The query can be passed into the WDQ editor to understand it better:



CLAIM[31:5] and NOCLAIM[569]

CLAIM ▾ AND CLAIM ▾

Prop/:Item/:Query — instance of [P31] : human [Q5]

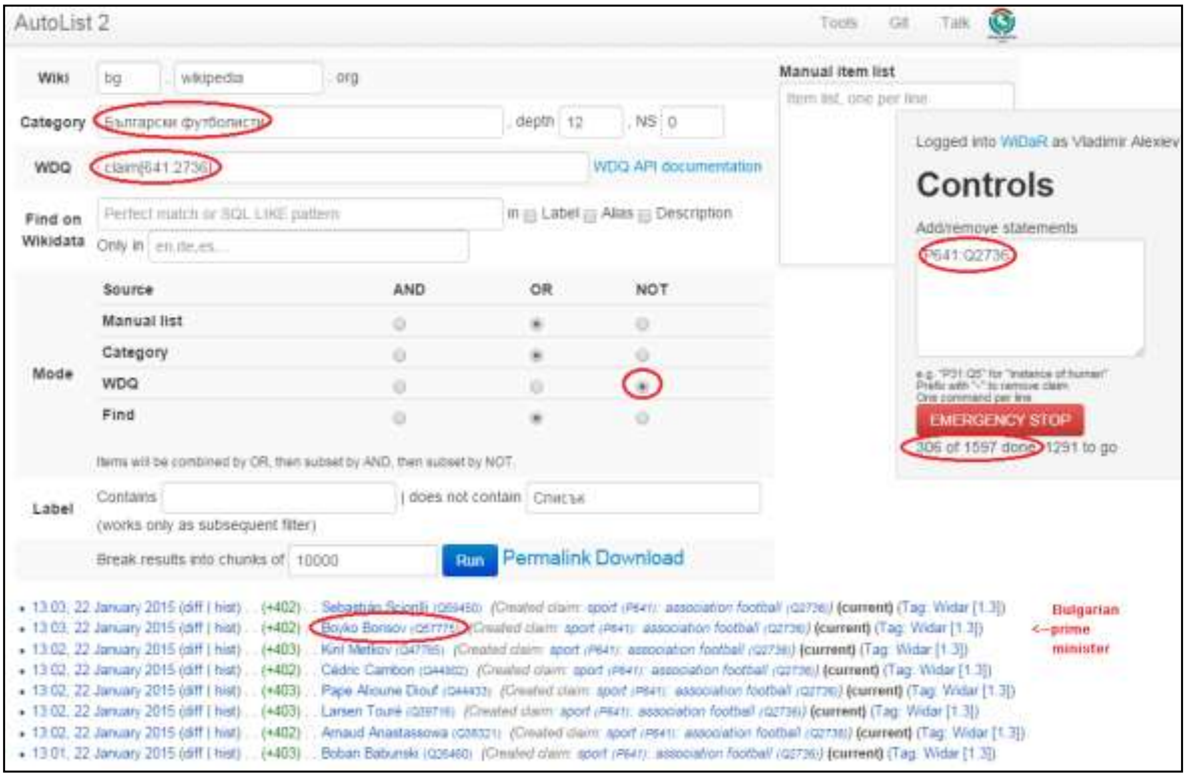
Prop/:Item/:Query — date of birth [P569]

3.2.1.3 Wikidata

AutoList2 (http://tools.wmflabs.org/autolist/) is a powerful tool that allows you to:

- Query any-language Wikipedia by category
- Query Wikidata by WDQ
- Query Wikidata labels (prefLabel), aliases (altLabel) by exact or substring
- Adjust with a manual list
- Combine these with boolean connectives
- Bookmark or download the results
- Apply any claims (statements) to the final result

Below is an example: One can take all articles on bg.wikpedia in category "Български футболисти" (Bulgarian soccer players), look for ones with missing claim "sport=association football" and add such claim. (This includes non-professional soccer players, e.g. Bulgarian prime minister Boyko Borissov.) This tool allows even people without MediaWiki bot programming experience to do batch-updates.



Wiki: bg . wikipedia . org

Category: Български футболисти , depth 12 , NS 0

WDQ: claim[641:2736] WDQ API documentation

Find on Wikidata: Perfect match or SQL LIKE pattern in □ Label □ Alias □ Description
Only in en,de,es,...

Mode:

Source	AND	OR	NOT
Manual list	○	●	○
Category	○	●	○
WDQ	○	○	●
Find	○	●	○

Items will be combined by OR, then subset by AND, then subset by NOT.

Label: Contains | does not contain Списък
(works only as subsequent filter)

Break results into chunks of 10000 Run Permalink Download

Manual item list: Item list, one per line

Logged into WiDaR as Vladimir Alexiev

Controls

Add/remove statements
P541:Q2736

e.g. "P31:Q5" for "instance of human"
Prefix with "-" to remove claim
One command per line

EMERGENCY STOP

306 of 1597 done 1291 to go

- 13:03, 22 January 2015 (diff | hist) . . (+402) . . Sebastián Sciorilli (Q59450) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])
- 13:03, 22 January 2015 (diff | hist) . . (+402) . . Boyko Borisov (Q57775) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])
- 13:02, 22 January 2015 (diff | hist) . . (+403) . . Kiril Metkov (Q47765) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])
- 13:02, 22 January 2015 (diff | hist) . . (+402) . . Cédric Cambon (Q44802) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])
- 13:02, 22 January 2015 (diff | hist) . . (+403) . . Pape Alioune Diouf (Q44433) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])
- 13:02, 22 January 2015 (diff | hist) . . (+403) . . Larsen Touré (Q29716) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])
- 13:02, 22 January 2015 (diff | hist) . . (+402) . . Arnaud Anastassowa (Q28321) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])
- 13:01, 22 January 2015 (diff | hist) . . (+403) . . Boban Babunski (Q26460) (Created claim: sport (P641): association football (Q2736)) (current) (Tag: Widar [1.3])

Bulgarian
<--prime
minister

Another example:

- List of all paintings (first 100). WDQ query "claim[31:3305213]". Total results: 34147
- List of paintings with image (on Commons) (first 100). WDQ query "claim[31:3305213] and claim[18]". Total results: 16139

3.2.1.4 Wikidata Generic Tree

Wikidata Tree (http://tools.wmflabs.org/wikidata-todo/tree.html) allows you to view the class hierarchy (or any other property tree), e.g.:

- Subclasses of Person (560): http://tools.wmflabs.org/wikidata-todo/tree.html?q=Q215627&rp=279
  - Note: Wikidata uses "human" for people, and "person" for anything that can have a personality, e.g. deity, artificial agent, etc.
- Subclasses of Location (3234): http://tools.wmflabs.org/wikidata-todo/tree.html?q=Q17334923&rp=279 The class hierarchy is currently quite a mess. Luckily, the direct types used for Humans and Organizations are not too many, and are ok
- Locations in Cambridgeshire as a dhttp://tools.wmflabs.org/wikidata-todo/tree.html?q=23112&rp=131&method=d33 star tree:

3.2.1.5 Wikidata Stats

- Wikidata Statistics shows the number of items (Content pages) and editorial statistics
- Stats tables and charts shows the growth since Feb 2013
- Year in Review shows a breakdown of items per number of labels and number of statements comparing Jan 2014 and Jan 2015
- Live stats provides up to date information on Wikidata size, number and percentage of statements of different kinds, and the WDQ clause used to access this kind of data element

data element	count	percent	clause
items	13116549		
labels	63086181		
sitelinks	41936042		link
strings	12834528	23.5	string
monolingual_strings	4255	0.0	
times	2786663	5.1	between
coordinates	1893742	3.5	around
connections	36772502	67.4	claim
quantities	294977	0.5	quantity
total statements	54586667	100.0	

Comparing the Live numbers to the triples in the next section:

- Labels=63M would leave 81.7M to descriptions & aliases, but in our opinion these are fewer than the labels
- Sitelinks=42M is only 30% of the number reported above? The correct count is not available, since WDQ does not return accurate results for empty `link[]` or `nolink[]` clauses. 4.4M items have link to enwiki or dewiki, and 8.7M do not have such links: one can estimate that 6M items have any wikilink, and the other 7M do not
- Total number of statements (54M) does not even reach the number of "simple statements"
- The percentage breakdown of statements gives a useful overview of the kind of data in Wikidata at present

3.2.1.6 Wikidata Download and RDF Counts

Wikidata provides comprehensive RDF data dumps: http://tools.wmflabs.org/wikidata-exports/rdf/exports/

- There is some discussion of implementing Incremental dumps (similar to http://live.dbpedia.org), but such are not yet available

- The full dumps are made quite often (monthly or bimonthly)
- Note: the interactive query tools described above work with a delay of 5-15 minutes compared to the live data

wikidata-?.nt.gz	Triples	size	description
terms	144702568	1.2G	item labels, descriptions, aliases (in all languages)
sitelinks	140980119	1.0G	links from Wikidata to Wikipedia and other MediaWiki project sites
simple-statements+	81086253	607.0M	one triple per statement: references omitted, statements with qualifiers not included
properties+	74510	1.4M	property definitions, including datatypes, labels, descriptions, aliases
taxonomy+	335334	1.5M	class hierarchy: "subclass of" with no qualifiers -> rdfs:subClassOf (1)
instances+	12331117	52.6M	class membership information: "instance of" with no qualifiers -> rdf:type
statements	220633163	2.9G	statements/claims, complete with references and qualifiers

(1) And items used as target of "subclass of" or "instance of" -> owl:Class

Wikidata statements (claims) may carry complex associated information in **qualifiers**, such as dates of applicability, source references, scope ("of"), etc. Such claims are exported to RDF in a complex way using reification: see [5] fig.3 and sec.3.2.

- The last file "statements" in the table above uses this complex modelling and is quite hard to work with.
- The files marked "+" are derived from "statements". They are quite simpler to work with, and also smaller.

3.2.1.7 Wikidata Coverage and Type Count

Ontotext has taken a recent count of all direct type ("instance of") RDF statements as of Dec 2014. The count files are on Gist

- There are 12331093 "instance of" statements. Wikidata has 13M items, so about 93% of all items have types (if it is assumed that only a small percent of the items have multiple types)
- There are 17875 classes with at least one instance, of which 6510 classes (36%) with at least 5 instances. The rest (64%) are a very long tail of items that are inappropriately used as classes, e.g. Indian Rhinoceros, Trumbull's Declaration of Independence, stud, meatloaf…

Specific classes, which are useful for Person/Organisation Recognition:

- There are 2.7M (2662626) **humans** (matches the number reported by WDQ Wikidata Query). This is fairly well focused, in that it collects a large proportion of all humans. There are a few exceptions, e.g. "minister", "table tennis player", "chess composer": these should be used as "occupation" while "instance of" should be "human".
- There are 5k **families**: 4569 noble family, 635 family, 465 Dutch noble family, 95 Belgian noble family, 35 clan
- There are some 22k **literary characters**: 11993 fictional character, 6963 fictional human, 2589 mythical character, 357 group of fictional characters, 159 fictional organization
- There are at some 215k **organisations** (not counting governments, city councils, etc.). These are spread across a wide list of classes, so the totals below are not comprehensive and represent the possible minimum:
    - 55k **businesses**: 47149 company, 2653 business, 2321 transport company, 885 public company, 718 corporation, 152 motorcycle manufacturer, 95 joint-stock company, 80 holding company
    - 66k **creative organizations** 42179 band, 17904 radio station, 6187 newspaper, 1540 film production company, 843 theatre company, 22 theatre troupe
    - 31k **sports clubs**: 26200 association football club, 5376 sports club, 184 American football club, 169 golf club, 154 country club
    - 30k **educational institutions**: 16611 high school, 6396 school, 6321 university, 1062 Engineering College, 771 college, 301 research institute
    - 20k **non-profit organisations**: 8929 organisation, 7026 political party, 2853 association, 1052 nonprofit organisation, 307 international organisation, 246 charitable organisation, 226 Esperanto organisation, 144 political organisation, 73 non-governmental organisation
    - 13k **GLAM orgs**: 438 art gallery, 83 art gallery; 882 library, 199 national library, 114 public library, 60 library, 28 Carnegie library, 27 academic library, 16 municipal library; 108 archive, 26 cantonal archives, 24 municipal archive; 6516 museum, 2176 art museum, 873 military museum, 569 museum ship, 513 historic house museum, 181 maritime museum, 151 musée de France, 119 aviation museum, 80 natural history museum, 68 science museum, 57 open-air museum, 48 railway museum, 37 local museum, 37 children's museum

In addition, the following types may be interesting:

- There are 40k+20k **names**: 40038 family name, 10320 given name, 5569 male given name, 4828 female given name.
  - Due to the good efforts of the WikiProject "Wikidata names", these items provide valuable information on names themselves, e.g. variations, male/female correspondences, etc.
  - This can probably be used for disambiguation or for generating language-specific name variants, but we have not investigated this topic
- Some 500k **Creative Works**: 154125 album, 140820 film, 59242 single, 51765 book, 31623 painting, 23055 scientific journal, 20032 song, 26789 video game, 18338 television program, 14838 short film, 13461 television series, 13098 silent film, 11876 periodical literature, 11297 episode, 6739 literary work, 6627 television season, 3488 sculpture, 2374 manuscript
- Some 110k **heritage sites and monuments**: 64806 Rijksmonument, 21076 Iranian National Heritage, 19696 scheduled monument, 1370 natural monument, 1150 World Heritage Site. This is expected to grow sharply for other countries as well.

The link given above also reports various defective classes.

3.2.1.8 Wikidata Names

Now it can be checked what person names (labels) are provided in Wikidata.

- Preferred names are repeated as rdfs:label, skos:prefLabel, schema:name
- Alternate names are in skos:altLabel

Some of the original strings differed only by punctuation, e.g.

- Lucas Cranach "el Vell" **vs.** Lucas Cranach el Vell **vs.** Lucas Cranach, "el Vell"
- Lucas Cranach o Velho **vs.** Lucas Cranach, o Velho
- Кранах Лукас Старший **vs.** Кранах, Лукас Старший

The comma is often used to indicate **last, first** name inversion (a variant used "for indexing"). But one cannot rely on it:

- "Lucas Cranach, o Velho" shows the comma is sometimes used for other purposes
- "Кранах Лукас Старший" shows the comma is not consistently applied to name inversion

So we removed the punctuation chars ,." and ended up with 70 Wikidata name forms for Cranach: ./cranach-wikidata.txt. Examine the file to get a feel for the names.

3.2.1.9 Wikidata Languages

Wikidata includes names in a variety of languages.

- Lang tags are included for all languages, which is valuable

There are 57 unique lang tags, representing 44 languages and 13 language variants (e.g. de=German vs. de-ch=Swiss German):

- af arz az be be-tarask bg br ca cs da de de-ch el en en-ca en-gb eo es eu fa fi fr ga hu hy it ja ka ko la lt lv mk nb nl nn pl pt pt-br ro ru sh sr sr-ec sr-el stq sv sw th uk zh zh-cn zh-hans zh-hant zh-hk zh-sg zh-tw

Only 3 of the language variants are truly distinct:

- zh-hans (Chinese Han Simplified) vs. zh-hant (Chinese Han Traditional); be (Belorussian) vs. be-tarask (Belorussian Taraskevica); sr=sr-ec (Serbian Cyrillic) vs. sr-el (Serbian Latin)

The other language variants carry the same name string, e.g.:

- en, en-ca, en-gb; de, de-ch; pt, pt-br; zh, zh-cn, zh-sg

Observations on prefLabel and altLabel:

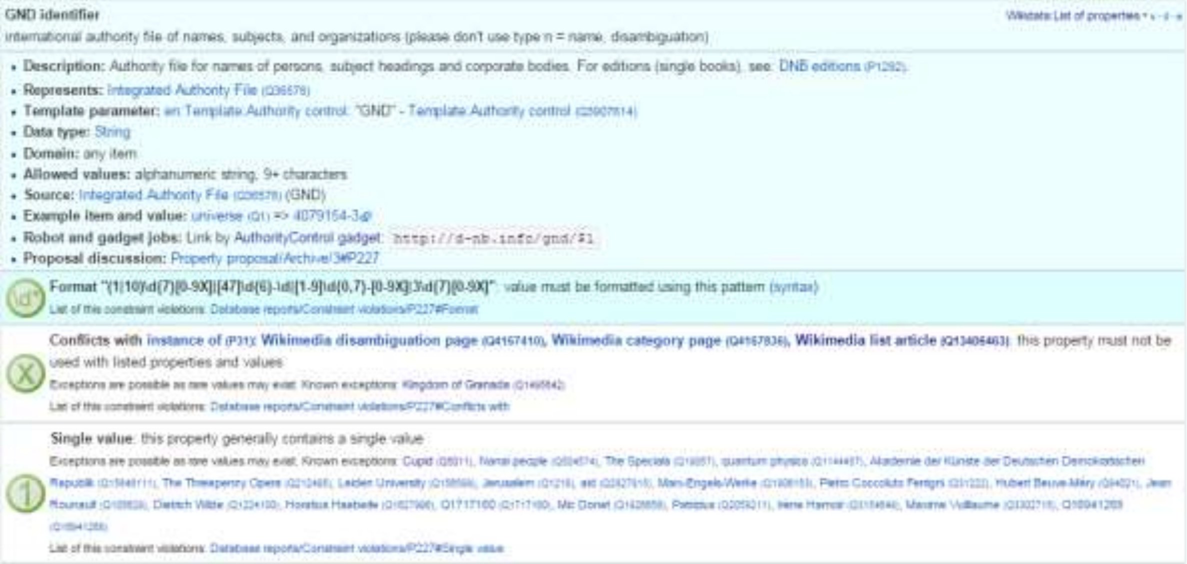
- There is a single prefLabel per language, following SKOS recommendations
- If the lang tag is taken into account, prefLabels and altLabels are disjoint
- But if you discount the lang tag, many of the altLabels are redundant. e.g. the German prefLabel "Lucas Cranach der Ältere"@de is repeated as altLabel for languages: lt lv nl pt stq sv.
- If you discount the lang tag, some of the prefLabels are also redundant

3.2.1.10 Wikidata Quality

Ontotext started using Wikidata in commercial applications since mid-December 2014, so the quality of different data elements can be estimated:

- Labels (names) are almost universally good
- Descriptions are sensible, though short, not authoritative, and often missing. Descriptions can be used only to disambiguate two items with the same name, but not to provide information about the item
- Linkage to different Wikipedias, Wikimedia Commons and other Wikimedia sites is always good.
- Direct types ("instance of") are accurate for most of the entities in Wikidata Coverage and Type Count

- The set of properties is good. There are established property proposal editorial practices, based on a detailed proposal template followed by discussion and "voting". e.g. see properties for Authority Control). If after some time there are some supporters, but no or very few opponents, the property is created only by a property creator or an administrator. All discussion, decisions and their rationale are preserved. Defined property metadata is collected, including guidelines for use (e.g. on what items it should be applied), to which register or authority file it corresponds (if any), examples, format validation, uniqueness constraints, lists of known exceptions, etc. e.g. see GND identifier.



GND identifier

international authority file of names, subjects, and organizations (please don't use type n = name, disambiguation)

Wikidata:List of properties • v · d · e

- **Description:** Authority file for names of persons, subject headings and corporate bodies. For editions (single books), see: DNB editions (P1292).
- **Represents:** Integrated Authority File (Q36578)
- **Template parameter:** en Template:Authority control: "GND" - Template:Authority control (Q5907814)
- **Data type:** String
- **Domain:** any item
- **Allowed values:** alphanumeric string, 9+ characters
- **Source:** Integrated Authority File (Q36578) (GND)
- **Example item and value:** universe (Q1) => 4079154-3
- **Robot and gadget jobs:** Link by AuthorityControl gadget: `http://d-nb.info/gnd/$1`
- **Proposal discussion:** Property proposal/Archive/3#P227

Format "(1|10)\d{7}[0-9X]|[47]\d{6}-\d|[1-9]\d{0,7}-[0-9X]|3\d{7}[0-9X]": value must be formatted using this pattern (syntax)

List of this constraint violations: Database reports/Constraint violations/P227#Format

Conflicts with instance of (P31): **Wikimedia disambiguation page** (Q4167410), **Wikimedia category page** (Q4167836), **Wikimedia list article** (Q13406463): this property must not be used with listed properties and values

Exceptions are possible as rare values may exist. Known exceptions: Kingdom of Granada (Q149584)

List of this constraint violations: Database reports/Constraint violations/P227#Conflicts with

Single value: this property generally contains a single value

Exceptions are possible as rare values may exist. Known exceptions: Cupid (Q5011), Nanai people [illegible], The Specials [illegible], quantum physics [illegible], Akademie der Künste der Deutschen Demokratischen Republik [illegible], The Threepenny Opera [illegible], Leiden University [illegible], Jerusalem (Q1218), ast [illegible], Marx-Engels-Werke [illegible], Pietro Coccoluto Ferrigni [illegible], Hubert Beuve-Méry [illegible], Jean Rouault [illegible], Dietrich Wilde [illegible], Horatius Haeberle [illegible], Q1717100 [illegible], Mc Donel [illegible], Patepius [illegible], Irene Harnoir [illegible], Maxime Vuillaume [illegible], Q10941269 [illegible]

List of this constraint violations: Database reports/Constraint violations/P227#Single value

- These constraints are used to discover violations, which can drive co-referencing and merging/splitting investigations. e.g. see violations for GND identifier.
- Nevertheless, the overall property design is still in flux. e.g. there is a current proposal to eliminate a number of properties such as place/date of birth/death/burial and replace them with a generic "significant event" where details are provided with qualifiers.
- The class hierarchy is not good at all. The reason is that there is no editorial control over "instance of" and "subclass", so anyone can "make" a class. 63% of all classes have fewer than 5 instances. Play with Wikidata Generic Tree to see some very idiosyncratic classes, and a messed up hierarchy. Just a couple of examples:

  location> geographic location> facility> laboratory> lab-on-a-chip

  > But "lab-on-a-chip" is a "device that integrates one or several laboratory functions on a single chip of only millimetres to a few square centimetres in size", hardly a "geographic location"

location> storage> data storage device> audio storage device> album

> Any NER implementor will balk at "albums are locations". The everyday understanding of "location" as "place" is implemented as the subclass "geographic location". But nevertheless, an "album" is a creative work, and as such is a conceptual object that persists even after all its copies are destroyed. It is definitely not a "storage device"

3.2.1.11 Wikidata Synchronization to Wikipedia

This is a summary of some important points about the future data freshness of Wikidata

- Most Wikidata data (labels and links) was originally extracted from Wikipedia
- Wikipedia inter-language links are maintained in Wikidata, and are therefore authoritative in Wikidata
- The idea is that all Wikipedias will gradually transition to using data from Wikidata. However, this is still long coming
- Articles are added to Wikipedia all the time and names are added/edited, and similarly items are added and labels are edited in Wikidata. This can lead to de-synchronisation between the two
- There are bots that can transfer Articles and names from Wikipedia to Wikidata, but it has not been investigated whether that happens regularly, especially for minor-language editions.
- In the converse direction, tools to create a Wikipedia article stub from Wikidata are not yet aware

3.2.2 Freebase

Freebase is a collaboratively edited knowledge base, quite similar to Wikidata but with some more sophisticated features.

It was created by MetaWeb in 2007 and purchased by Google in 2010. It was used in the Google Knowledge Graph, together with Big Data provided by other companies. It is an important dataset that has been used in various applications, including commercial ones. In some sense it has provided inspiration to Wikidata.

On December 16, 2014, the Google Knowledge Graph Team announced that Freebase will be retired. The plan is to transfer the Freebase data to Wikidata (complementing with an application that can help editors to provide source references), stop write Freebase access at end-March 2015, and retire Freebase end-June 2015.

Some investigation of Freebase has been done, but following this announcement it was decided that Freebase data will not be loaded.

3.2.2.1 Freebase Access and Names

- The Freebase URL for Cranach is http://www.freebase.com/m/0kqp0.
- An "almost Turtle" file is available at http://rdf.freebase.com/m/0kqp0, but some fixes are needed:
  - Replace hex escape sequences \x in literals with unicode escape sequences \u00
  - Replace dollar escapes in URLs with proper URL escaping
  - Replace the quotes surrounding literals ("…") with triple quotes """…""" since some literals include quotes
- Freebase provides 32 names for Cranach (./cranach-freebase.txt), all with language tags

3.2.3 DBPedia

DBpedia is structured information extracted from Wikipedia and is the centre of the Linked Open Data cloud. It was first released in Jan 2007 and has been continuously improved ever since.

3.2.3.1 DBpedia Stats

Article [1] presents very comprehensive statistics (p.12 table 2). The most recent version of these statistics is online. EN DBpedia being the first and largest language editions is taken as “Canonicalized Data” ("CD") (namespace http://dbpedia.org/resource; there is no namespace http://en.dbpedia.org/resource). Other editions are called “Localized Data” ("LD"). EN DBpedia provides the following number of entities:

- 1,445k persons
- 735k places
- 241k organisations
- 411k creative works: 123k music albums, 87k films, 19k video games…
- 252k species; etc.
- 4,584k total

The total number across editions is harder to calculate since it depends on the degree of cross-language overlap popular entities appear in many editions, while purely "local heroes" may appear in a single edition. The numbers grow to:

- 1,471k persons
- 818k places
- 266k organisations
- 462k creative works
- 279k species; etc.

It is worth to compare to Wikidata Counts in section Wikidata Coverage and Type Count It is estimated that national editions add 15% more entities and perhaps 50% more labels (names).

3.2.3.2 DBpedia Quality

To understand the dynamics of DBpedia, one should understand raw properties vs. mapped properties and classes, which is described really well in [1]. In brief, the process is as follows:

- Extracts all properties from all significant templates applied to the article. These properties are different for every language edition and are spelled in the national language, so they are called **raw**. Various heuristics are applied to recognise dates, numbers, links. No type information is applied here, which leads to some problems, e.g.:
    - The name of the asteroid 1111 Reinmuthia is extracted as dbpprop:name 1111 (xsd:integer) because of a heuristic "if the field starts with an integer, assume it is an integer"
    - A template field like

```
firstAscent = [[John Smith]], [[England|English]] expedition [[1 May]] [[1941]
firstAscent = in [[Prehistory]]
```

    will extract resources of variegated types: person, country, notable month-day, notable year, and historic period.

- Extracts a number of other characteristics, e.g. all used templates and categories, links, redirects, abstract (text before the first heading), geographic coordinates, etc.
- Reads crowd-sourced class and property definitions and mappings from http://mapping.dbpedia.org
- Computes **mapped** properties from the raw properties and mappings. There is no editorial process in the mapping wiki, so there are significant defects, especially for languages other than English. This involves:
    - Classes, e.g. nonsensical class like VicePresident
    - Properties, e.g. DBpedia has no less than 86 "name" properties of which about half should be eliminated

- Mappings. The problems here are most extensive and vary from non-standard properties (e.g. sex="a" on bg.dbpedia to indicate Female) to mixing the predecessors/successors of a public official across several terms (pl.dbpedia)

Because domains and ranges are not used when extracting raw properties nor checked when mapping, this leads to data problems. e.g. the `firstAscent` template property (see above) is mapped to two:

- `firstAscentYear a owl:DatatypeProperty; rdfs:range xsd:gYear`
  - Will get value `0001` since that's the first number that appears (instead of 1941)
- `firstAscentPerson a owl:ObjectProperty; rdfs:range Person`
  - Will get values `dbr:John_Smith, dbr:England, dbr:1_May, dbr:1941, dbr:Prehistory`, of which only 1 is a Person!
  -

3.2.3.3 DBpedia Class Errors

Mapping problems also lead to class errors. For example:

- dbr:United_Nations has type dbo:Country instead of dbo:Organisation
  - On enwiki United_Nations uses Infobox_Geopolitical_organization
  - The mapping Infobox_Geopolitical_organization has mapToClass = Organisation
  - however the template Infobox_Geopolitical_organization on enwiki is redirected to Infobox_Country.
  - So the mapping Infobox_Geopolitical_organization is disused, but the mapping wiki does not warn about it
  - We need to merge the mapping Infobox_Geopolitical_organization into the mapping Infobox_Country, discriminating on some field (e.g. `org_type`) whether to emit class Organisatin, GeopoliticalOrganization or Country. See more details in discussion
- **bgdbr:Лили_Иванова**, the icon of Bulgarian pop music with 53 years on stage and still going, until recently was mapped to Band (and thus Organisation) instead of MusicalArtist (and thus Person). The reason is that the mapping Музикален_изпълнител (Musical Artist) mapped all cases to Band. Now between several cases is distinguished (translated here from BG to EN for easier understanding):
  - if "members", "former members", or "established" is set -> Band
  - if "background" is "quartet", "ensemble", "choir" -> Band
  - if "background" is "composer" -> MusicComposer

  - if "background" is "director" -> MusicDirector
  - if "background" is "she-singer" -> MusicalArtist & gender = Female
  - if "background" is "he-singer" -> MusicalArtist & gender = Male
  - if "suffix" is "a" -> MusicalArtist & gender = Female
  - else -> MusicalArtist & gender = Male

There are **a lot of cases** like this that need to be investigated and resolved.

3.2.3.4 DBpedia Potential Improvements

Discrepancies in type, gender, agenthood have serious negative impact on Enrichment.

These problems have seen a lot of attention lately, see forum and tracker

- The formation of a DBpedia Ontology Committee is foreseen
- This will be one of the important points for the upcoming DBpedia meeting February 9, 2015 in Dublin, Ireland; with topics like:
  - Break Out Session 3: The new DBpedia Ontology
  - DBpedia Ontology and Extractor Problems
  - DBpedia in Web Protege, by Alexandru Todor
  - Discussion on the new ontology editing workflow and future directions of the DBpedia ontology

3.2.3.5 DBpedia Downloads

The latest download was extracted in August and September 2014. This includes directories for 124 language editions:

- af als am an ar arz ast az ba bat_smg be be_x_old bg bn bpy br bs bug ca ce ceb ckb cs cv cy da de el en eo es et eu fa fi fr fy ga gd gl gu he hi hr ht hu hy ia id io is it ja jv ka kk kn ko ku ky la lb lmo lt lv map_bms mg mk ml mn mr ms my mzn nap nds ne new nl nn no oc pa pl pms pnb pt qu ro ru sa sah scn sco sh si simple sk sl sq sr su sv sw ta te tg th tl tr tt uk ur uz vec vi vo wa war yi yo zh zh_min_nan zh_yue

Notes:

- "simple" is a kind of English, used in the Simple English Wikipedia, where articles are written with a repertoire of a couple thousand words only
- "commons" is an extract from Wikimedia Commons, which includes metadata for 15M freely reusable images, diagrams and multimedia
- "links" provides cross-references to various other datasets

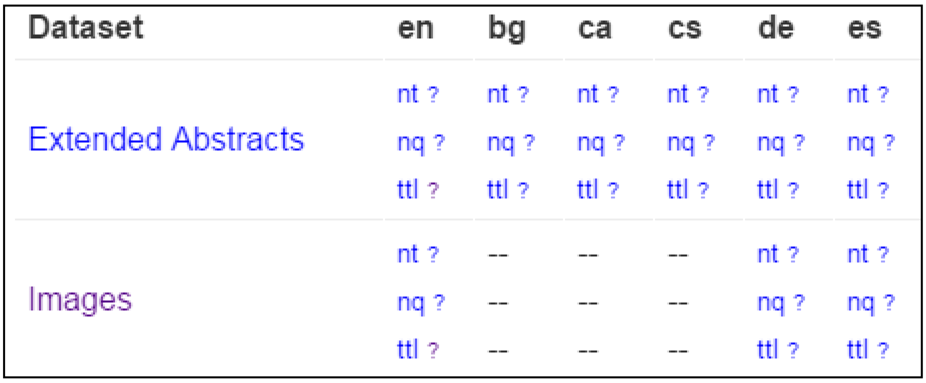
If you look at one of the editions e.g. EN, you'll see a daunting picture: 162 files of size 37.6Gb zipped. But they come in quadruples, e.g.

labels_en.nq.bz2	Encoded URIs. Quads: each statement has the wikipedia line that generated it
labels_en.nt.bz2	Encoded URIs
labels_en.tql.bz2	International IRIs. Quads: each statement has the wikipedia line that generated it
labels_en.ttl.bz2	International IRIs

If the triplestore can handle Unicode IRIs and this fine-grained provenance is not relevant, one can use the last one (ttl) only.

An excellent description of the downloads is available, although a few of the files are not listed there.

- It presents the files in a logical sequence and has some description
- There is a preview of each file: the first 100 lines, anchored at "?".
- It shows at a glance which files are not available for download for a particular language, e.g.



Dataset	en	bg	ca	cs	de	es
Extended Abstracts	nt ? nq ? ttl ?	nt ? nq ? ttl ?	nt ? nq ? ttl ?	nt ? nq ? ttl ?	nt ? nq ? ttl ?	nt ? nq ? ttl ?
Images	nt ? nq ? ttl ?	-- -- --	-- -- --	-- -- --	nt ? nq ? ttl ?	nt ? nq ? ttl ?

For example, images (links from DBpedia resources to Commons images) were missing fo BG. But they are important for bg.dbpedia, we took care to generate them.

A rather unique feature of DBpedia is DBpedia Live. It can provide RDF updates tracking the minutely edits on Wikipedia, Wikipedia infoboxes, and the Mapping wiki too. A stream of changes is generated and a Synchronization Tool is provided, which makes it easier to deploy a continuously updating RDF server.

DBpedia Loaded Languages

The datasets loaded on dbpedia.org include:

- 27 en files: article_categories category_labels disambiguations external_links freebase_links geo_coordinates geonames_links_en homepages images infobox_properties infobox_property_definitions instance_types instance_types_heuristic interlanguage_links_chapters iri_same_as_uri labels long_abstracts mappingbased_properties_cleaned page_ids persondata redirects_transitive revision_ids revision_uris short_abstracts skos_categories specific_mappingbased_properties wikipedia_links
- labels, short and long abstracts in the following additional 11 languages:
  - ar, de, es, fr, it, ja, nl, pl, pt, ru, zh
- 37 linkset files to external datasets, including opencyc, umbel, yago

Names found in a language edition are not necessarily limited to that language.

Unfortunately, DBpedia lang tags on fields other than rdfs:label are sometimes missing or unreliable. The reason is that some national mappings do not specify a language tag adequately.

DBpedia sameAs

Just like Wikipedia, DBpedia has different language editions. The inter-language links generate owl:sameAs statements across the editions. The query is tried on http://dbpedia.org/sparql:

```
select * {dbpedia:Lucas_Cranach_the_Elder owl:sameAs ?x}
```

Note: although sameAs is supposed to be symmetric (actually an equivalence), this returns more results than the following query:

```
select * {?x owl:sameAs dbpedia:Lucas_Cranach_the_Elder}
```

This returns results like

```
http://rdf.freebase.com/ns/m.0kqp0
http://wikidata.org/entity/Q191748
http://wikidata.dbpedia.org/resource/Q191748
http://yago-knowledge.org/resource/Lucas_Cranach_the_Elder
http://sw.cyc.com/concept/Mx4rvXh1w5wpEbGdrcN5Y29ycA

http://af.dbpedia.org/resource/Lucas_Cranach_die_Ouere
http://arz.dbpedia.org/resource/لوكاس_كراناك_الاكبر
http://az.dbpedia.org/resource/Lukas_Kranax_(böyük)
http://be.dbpedia.org/resource/Лукас_Кранах_Старэйшы
http://be_x_old.dbpedia.org/resource/Люкас_Кранах_Старэйшы
http://bg.dbpedia.org/resource/Лукас_Кранах_Стари
```

See ./dbpedia-sameas.txt for the full set of owl:sameAs for Cranach.

- The first few are links to Freebase, Wikidata (one correct URL and another "bastardized" by DBpedia), Yago Knowledge and Open Cyc

- The rest are the interlanguage links.

The sameAs do not return extra data on http://dbpedia.org, e.g.:

```
select * {<http://de.dbpedia.org/resource/Lucas_Cranach_der_Ältere> ?p ?o}
select * {<http://bg.dbpedia.org/resource/Лукас_Кранах_Стари> ?p ?o}
```

The labels and abstracts in the 11 additional languages are attached to the en URLs.

3.2.3.6 Wikipedia Redirects

Wikipedia redirect page goes to the target of the redirect. e.g. http://en.wikipedia.org/wiki/Cranach,_Lucas_the_Elder goes to the page about Cranach. A redirect may point to another redirect, but the DBpedia extractor chases all redirects to their ultimate target.

- DBpedia implements a similar redirect: http://dbpedia.org/resource/Cranach,_Lucas_the_Elder goes to the DBpedia resource/page about Cranach

However, DBpedia also includes statements that we can use:

```
select * {?x dbpedia-owl:wikiPageRedirects+
dbpedia:Lucas_Cranach_the_Elder}
```

returns all EN redirects for Cranach, which are:

```
http://dbpedia.org/resource/Cranach,_Lucas_the_Elder
http://dbpedia.org/resource/Cranach_the_Elder
http://dbpedia.org/resource/Lucas,_the_Elder_Cranach
http://dbpedia.org/resource/Lucas_Cranach,_Sr.
http://dbpedia.org/resource/Lucas_Cranach_der_%C3%84ltere
http://dbpedia.org/resource/Lucas_Cranach_der_Aeltere
http://dbpedia.org/resource/Lucas_Cranach_der_Altere
http://dbpedia.org/resource/Lucas_Cranach_the_elder
http://dbpedia.org/resource/Lucas_Muller
http://dbpedia.org/resource/Lucas_the_Elder_Cranach
http://dbpedia.org/resource/Lucius_Cranach_the_Elder
http://dbpedia.org/resource/Lucius_Cranach_the_elder
```

Check of the first one:

```
describe <http://dbpedia.org/resource/Cranach,_Lucas_the_Elder>
```

It returns a number of statements, of which the most important are:

```
<http://dbpedia.org/resource/Cranach,_Lucas_the_Elder> rdfs:label
"Cranach, Lucas the Elder"@en .
<http://dbpedia.org/resource/Cranach,_Lucas_the_Elder> dbpedia-
owl:wikiPageRedirects dbpedia:Lucas_Cranach_the_Elder ;
```

Not all redirects provide alternative names for an entity (e.g. https://en.wikipedia.org/wiki/God_does_not_play_dice goes to the page Albert_Einstein, although this is something he said, not an alternative name for him). But most provide alternative names, so we can use them:

```
select ?x {[] dbpedia-owl:wikiPageRedirects
<http://dbpedia.org/resource/Lucas_Cranach_the_Elder>;
  rdfs:label ?x}
```

Because redirects are resolved to the ultimate target, it is not necessary to use a property path "+" (Kleene closure)

3.2.3.7 DBpedia Names

Different editions use different **raw** properties for names. A lot of them but not all are mapped to standard properties, because name properties are not always used consistently across DBpedia mappings. We explored the different name properties on en, fr, de DBpedia and came up with a query like this:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
prefix dbo: <http://dbpedia.org/ontology/>
prefix prop: <http://dbpedia.org/property/>
prefix prop-de: <http://de.dbpedia.org/property/>
prefix prop-fr: <http://fr.dbpedia.org/property/>

select ?x ?p ?n {
  {?x dbo:wikiPageRedirects <http://dbpedia.org/resource/Lucas_Cranach_the_Elder>;
rdfs:label ?n} union
  {<http://dbpedia.org/resource/Lucas_Cranach_the_Elder> ?p ?n.
    filter (?p in (
    foaf:name, foaf:givenName, foaf:surname, foaf:familyName, rdfs:label,
skos:prefLabel, skos:altLabel, dbo:birthName,
    prop:birthName, prop:name, prop:title,
    prop-de:name, prop-de:alternativnamen,
    prop-fr:nom, prop-fr:commonsTitre, prop-fr:nomDeNaissance,
))}} order by ?x ?p ?n
```

Note: unfortunately some DBpedia endpoints (e.g. Italy) do not support SPARQL 1.1.

Using this query across several national DBpedias (or a variant with sameAs on the LOD Cache), we collected 43 names for Cranach: ./cranach-dbpedia.txt

3.2.3.8 DBpedia Name Mapping

Mapped name properties include:

```
foaf:name, foaf:givenName, foaf:surname, foaf:familyName, rdfs:label,
skos:prefLabel, skos:altLabel, dbo:birthName
```

You may wonder why do we need "raw" properties like these:

```
prop:birthName, prop:name, prop:title,
prop-de:name, prop-de:alternativnamen,
prop-fr:nom, prop-fr:commonsTitre, prop-fr:nomDeNaissance
```

The answer is that some templates take care to map all name properties, but others do not. Here we find people with the raw property prop:birthName that do not have the mapped property dbo:birthName

```
prefix dbo: <http://dbpedia.org/ontology/>
prefix prop: <http://dbpedia.org/property/>
select * {
  ?x prop:birthName ?n
  filter (lang(?n)="en" &&
    !(str(?n) in ("?", "???", "Unknown", "unknown")) &&
    not exists {?x dbo:birthName ?n}}}
```

- The raw property grabs anything it finds in the template field. If removing the condition `lang(?n)` all kinds of miscellany are displayed, from dates to families.
- language tags are fixed to en (by default), so are not reliable. e.g. "Никола́й Ива́нович Буха́рин"@en is in Russian not English

3.2.4 VIAF

VIAF is a large-scale collaboration of national libraries and OCLC to produce a Virtual International Authority File. As of Dec 2014, VIAF has 35 contributing institutions (9 through the LCC NACO) and 9 contributors in test (including ISNI, Wikipedia, Perseus).

3.2.4.1 VIAF Algorithms

VIAF uses sophisticated matching and clustering algorithms [3] to match named entities across name authorities. These include people, organisations, conferences, places, works, expressions (e.g. a certain edition or translation of a work), subject headings, etc. VIAF is somewhat conservative in not making possible matches that are not warranted by sufficient information.

VIAF cluster IDs are relatively stable, but when monthly updates are received from the contributing institutions, it is possible that an authority record is reassigned to another VIAF cluster, or two VIAF clusters are merged, or a VIAF ID is abandoned. Nevertheless VIAF makes everything possible to preserve IDs:

- when a new cluster is formed, it first seeks to reuse an abandoned ID that was previously used for some of the records in the cluster
- when an ID is abandoned, leaves a redirect to the surviving cluster that holds most records from the abandoned cluster

3.2.4.2 VIAF Counts

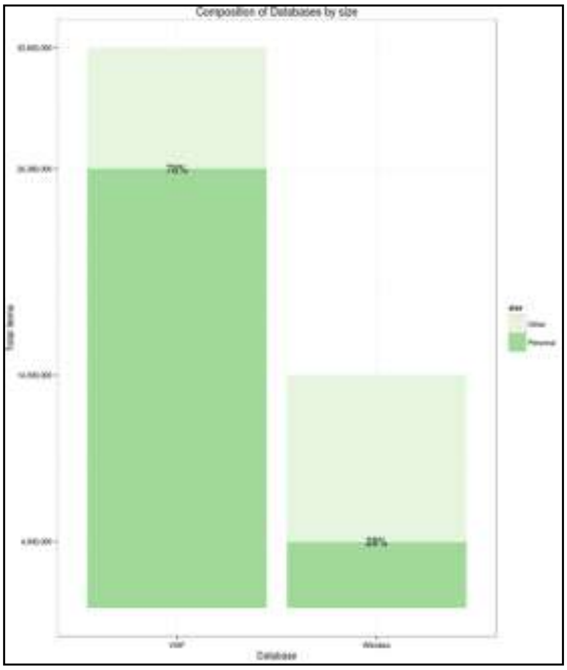
Recent VIAF counts are provided in the 2014 Annual Report. The number of VIAF clusters is as follows (also see Co-referencing for breakdown per VIAF member):

- Personal: 35,163,929
- Corporate/conferences: 5,425,304
- Geographic: 416,316
- Work: 1,685,745
- Expression: 287,211

Also interesting are the numbers on p6, in particular:

- Wikipedia/Wikidata: 1,135,025 Person records imported, of which 37% are matched

To appreciate the size, an image from [6] is reproduced that compares VIAF with Wikidata (thus indirectly DBpedia):



It is worth to compare to Wikidata Coverage and Type Count , which counts "human" items in Wikidata as 2.7M.

- This is lower than on the graphic, but higher than the number on p6 (how were these 1.1M records selected?)

3.2.4.3 VIAF Access

VIAF has a basic search at http://viaf.org/, and an advanced (SRU-based) search at http://viaf.org/viaf/search/.

If searching for "Personal name: Lucas Cranach" you may find:

- (top) a main cluster http://viaf.org/viaf/49268177 that carries a lot of info and is the result of matching many source records (including from DNB)
- (middle) 31 persons who are either related to Cranach (e.g. Maximilian I Holy Roman Emperor, painted by Cranach in 1509), or share a name
- (bottom) two stand-alone (singleton) clusters (coming from DNB):
  - http://viaf.org/viaf/308208350 from DBN: "Cranach, Lucas d. Ä. oder d. J." (The Elder or The Younger), to be used for works with unclear attribution to the father or the son
  - http://viaf.org/viaf/238031633 from DNB: "Cranach, Lucas" marked (undifferentiated) (sparse), for which there is too little info to warrant a match.
  - http://viaf.org/viaf/96020412 from ULAN: which has this note: "Given that the name is rather common, it is uncertain whether or not this artist is identifiable as one of the two famous artists named "Lucas Cranach."

VIAF is conservative in matching: even though the names of these clusters match, there are no years, so VIAF does not cluster them.

The main Cranach cluster has 44 Works, several download formats in Record Views, and 71 names: ./cranach-VIAF.txt.

The RDF is at http://viaf.org/viaf/49268177/rdf.xml and is available here in Turtle for easier understanding: ./cranach-viaf.ttl. It follows a dual approach as explained in [7] sec 3.3. An abbreviated version follows:

```
<http://viaf.org/viaf/49268177/> a foaf:Document ;
  void:inDataset     <http://viaf.org/viaf/data> ;
  foaf:primaryTopic  <http://viaf.org/viaf/49268177> .

<http://viaf.org/viaf/49268177> a schema:Person ;
  schema:alternateName   "Sunder-Maler, Lucas" , "Müller, Lukas" ...;
  schema:birthDate       "1472-10-04" ;
  schema:deathDate       "1553-10-16" ;
  schema:description     "German painter, draftsman, and printmaker, 1472-1553" ;
  schema:familyName      "קראנאך" , "Cranach" , "Кранах" ;
  schema:givenName       "Лукас" , "Lucas" , "לוקאס האב" ,
                         "Lucas the Elder (studio of)" ...;
  schema:name            "Кранах, Лукас" , "קראנאך, לוקאס, האב (סדנת)" ,
                         "Cranach, Lucas, the Elder" ...;
  schema:sameAs          <http://data.bnf.fr/ark:/12148/cb12176451h#foaf:Person> ,
                         <http://dbpedia.org/resource/Lucas_Cranach_the_Elder> ,
                         <http://d-nb.info/gnd/118522582> ,
                         <http://www.idref.fr/028710010/id> ,
                         <http://libris.kb.se/resource/auth/182422> ;
  foaf:isPrimaryTopicOf  <http://en.wikipedia.org/wiki/Lucas_Cranach_the_Elder> .

<http://viaf.org/viaf/sourceID/BNF%7C12176451#skos:Concept> a skos:Concept ;
  rdfs:seeAlso     <http://catalogue.bnf.fr/ark:/12148/cb12176451h> ;
  skos:altLabel    "Cranach der Ältere Lucas 1472-1553" ,
                   "Cranach Lukas 1472-1553" ,
                   "Cranach l'ancien Lucas 1472-1553"...;
  skos:exactMatch  <http://data.bnf.fr/ark:/12148/cb12176451h> ;
  skos:inScheme    <http://viaf.org/authorityScheme/BNF> ;
  skos:prefLabel   "Cranach, Lucas, 1472-1553." ;
  foaf:focus       <http://viaf.org/viaf/49268177> .
```

- The central node is a schema:Person, having birth/death dates, names, alternate names, even given/family names (though "studio of" is hardly a given name)
- The Person is declared owl:sameAs all corresponding nodes in contributing organizations that have an appropriate type (e.g. foaf:Person for BNF, dbo:Person for DBPedia)
- There are two documents (the VIAF page and Wikipedia page) that point to the Person using foaf:primaryTopic.
- There is a skos:Concept for each of the contributor nodes (members of the cluster) that points to the Person using foaf:focus
- These Concepts hold the prefLabel and altLabels as determined by the contributing institution

- VIAF does not have language tags, which is an omission

Overall, this structure is perfectly correct and provides both a lot of names, and also a lot of links.

3.2.4.4 VIAF Download

VIAF provides monthly dumps at http://viaf.org/viaf/data/ (this file is RDFa, i.e. both human and machine readable description). The following files are of interest. The first is analysed and the second is loaded to a repository:

file	gz	description
links.txt+	0.4G	Co-reference VIAF->contributor id, including external links such as Wikipedia
clusters-rdf.nt+	8.3G	one line per statement, all statements for each cluster
clusters-rdf.xml	4.2G	one line per cluster, containing RDF like the above Cranach link
persist-rdf.xml	0.09G	redirections between VIAF clusters. Happens when a cluster is split or merged, see VIAF Algorithms

The average compression is 4.8x. The files are pretty large, but manageable (unzipped: links.txt 2Gb, clusters-rdf.nt 40G)

3.2.5 ISNI

ISNI (International Standard Name Identifier) is an international cooperation that on one hand feeds from VIAF, and on the other hand caters to easy institutional registration of modern authors (whereas ORCID allows easy personal registration).

[2] explains well the similarities and differences between ISNI and VIAF.

The ISNI record for Cranach is http://isni.org/isni/0000000121319721 and has 51 names: ./cranach-ISNI.txt. An "almost RDF" file is available at http://isni.org/isni/0000000121319721.rdf but unfortunately this is not valid RDF:

- It starts with a custom element <isni:PersonPublicIdentity>
- It references a non-existing http://isni.org/ontology

The ISNI names are a subset of the VIAF names, so the conclusion is that ISNI can be ignored.

3.2.6 Getty ULAN

The Union List of Artist Names (ULAN) of the Getty Research Institute is a well-known personal name thesaurus.

- ULAN publication as LOD is expected in Mar 2015, similar to the AAT and TGN publications at http://vocab.getty.edu/sparql

The Cranach record is at http://vocab.getty.edu/ulan/500115364 and has 25 names: ./cranach-ULAN.txt.

- ULAN is a relatively small authority (230k records)
- ULAN names are subsumed by VIAF since ULAN is a fully-fledged contributor to VIAF
- However, ULAN is carefully curated, every name/fact has a documented source, and it includes valuable person information such as roles (types), relations (e.g. influenced, student), life events. These can be useful for disambiguation

3.2.7 Yago Knowledge

Yago provides an important contribution to DBpedia in the form of additional instance types, and integration to Wordnet. While DBpedia instance types are determined by the applied templates, Yago types are determined by NLP over the Wikipedia categories.

Yago has the same coverage as DBpedia (it does not have independently developed entries).

The Yago record for Cranach is at http://yago-knowledge.org/resource/Lucas_Cranach_the_Elder

- It is in standard NTriples format (text/plain)
- There are 37 names: cranach-yago.txt
- Most do not have language tags, except 4 (de, lv, pl, simple; the latter does not conform to RDF/IANA rules)

3.2.8 British Museum

The British Museum LOD collection (http://collection.britishmuseum.org) uses a number of thesauri (about 40).

- Many of them are visible in CSV format at *Github*
- The person-institution thesaurus has 176,461 entries, which can be download in a richer form here

The Cranach record is at http://collection.britishmuseum.org/id/person-institution/23953 and has only two names: Lucas Cranach the Elder and Cranach, Lucas. So it is not considered below.

3.2.9 LOD Cache

The LOD Cache SPARQL endpoint http://lod.openlinksw.com/sparql by Open Link Software includes a lot of aggregated data from LOD datasets. It includes the following name sources considered above:

- Wikidata
- DBpedia: EN & FR (in full, not just labels and abstracts in foreign languages like dbpedia.org)

  - The following DBpedias are not included: IT, DE
- FreeBase

Some caveats:

- Unfortunately the endpoint is quite unreliable. The SPARQL Endpoint Status service showed 84.6% availability for the month of Nov 2014. At 2014-12-02 11:18 the endpoint returned this error:

```
Virtuoso 08C01 Error CL...: Cluster could not connect to host 2
oplbfc3:22202 error 111
```

- The update rate is unclear, so one should be careful to evaluate whether all data is present by consulting the original sources

Following query is a combination of DBpedia sameAs , Wikipedia Redirects and DBpedia Names . The result is a table from-LOD-cache.tdv with 216 rows. The unique labels only (there's 88) are checked and compared to Wikipedia+VIAF.

```
perl -pe '$_=(split/\t/)[2]; s{"(.*)"@?[\w-]*}{$1}; s{[,.]}{}g' from-LOD-
cache.tdv |sort|uniq > from-LOD-uniq.txt
cat cranach-wikidata.txt cranach-VIAF.txt | sort | uniq > Wikidata-VIAF-
uniq.txt
```

- There are 146 names in Wikidata-VIAF-uniq.txt and 83 in from-LOD-uniq.txt
- There are only 4 unique contributions in from-LOD-uniq.txt:

```
Cranach the Elder
Lucas Cranach "el Vell"
Lucas Cranach "el Viejo"
Lucas Maler
```

Overall, for any production work it is recommended to load the desired datasets to a local repository. Otherwise continuity of service cannot be guaranteed.

3.3 Comparing Sources

After fetching the name forms from all sources, the overlaps and unique contributions are analysed. They are tabulated to a common file, using common Unix tools (perl, join, uniq, sort) and Excel

- All files from different sources are concatenated, unified and sorted, obtaining 155 names Note: if working on Windows (e.g. using Cygwin), convert all files to Unix newlines: `conv -U *.txt` Unicode BOM should not be used, as sort and join do not work
- Tabulation with a series of commands like this (in ./cranach-table.sh)

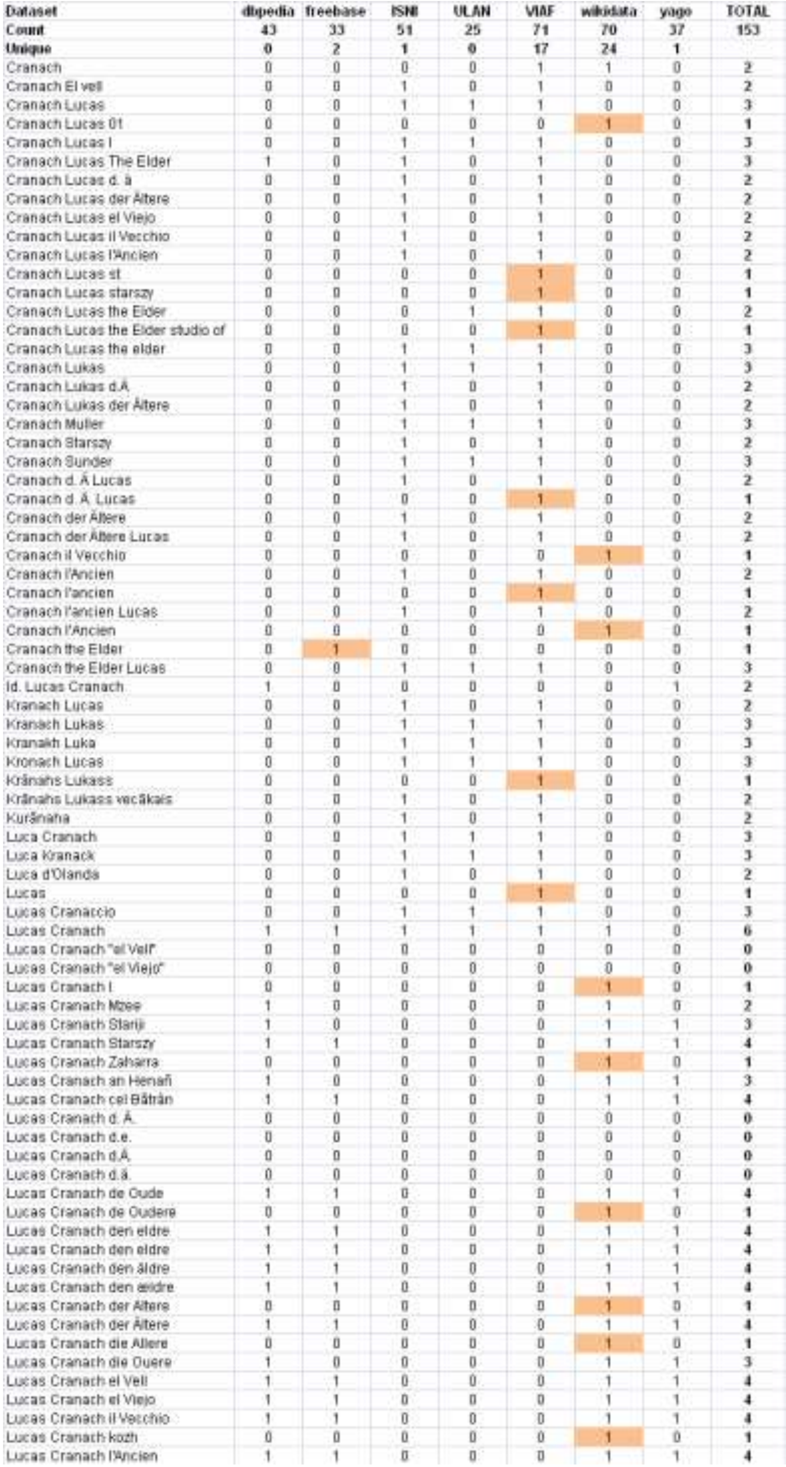
```
perl -pe 's{(.+)}{$1|1}' Cranach-VIAF.txt \
  | join -t '|' -a1 -e0 -o1.1,1.2,1.3,1.4,1.5,2.2 Cranach4.txt - > Cranach5.txt
```

- The perl command adds "|1" to the end of each line. "1" indicates there is a value, and "|" is a record separator
- join -t sets the tab separator, -a1 does a left outer join, -e0 replaces missing values (rows from the right line) with "0".
- -o1.1,1.2,1.3,1.4,1.5,2.2 sets the output format: all 5 columns from the left file (into which 4 inputs have already been merged), then the "0"/"1" indicator from the right file

3.3.1 Source Counts

The merged table is opened with Excel, where some calculations and conditional formatting are added: ./cranach-table.xlsx.

- Count is the number of names per dataset
- Unique is the unique contributions, which are highlighted in red. VIAF and Wikidata have most uniques



Dataset	dbpedia	freebase	ISNI	ULAN	VIAF	wikidata	yago	TOTAL
Count	43	33	51	25	71	70	37	153
Unique	0	2	1	0	17	24	1	
Cranach	0	0	0	0	1	1	0	2
Cranach El vell	0	0	1	0	1	0	0	2
Cranach Lucas	0	0	1	1	1	0	0	3
Cranach Lucas 01	0	0	0	0	0	1	0	1
Cranach Lucas I	0	0	1	1	1	0	0	3
Cranach Lucas The Elder	1	0	1	0	1	0	0	3
Cranach Lucas d. à	0	0	1	0	1	0	0	2
Cranach Lucas der Ältere	0	0	1	0	1	0	0	2
Cranach Lucas el Viejo	0	0	1	0	1	0	0	2
Cranach Lucas il Vecchio	0	0	1	0	1	0	0	2
Cranach Lucas l'Ancien	0	0	1	0	1	0	0	2
Cranach Lucas st	0	0	0	0	1	0	0	1
Cranach Lucas starszy	0	0	0	0	1	0	0	1
Cranach Lucas the Elder	0	0	0	1	1	0	0	2
Cranach Lucas the Elder studio of	0	0	0	0	1	0	0	1
Cranach Lucas the elder	0	0	1	1	1	0	0	3
Cranach Lukas	0	0	1	1	1	0	0	3
Cranach Lukas d.Ä	0	0	1	0	1	0	0	2
Cranach Lukas der Ältere	0	0	1	0	1	0	0	2
Cranach Muller	0	0	1	1	1	0	0	3
Cranach Starszy	0	0	1	0	1	0	0	2
Cranach Sunder	0	0	1	1	1	0	0	3
Cranach d. Ä Lucas	0	0	1	0	1	0	0	2
Cranach d. Ä. Lucas	0	0	0	0	1	0	0	1
Cranach der Ältere	0	0	1	0	1	0	0	2
Cranach der Ältere Lucas	0	0	1	0	1	0	0	2
Cranach il Vecchio	0	0	0	0	0	1	0	1
Cranach l'Ancien	0	0	1	0	1	0	0	2
Cranach l'ancien	0	0	0	0	1	0	0	1
Cranach l'ancien Lucas	0	0	1	0	1	0	0	2
Cranach l'Ancien	0	0	0	0	0	1	0	1
Cranach the Elder	0	1	0	0	0	0	0	1
Cranach the Elder Lucas	0	0	1	1	1	0	0	3
Id. Lucas Cranach	1	0	0	0	0	0	1	2
Kranach Lucas	0	0	1	0	1	0	0	2
Kranach Lukas	0	0	1	1	1	0	0	3
Kranakh Luka	0	0	1	1	1	0	0	3
Kronach Lucas	0	0	1	1	1	0	0	3
Krānahs Lukass	0	0	0	0	1	0	0	1
Krānahs Lukass vecākais	0	0	1	0	1	0	0	2
Kurānaha	0	0	1	0	1	0	0	2
Luca Cranach	0	0	1	1	1	0	0	3
Luca Kranack	0	0	1	1	1	0	0	3
Luca d'Olanda	0	0	1	0	1	0	0	2
Lucas	0	0	0	0	1	0	0	1
Lucas Cranaccio	0	0	1	1	1	0	0	3
Lucas Cranach	1	1	1	1	1	1	0	6
Lucas Cranach "el Vell"	0	0	0	0	0	0	0	0
Lucas Cranach "el Viejo"	0	0	0	0	0	0	0	0
Lucas Cranach I	0	0	0	0	0	1	0	1
Lucas Cranach Mzee	1	0	0	0	0	1	0	2
Lucas Cranach Stariji	1	0	0	0	0	1	1	3
Lucas Cranach Starszy	1	1	0	0	0	1	1	4
Lucas Cranach Zaharra	0	0	0	0	0	1	0	1
Lucas Cranach an Henañ	1	0	0	0	0	1	1	3
Lucas Cranach cel Bătrân	1	1	0	0	0	1	1	4
Lucas Cranach d. Ä.	0	0	0	0	0	0	0	0
Lucas Cranach d.e.	0	0	0	0	0	0	0	0
Lucas Cranach d.Ä.	0	0	0	0	0	0	0	0
Lucas Cranach d.ä.	0	0	0	0	0	0	0	0
Lucas Cranach de Oude	1	1	0	0	0	1	1	4
Lucas Cranach de Oudere	0	0	0	0	0	1	0	1
Lucas Cranach den eldre	1	1	0	0	0	1	1	4
Lucas Cranach den eldre	1	1	0	0	0	1	1	4
Lucas Cranach den äldre	1	1	0	0	0	1	1	4
Lucas Cranach den ældre	1	1	0	0	0	1	1	4
Lucas Cranach der Altere	0	0	0	0	0	1	0	1
Lucas Cranach der Ältere	1	1	0	0	0	1	1	4
Lucas Cranach die Allere	0	0	0	0	0	1	0	1
Lucas Cranach die Ouere	1	0	0	0	0	1	1	3
Lucas Cranach el Vell	1	1	0	0	0	1	1	4
Lucas Cranach el Viejo	1	1	0	0	0	1	1	4
Lucas Cranach il Vecchio	1	1	0	0	0	1	1	4
Lucas Cranach kozh	0	0	0	0	0	1	0	1
Lucas Cranach l'Ancien	1	1	0	0	0	1	1	4

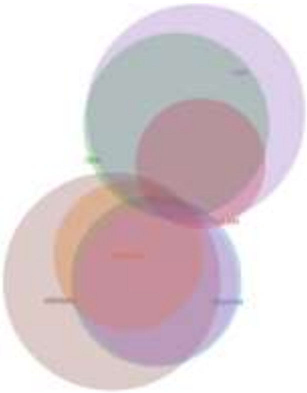
3.3.2 Venn Diagram

It is hard to figure out the correlations between sets from this table, so it was decided to make a Venn diagram. Most Venn libraries can work with 3 or maximum 4 sets, but the excellent venn.js can work with **any number** of sets. Using the `Algorithm::Combinatorics` perl module, a script ./cranach-venn.pl was created that counts the cardinalities of all set intersections (potentially 2^7=128). The result was formatted as ./cranach-venn.jsonp, following an example in venn.js:

```
perl cranach-venn.pl cranach-table.txt > cranach-venn.jsonp
```

The result is ./cranach-venn.html.

- We **strongly recommend** that you play with the interactive version ./cranach-venn.html, since it highlights intersections and reveals their cardinalities, allowing better understanding of the arrangement.
- The diagram is approximate, e.g. ULAN is wholly within VIAF: if you try to point out the little sliver ULAN\VIAF, you'll discover it has cardinality 0. But it is quite accurate!



Notes:

- A striking revelation is that the 3 "library-tradition" datasets (VIAF, ISNI, ULAN) and the 4 "LOD-tradition datasets" (Wikidata, DBpedia, Freebase, Yago) have almost nothing in common: only 5 names. Library datasets contribute many permutations and qualifiers (e.g. "der Altere" vs. "d A"), while LOD datasets contribute many languages.
- The datasets in each "tradition" are very similar. The reason is obvious: ISNI and ULAN are fully-fledged contributors in VIAF, so VIAF subsumes them. As for the LOD datasets, each has copied from the others liberally. DBpedia appears as a subset of Wikidata as only en, de, fr names were selected(See DBpedia Names). Yago covers the en DBpedia, and Freebase does not contribute many unique names either.

- The circles represent number of names for this single example, not dataset coverage. Remember that VIAF is some 12x bigger than Wikidata, see VIAF Counts
- The focus should be on Wikidata and VIAF. If DBpedia, Freebase, Yago are dropped, 4 names are lost, if ISNI and ULAN are dropped only 1 name is lost.

3.4 Co-referencing

Co-referencing is the alignment of Authority databases, typically by aggregation of identifiers from one database to another. An example can be seen best on the Reasonator page for Cranach (the right side). (This data is used in the next section.) Each co-reference ID is also a link. Of course, whenever the target Authority has an RDF representation, the links are also machine-navigable.

As one can surmise from the previous section, the two currently most-prominent Person Authorities (**hubs**) are VIAF and Wikidata, which is also confirmed by [6].

- The benefits of co-referencing are significant for Authority providers, as it allows cross-checking, adding missing information, and leveraging independent work done in other datasets
- There are also benefits of co-referencing for consumers such as Europeana enrichment: significantly enlarged coverage (union of two datasets) while avoiding the danger of duplicate entities; increasing the number of names and extra characteristics for individual objects.

3.4.1 VIAF Co-referencing

VIAF co-referencing is performed across the contributing datasets by sophisticated algorithms, see VIAF Algorithms . [4] describes how VIAF -> Wikipedia matchings were imported automatically to Wikidata by a "bot".

A recent count of VIAF correlations using the Links file has been done. These are links from VIAF to other authorities, which allows to surmise the **matched** item counts for each dataset as well.

- xR and xA are auxiliary authorities developed by OCLC, which serve as sort of "control files" to take care of difficult cases

count	code	dataset
320898	BAV	Vatican
73421	BIBSYS	Norway
144299	BNC	Catalunya
562244	BNE	Spain
2036493	BNF	France (BnF)
101500	DBC	Denmark (DBC)
10531522	DNB	Germany
37004	EGAXA	Egypt
169028	ICCU	Italy
9953	IMAGINE	Israel
7655649	ISNI	ISNI
232327	JPG	Getty (ULAN)
689827	LAC	Canada
9154093	LC	LC (NACO)
158515	LNB	Latvia
11000	LNL	Lebanon
1032862	NDL	Japan (NDL)
743215	NKC	Czech
1016708	NLA	Australia
408	NLB	Singapore
570840	NLI	Israel
844024	NLP	Poland (Nat lib)
473518	NSK	Croatia
33727	NSZL	Hungary
2555033	NTA	Netherlands
1351105	NUKAT	Poland (NUKAT)
1228	PERSEUS	Perseus
373078	PTBNP	Portugal
220304	RERO	Swiss (RERO)
997	RSL	Russia
187073	SELIBR	Sweden
209	SRP	Syriac
2508374	SUDOC	France (Sudoc)
45633	SWNL	Swiss (Nat lib)
5723	VLACC	Belgium (Flemish)
377650	WKP	Wikipedia
267	XA	xA OCLC file
2018647	XR	xR OCLC file
27684634	VIAF	TOTAL

3.4.2 VIAF vs. Wikidata Co-referencing

Next, some co-referencing action between the two hubs is shown:

- co-refThe Wikidata co-reference IDs on the Reasonator page for Cranach were already shown.
- co-ref
- VIAF has an API "justlinks" to return only the co-references, e.g. for Cranach: http://viaf.org/viaf/49268177/justlinks.json (Note: 4 of the fields were URLs, the ID is left out for easier comparison)

VIAF	id in VIAF	Wikidata	id in Wikidata
viafID	49268177	VIAF	49268177
BAV	ADV10197613		
BNC	.a10853637		
BNE	XX907273		
BNF	cb12176451h	BNF	12176451h
DNB	118522582	GND	118522582
ISNI	0000000121319721	ISNI	0000 0001 2131 9721
JPG	500115364	ULAN	500115364
LC	n50020861	LCCN	n50020861
LNB	LNC10-000002573		
NDL	00436834		
NKC	jn20000700335		
NLA	000035031951		
NLI	000035532,001445575,001448179		
NLP	a16828161		
NTA	068435312	NTA PPN	068435312
NUKAT	vtls000190728		
SELIBR	182422		
SUDOC	028710010		
WKP	Lucas_Cranach_the_Elder	Many Wikipedias	
IMAGINE	T7238,T267474		
		Cantic	a10853637
		Commons Creator	Lucas Cranach (I)
		Commons category	Lucas Cranach d. Ä.
		Freebase	/m/0kqp0
		RKDartists	18978
		SIMBAD	CRANACH, Lucas the Elder
		Your Paintings	lucas-the-elder-cranach

As one can see, there are a number of "gaps" in each hub that could be filled out from the other hub.

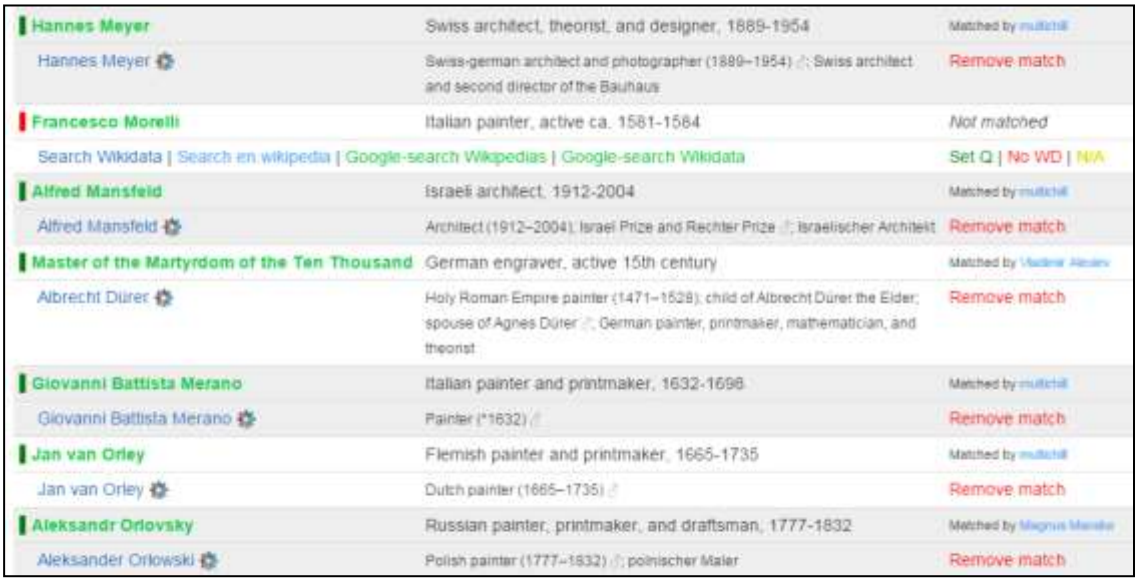
- E.g. RKDartists is an important Authority that does not yet participate in VIAF. There are already 21760 RKDartist id's on Wikidata. These could be imported to VIAF for free!
- In this case each hub has the ID of the other hub. But this need not always be the case:
  - Wikidata has 504736 items with VIAF id
  - Wikidata has 567240 items with VIAF or GND
  - Since all GND items are likely to be in VIAF, this shows that in Wikidata, 62504 items with GND id do not have a VIAF ID. VIAF IDs can be assigned to these easily.
- Missing data (e.g. birth/death date/place) can be filled out from one hub to the other

A WikiProject Authority Control was recently proposed to coordinate such developments.

3.4.3 Wikidata Co-referencing with Mix-n-Match

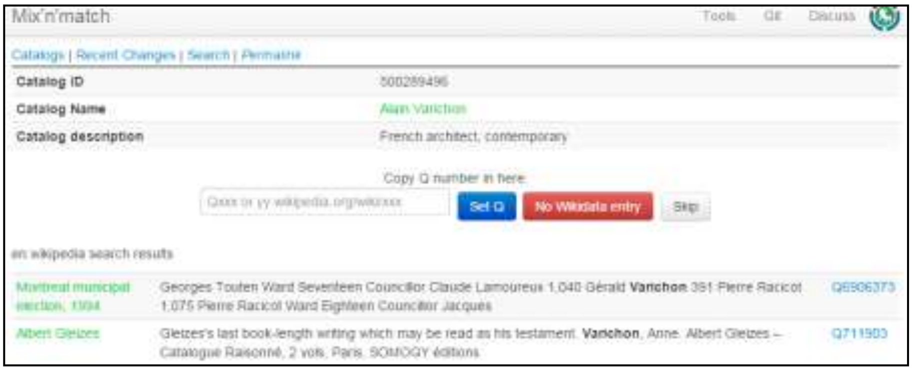
Mix-n-Match is a tool for matching Wikidata items to authority databases, by Magnus Manske who also created Reasonator. In this way the authority databases can be co-referenced, and thereon linked to Wikipedia. It has (simple) automatic matching based on names and dates, followed by crowd-sourced edits. [8] and [9] describe using the tool to co-reference the Oxford Dictionary of National Biography. Some examples follow:

- List of datasets (catalogues) subject to matching with statistics
- Matching of ULAN



Hannes Meyer	Swiss architect, theorist, and designer, 1889-1954	Matched by multichill
Hannes Meyer	Swiss-german architect and photographer (1889–1954); Swiss architect and second director of the Bauhaus	Remove match
Francesco Morelli	Italian painter, active ca. 1581-1584	*Not matched*
Search Wikidata | Search en wikipedia | Google-search Wikipedias | Google-search Wikidata		Set Q | No WD | N/A
Alfred Mansfeld	Israeli architect, 1912-2004	Matched by multichill
Alfred Mansfeld	Architect (1912–2004), Israel Prize and Rechter Prize ; Israelischer Architekt	Remove match
Master of the Martyrdom of the Ten Thousand	German engraver, active 15th century	Matched by Vladimir Alexiev
Albrecht Dürer	Holy Roman Empire painter (1471–1528), child of Albrecht Dürer the Elder; spouse of Agnes Dürer ; German painter, printmaker, mathematician, and theorist	Remove match
Giovanni Battista Merano	Italian painter and printmaker, 1632-1698	Matched by multichill
Giovanni Battista Merano	Painter (*1632)	Remove match
Jan van Orley	Flemish painter and printmaker, 1665-1735	Matched by multichill
Jan van Orley	Dutch painter (1665–1735)	Remove match
Aleksandr Orlovsky	Russian painter, printmaker, and draftsman, 1777-1832	Matched by Magnus Manske
Aleksander Orlowski	Polish painter (1777–1832) ; polnischer Maler	Remove match

- Matching in "game" mode: 1 record at a time for casual users


Mix'n'match
Tools
Git
Discuss
Catalogs | Recent Changes | Search | Permalink
Copy Q number in here
Qxxx or yy.wikipedia.org/wiki/xxx
Set Q
No Wikidata entry
Skip

Catalog ID	500289496
Catalog Name	Alain Varichon
Catalog description	French architect, contemporary

en.wikipedia search results

Montreal municipal election, 1994	Georges Touten Ward Seventeen Councillor Claude Lamoureux 1,040 Gérald **Varichon** 361 Pierre Racicot 1,075 Pierre Racicot Ward Eighteen Councillor Jacques	Q6906373
Albert Gleizes	Gleizes's last book-length writing which may be read as his testament. **Varichon**, Anne. Albert Gleizes – Catalogue Raisonné, 2 vols. Paris. SOMOGY éditions	Q711903

3.4.4 Downloading Co-references from Mix-n-Match

- Download TDV of matches for a given catalog (ULAN):

Download BEACON co-reference file from wikidata. BEACON is a simple tuple or triple format. The query parameters correspond to the result fields as follows: source->PREFIX, prop-TARGET

- VIAF-wikidata-ULAN:

```
#PREFIX: https://viaf.org/viaf/
#TARGET: http://vocab.getty.edu/ulan/
100001869|Q29418|500008217
```

- ULAN-wikidata-VIAF:

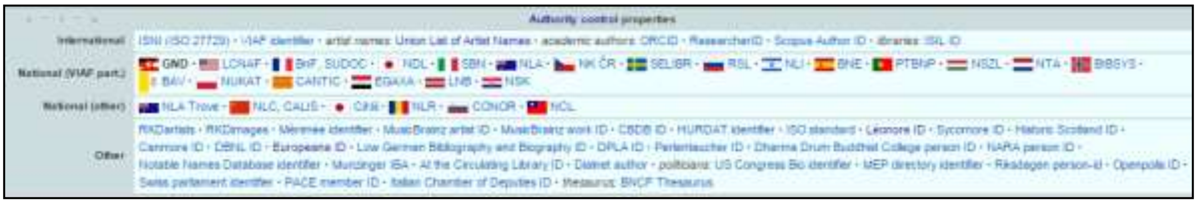
```
#PREFIX: http://vocab.getty.edu/ulan/
#TARGET: https://viaf.org/viaf/
500000006|Q123948|20472726
```

- RKDartists-wikidata-ULAN: no problem, even though RKDartists is not yet in VIAF!

```
#PREFIX: https://rkd.nl/explore/artists/
#TARGET: http://vocab.getty.edu/ulan/
1|Q3651930|500067169
10008|Q715909|500023946
100086|Q3161825|500068086
100140|Q3383669|500126269
```

3.4.5 Wikidata Authority Identifiers

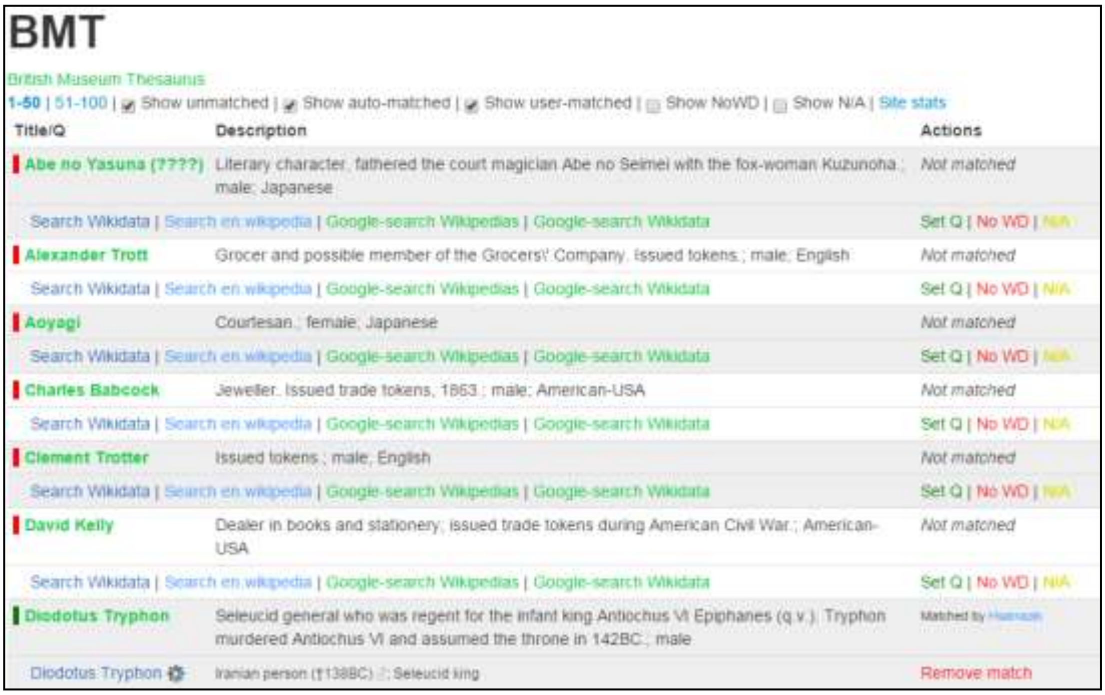
A prerequisite for co-referencing is to register authority files as Wikidata items, and their IDs as Wikidata properties (carrying annotation "Wikidata property for authority control"). All kinds of international and national authority files are already registered (e.g. see a big list on Wikisource or a sampling on Wikisouce), and new ones are proposed daily. These identifiers are used in items and articles, and displayed as a visually striking Authority Control box



Authority control properties	
International	ISNI (ISO 27729) · VIAF identifier · artist names: Union List of Artist Names · academic authors: ORCID · ResearcherID · Scopus Author ID · libraries: ISIL ID
National (VIAF part.)	GND · LCNAF · BnF, SUDOC · NDL · SBN · NLA · NK ČR · SELIBR · RSL · NLI · BNE · PTBNP · NSZL · NTA · BIBSYS · BAV · NUKAT · CANTIC · EGAXA · LNB · NSK
National (other)	NLA Trove · NLC, CALIS · CiNii · NLR · CONOR · NCL
Other	RKDartists · RKDimages · Mérimée identifier · MusicBrainz artist ID · MusicBrainz work ID · CBDB ID · HURDAT identifier · ISO standard · Léonore ID · Sycomore ID · Historic Scotland ID · Canmore ID · DBNL ID · Europeana ID · Low German Bibliography and Biography ID · DPLA ID · Parlement.com ID · Dharma Drum Buddhist College person ID · NARA person ID · Notable Names Database identifier · Munzinger IBA · At the Circulating Library ID · Dialnet author · politicians: US Congress Bio identifier · MEP directory identifier · Riksdagen person-id · Openpolis ID · Swiss parliament identifier · PACE member ID · Italian Chamber of Deputies ID · thesaurus: BNCF Thesaurus

3.4.6 British Museum Co-referencing

The British Museum thesauri are not co-referenced. Since the British Museum has published 2.5M objects as LOD, it would be quite valuable to co-reference the British Museum thesauri. A proposal to do this on Wikidata using the TDV export (see British Museum) was recently made, and co-referencing has already started:



BMT

British Museum Thesaurus

1-50 | 51-100 | ☑ Show unmatched | ☑ Show auto-matched | ☑ Show user-matched | □ Show NoWD | □ Show N/A | Site stats

Title/Q	Description	Actions
Abe no Yasuna (????)	Literary character, fathered the court magician Abe no Seimei with the fox-woman Kuzunoha.; male; Japanese	Not matched
Search Wikidata | Search en wikipedia | Google-search Wikipedias | Google-search Wikidata		Set Q | No WD | N/A
Alexander Trott	Grocer and possible member of the Grocers\' Company. Issued tokens.; male; English	Not matched
Search Wikidata | Search en wikipedia | Google-search Wikipedias | Google-search Wikidata		Set Q | No WD | N/A
Aoyagi	Courtesan.; female; Japanese	Not matched
Search Wikidata | Search en wikipedia | Google-search Wikipedias | Google-search Wikidata		Set Q | No WD | N/A
Charles Babcock	Jeweller. Issued trade tokens, 1863.; male; American-USA	Not matched
Search Wikidata | Search en wikipedia | Google-search Wikipedias | Google-search Wikidata		Set Q | No WD | N/A
Clement Trotter	Issued tokens.; male; English	Not matched
Search Wikidata | Search en wikipedia | Google-search Wikipedias | Google-search Wikidata		Set Q | No WD | N/A
David Kelly	Dealer in books and stationery; issued trade tokens during American Civil War.; American-USA	Not matched
Search Wikidata | Search en wikipedia | Google-search Wikipedias | Google-search Wikidata		Set Q | No WD | N/A
Diodotus Tryphon	Seleucid general who was regent for the infant king Antiochus VI Epiphanes (q.v.). Tryphon murdered Antiochus VI and assumed the throne in 142BC.; male	Matched by Hannah
Diodotus Tryphon	Iranian person (†138BC) ♂; Seleucid king	Remove match

3.4.7 Wikidata Correlation Ids on DBpedia

Some correlations are already available on the DBpedia or LOD Cache endpoints

```
PREFIX wikidata: <http://www.wikidata.org/entity/>
prefix dbo: <http://dbpedia.org/ontology/>
prefix prop-de: <http://de.dbpedia.org/property/>

select ?p ?n {
  {<http://dbpedia.org/resource/Lucas_Cranach_the_Elder> ?p ?n}
  union
  {?x owl:sameAs <http://dbpedia.org/resource/Lucas_Cranach_the_Elder>; ?p ?n}
  filter (?p in (
    wikidata:P214, dbo:viafid, dbo:viafId, # VIAF
    wikidata:P213,                         # ISNI
    wikidata:P646,                         # FreeBase
    wikidata:P244, prop-de:lccn,           # US LCNAF=LCCN
    wikidata:P245,                         # US ULAN
    wikidata:P227, dbo:individualisedGnd,  # DE GND
    wikidata:P268,                         # FR BnF
    wikidata:P650,                         # NL RKDartists
    wikidata:P1273                         # CAT CANTIC
  ))}
```

It is important to remember that in DBpedia the entity URL is changed to http://wikidata.dbpedia.org/resource/Q191748, and is declared `owl:sameAs` the DBpedia URL. `sameAs` is used instead of this "bastardized" wikidata URL

http://live.dbpedia.org/sparql includes more up to date information. The number of correlations can be counted.

```
prefix wikidata: <http://www.wikidata.org/entity/>
prefix dbo: <http://dbpedia.org/ontology/>
prefix prop-de: <http://de.dbpedia.org/property/>

select ?p ?t (count(*) as ?c) {
  ?x ?p ?y
  filter ( ?p in (
    wikidata:P214, dbo:viafid, dbo:viafId, # VIAF
    wikidata:P213,                         # ISNI
    wikidata:P646,                         # FreeBase
    wikidata:P244, prop-de:lccn,           # US LCNAF=LCCN
    wikidata:P245,                         # US ULAN
    wikidata:P227, dbo:individualisedPnd,  # DE GND
    wikidata:P268,                         # FR BnF
    wikidata:P650,                         # NL RKD
    wikidata:P1273                         # CAT CANTIC
  ))
  optional {?x a ?t1 filter (?t1 in (dbo:Person, dbo:Organisation))}
  optional {?x a ?t2 filter (?t2 in (dbo:Agent))}
  bind (coalesce (?t1,?t2) as ?t)
} group by ?p ?t order by desc(?c)
```

p	t	c
dbo:viafId	dbo:Person	262469
dbo:viafId	dno:Agent	1227
dbo:viafId	dbo:Organisation	255
dbo:individualisedPnd		3
Dbo:individualisedPnd	dbo:Person	3

Note: http://dbpedia.org/sparql returns only 16k

3.4.8 Finding Errors in Authorities through Wikipedia/Wikidata

The power of the crowd can help maintain authority control files by finding errors and researching cases where records should be merged or split. e.g. VIAF errors on Wikipedia has lists in the following categories:

- 1.1 Wikipedia article is not the same as the VIAF identity
- 1.2 Two or more VIAF identities for the same article
- 1.3 VIAF merges different identities (into one cluster)
- 1.4 Parallel VIAF clusters for one identity
- 1.5 Wikipedia link inside VIAF is out of date
- 1.6 Articles about multiple people assigned the VIAF identity for one of them
- 1.7 Other errors

Wikidata provides automatic integrity checking, e.g. no two items should have the same id, one item should have no more than one id, etc.).

- The VIAF ID constraint violations report lists some 3500 items that should be investigated.
- For example, Q192187 Communist Party of the Russian Federation (Gennady Zyuganov) had 6 VIAF ID's? A quick investigation in VIAF shows that only 146251554 is correct, whereas the rest represent subunits and conferences:
  - 233350017: a subunit: S︡entral'nyĭ komitet. Otdel po informat͡sionno-analiticheskoĭ rabote i provedenii͡a vybornykh kompaniĭ
  - 300667542: a conference: S︡entral'nyĭ komitet: 13th Plenum 2012
- A similar investigation was done for ULAN resulting in:
  - 9 candidates for merging in ULAN. Getty have already acted upon them
  - 25 candidates for merging in Wikidata, for example 500003014: Baldassare Estense (Q804745) vs. Baldassare D' Este (Q18507908)

4. Conclusions

The following conclusions from this analysis can be drawn:

- The best datasets to use for Person enrichment (NER) are VIAF and Wikidata
- The best approach is loading them to a local repository in order to ensure levels of service
- Names and other attributes (e.g. years, descriptions) are extracted with agreed queries, producing dynamic gazetteers
- For Wikidata files `terms, simple-statements, properties, taxonomy, instances` were loaded and only enwiki `sitelinks` for a total of maximum 315M triples.

Redundant triples are skipped, see

  - Wikidata Names : rdfs:label, schema:name. The ontologies were not loaded, to avoid the inference of rdfs:label from skos:prefLabel or skos:altLabel
  - It would be an idea worth thinking of, contacting the Wikidata developers to also emit one statement (the preferred or first in order) per item-property slot, even if the statement is qualified.

- For VIAF the file clusters-rdf.nt is loaded, about 300M triples.
- Two datasets are aligned by co-reference IDs.
- co-refParticipating in Co-referencing initiatives is beneficial for the Europeana community, and the wider CH and LOD communities. For example, a first initiative could be to cross-check VIAF Wikipedia links against Wikidata VIAF links

5. References

1. Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, Christian Bizer, DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. Semantic Web Journal, 2013.
2. Anila Angjeli, Andrew Mac Ewan and Vincent Boulet, ISNI and VIAF - Transforming ways of trustfully consolidating identities. IFLA 2014, July 2014.
3. Thomas B. Hickey and Jenny A. Toves, Managing Ambiguity In VIAF, D-Lib Magazine, Volume 20, Number 7/8, July/August 2014. 10.1045/july2014-hickey
4. Maximilian Klein and Alex Kyrios, VIAFbot and the Integration of Library Data on Wikipedia, Code4Lib Journal, Issue 22, 2013-10-14
5. Fredo Erxleben, Michael Günther, Markus Krötzsch, Julian Mendez and Denny Vrandecic, Introducing Wikidata to the Linked Data Web, 2014
6. Maximillian Klein, Authority Addicts: The New Frontier of Authority Control on Wikidata, (17 slides: pptx presentation, SlideShare). Wikimania 2013 International Wikimedia Conference, 7-11 August 2013, Hong Kong
7. Vladimir Alexiev, Joan Cobb, Gregg Garcia, and Patricia Harpring. Getty Vocabularies Linked Open Data: Semantic Representation. Manual, Getty Research Institute, 2.0 edition, August 2014.
8. Wikidata identifiers and the ODNB – where next?, blog, 26 November 2014
9. Wikidata and identifiers – part 2, the matching process, blog, 27 November 2014