FAGI-gis version 1.1

1. Overview

FAGI-gis is a tool developed, mainly, to facilitate the fusion of interlinked RDF entities containing spatial data. It is designed to retrieve data through SPARQL endpoints. This allows for FAGI-gis to operate on already existing and publicly available datasets without the need for any special formatting or input. It also supports the fusion and handling of other, non-spatial metadata related to these entities.

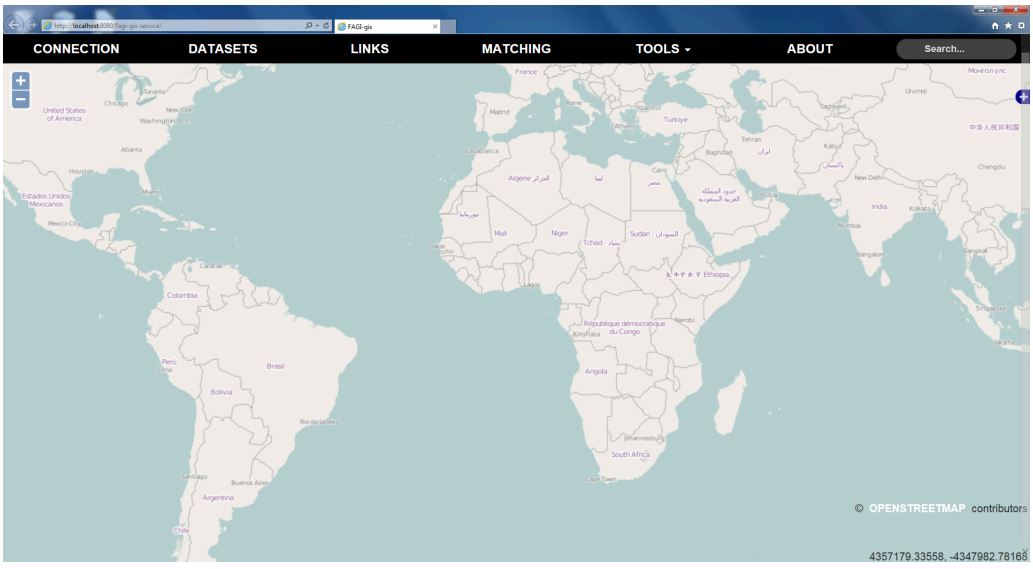
The user provides the tool with two source datasets and a list of linked entities between them, either in file format or through an available SPARQL endpoint(Section 2.5.3 specifies the required format). The tool analyzes the datasets, discovering how geometric information is stored along with their accompanied metadata. Knowing the data structure, FAGI-gis offers the user various options and recommendations for fusing each entity pair into a new, fused, richer entity.

The tool supports an interactive interface offering visualization of the data at every part of the process. Especially for spatial data, it provides map previewing and graphical manipulation of geometries.

FAGI-gis provides advanced fusion functionality through batch mode fusion, clustering of links, link discovery/creation, property matching, property creation etc. In what follows, a step-by-step demonstration of the offered facilities is provided.

2. Usage

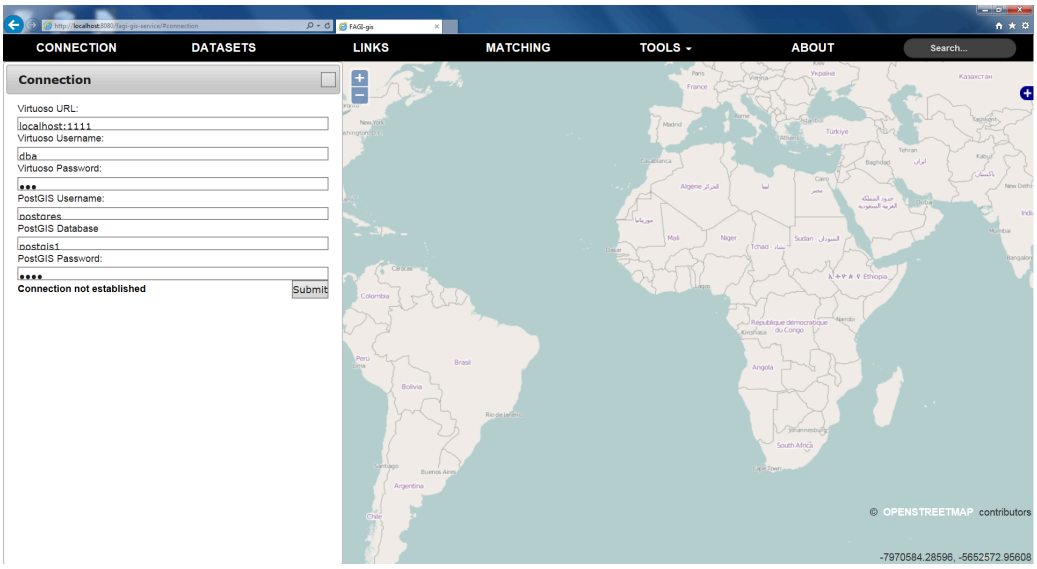
The interface for FAGI consists of a menu bar and the map component. Below, we present the functionality by each tab of the menu.



2.1. Connection

The **'Connection'** tab allows the specification of credentials for connecting to the required databases installed on the system, namely, Virtuoso and PostGIS. These are configured locally during installation of FAGI-gis and are essential for the fusion process. Clicking on submit will attempt connection to the databases.

- Virtuoso URL specifies the address and port of the local Virtuoso instance that can be used as a backend for manipulating RDF data locally.
- Virtuoso Username/Pass are the credentials for that same local Virtuoso instance. FAGI-gis requires the user to have Administrative privileges. This is needed in order to perform some actions more efficiently
- PostGIS Username/Pass are the credentials for the running PostgreSQL/PostGIS database. The user needs to be a Superuser in order to be able to create the needed GIS extensions and databases
- PostGIS Database is the name of the database that FAGI-gis is going to create for storing and handling the spatial data from the datasets.

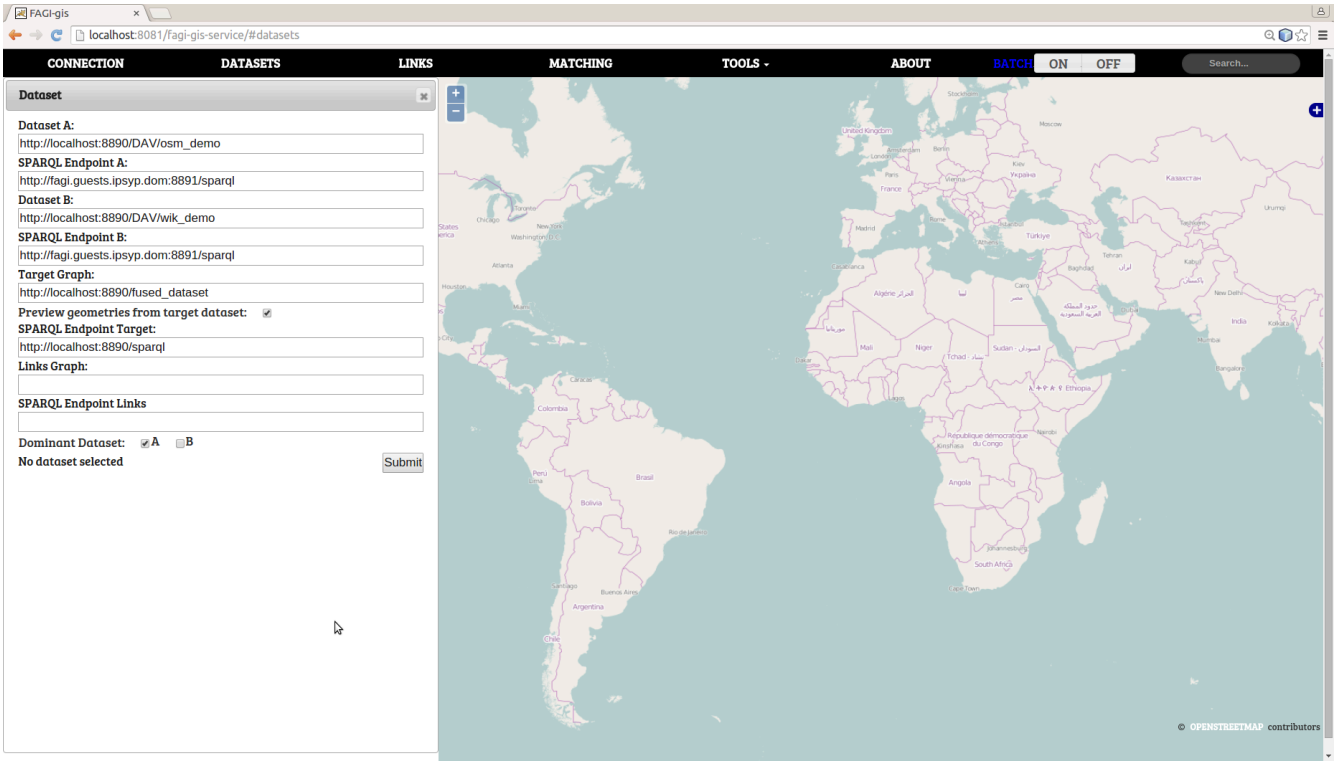


2.2. Datasets

The **'Datasets'** tab is for specifying the dataset input for FAGI-gis. A dataset is a pair consisting of the RDF graph name and the SPARQL service hosting the graph. FAGI-gis refers to these two datasets as dataset A and dataset B. Clicking on submit will validate the input and perform additional actions.

- Dataset A and B specify the source datasets and their corresponding SPARQL service.
- Target dataset is the destination for the outputting the final, fused graph. The target endpoint needs to be an endpoint that gives the SPARQL Update rights to the SPARQL user. A user can choose to preview any already fused geometry from the target dataset.
- Links dataset indicates the graph containing the links between entities. Leaving the links graph/endpoint pair blank(empty) means that the links will be provided through a file in the next phase.
- Dominant dataset indicates which dataset is considered to be the more accurate, meaning

that its information should be preferred over the other' s during fusion. In the process, the subject used for any generated triple is that corresponding to the dominant dataset. The chosen dataset also provides the dominant RDF ontology for newly created entities.



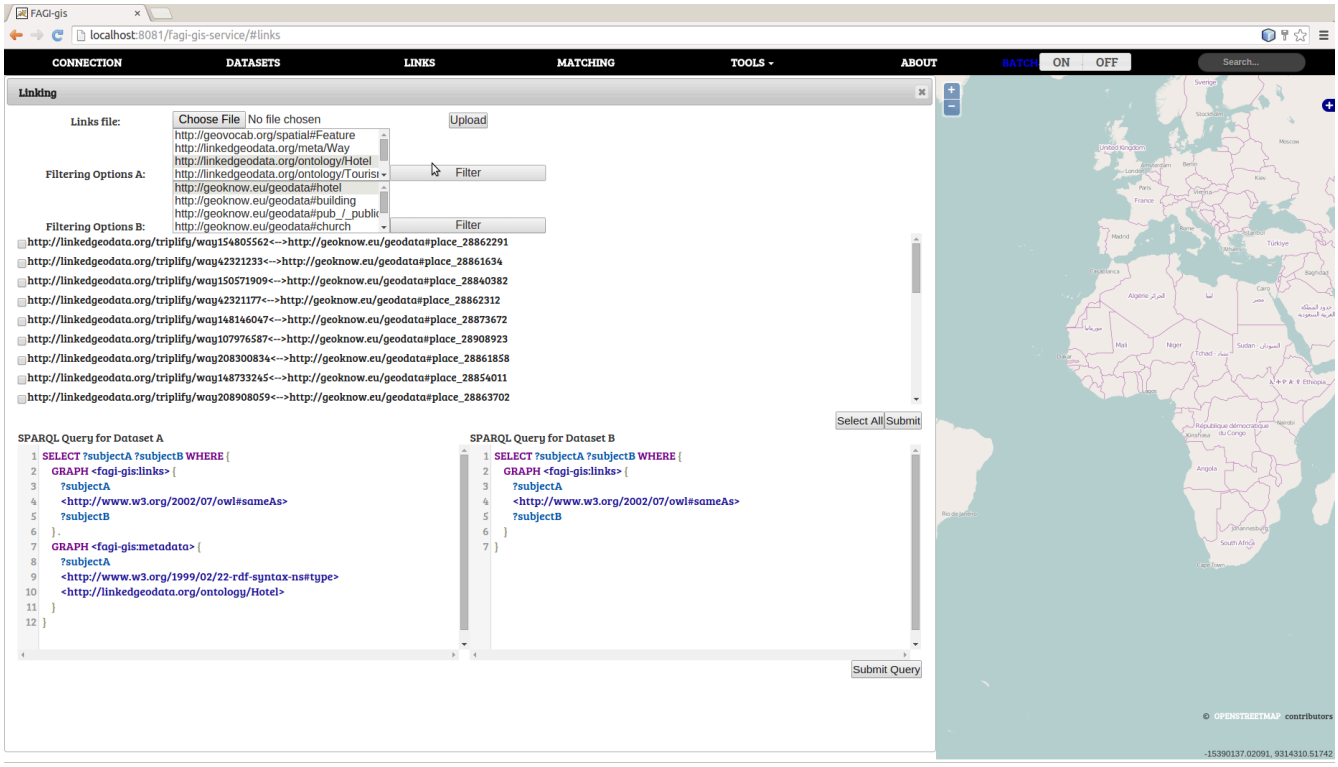
2.3. Links

The '**Links**' tab is for uploading and handling the links. Links are provided either through the graph/service pair specified in the '**Datasets**' tab or through a file in N3 format.

After supplying the links, a list of checkable items is generated in order to select the desired links for final fusion.

Links can also be filtered by type allowing for better control and categorization of the linked entities. The available types of each dataset are discovered by FAGI-gis and provided in the filter selection windows. Selecting multiple types is also possible. FAGI-gis also provides the means to filter the links through custom SPARQL queries, guiding the user by providing a simple, partially fixed structure.

On clicking 'Submit', FAGI-gis will perform *schema matching* on the properties of the linked entities in order to recommend possible matchings between the properties of dataset A and B.



2.4. Schema Matching

The '**Schema Matching**' tab offers recommendations for metadata fusion pairs based on various metrics. Only the property name of each triple is considered when calculating a similarity score.

In this phase, schema matching is done on a subset of the entities graph and allows declaring the property pairs that should be considered for fusion on each of the linked entities. In a later phase, property selection can be done on a per-link level allowing for more fine-grained handling of property matching.

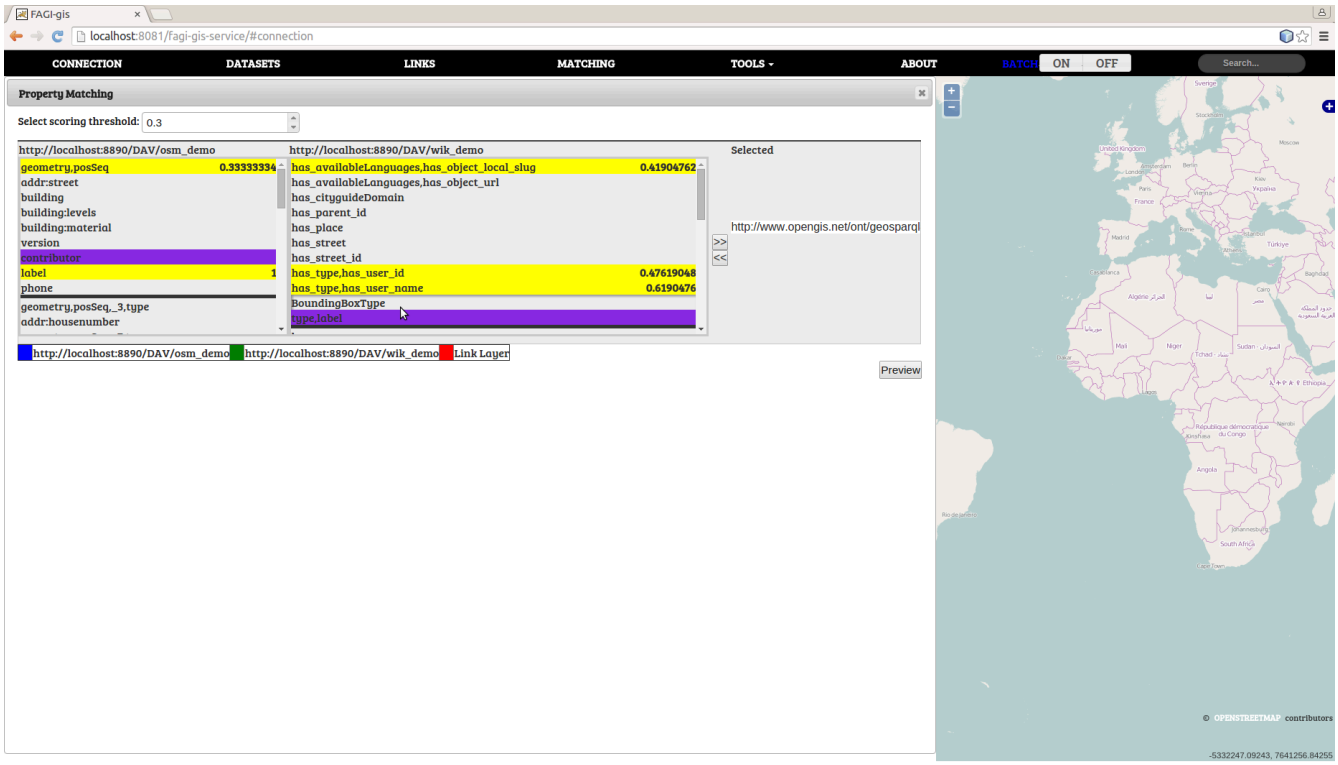
The first list contains property chains from Dataset A and the second from Dataset B (see Section 2.5.3 for the definition of a property chain). Properties above the separator lines indicate that a match has been found between the specific property and some property(ies) of the other dataset. By selecting a matched property from the list, the proposed matches are highlighted with the matching score presented. You can select multiple properties from each list using Left-Click + Ctrl-Key to create a big variety of possible fusion pairs.

After clicking the right arrow (>>) the selections are submitted to the the selected-for-fusion list (the geometry property is always included). By default, for naming the new property, FAGI uses the concatenation of the selected values separated by the symbol (=>). According to the subsequent metadata fusion action, the tool uses this separator to decide on the final property names. The user is free to supply his own name for the new property by editing the appropriate field.

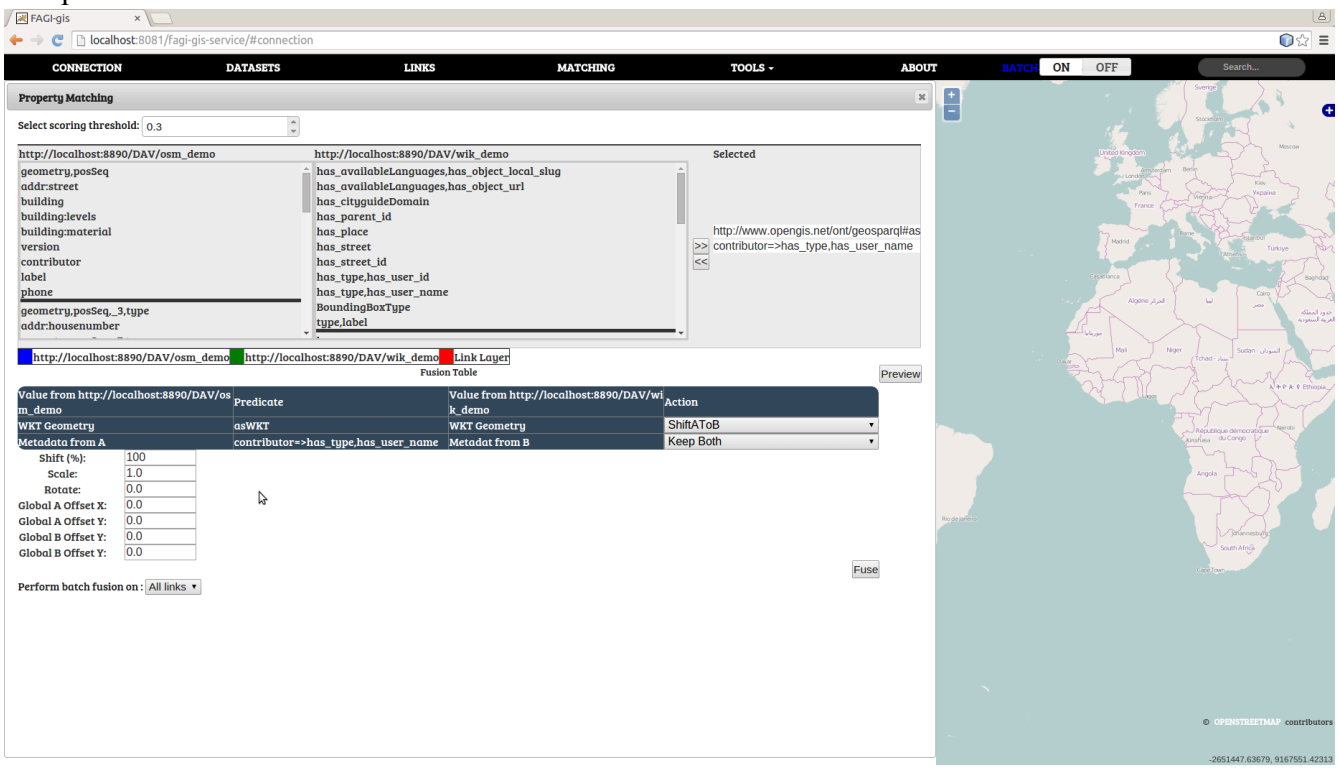
Clicking the *Preview* button will visualize the linked entities on the map.

Additionally, one can choose to turn on the *Batch Fusion* mode of FAGI-gis and get a selection of options for performing fusion on each of the linked entities. The table contains the property pair information and a list of available actions. More user supplied options for the fusion of geometric data are given below along with the ability to further filter the link entities that get fused in the end.

Select a property pair for fusion. Adding it to the selected-for-fusion list will include it for fusion in each of the linked entities

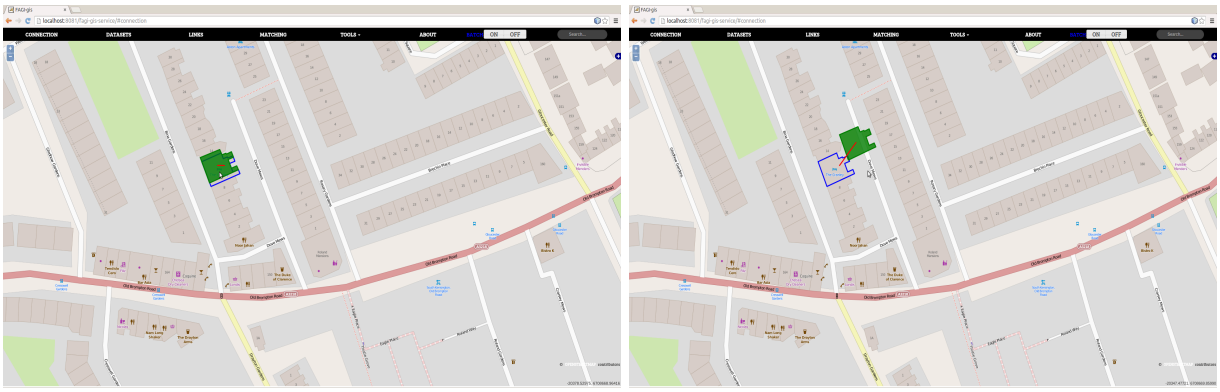


Turning Batch Fusion Mode ON brings up the options for batch fusion of entities. Clicking on fuse performs the selected actions on all links or a selected cluster.

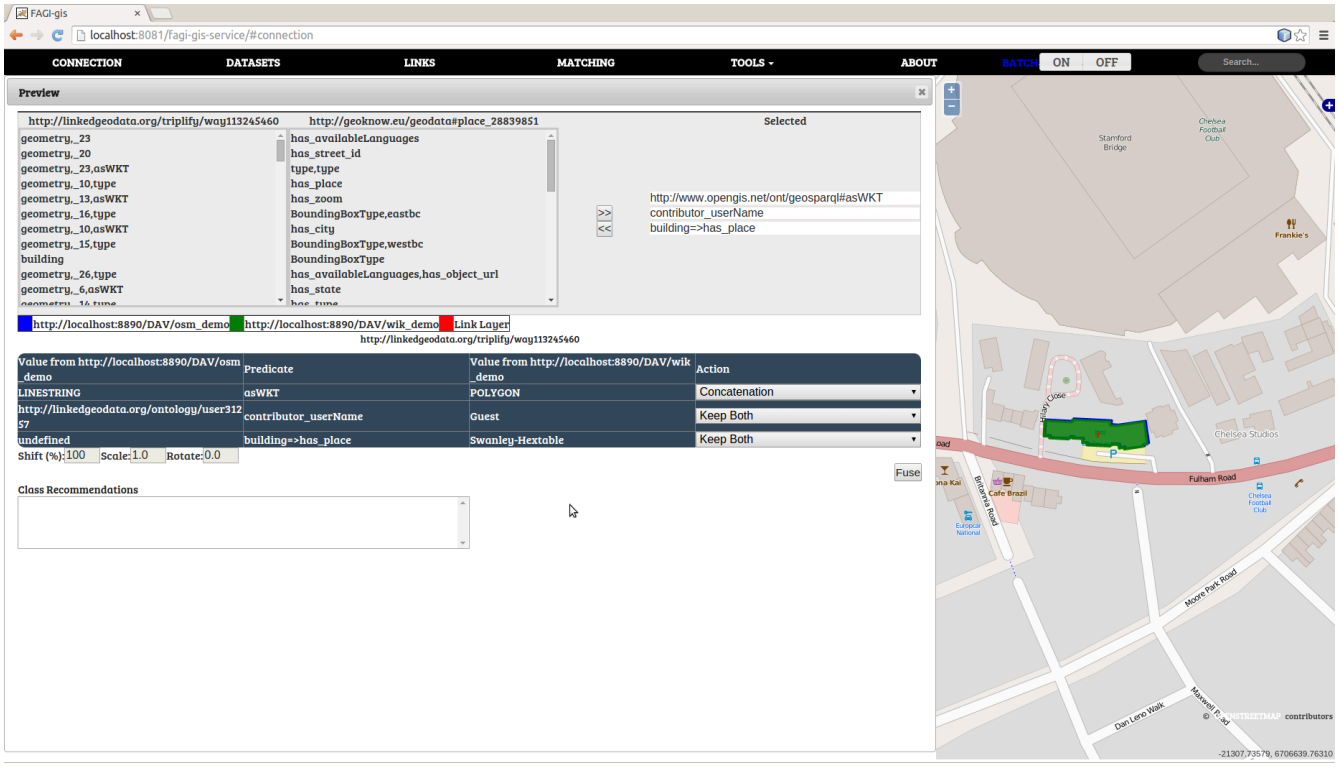


2.5. Fusion

After previewing, one can see the linked entities on the map. A thin red line is drawn between two linked geometries. At this point the user can freely reposition the geometry of an entity just by dragging the previed object. Any shifting action will be respected during fusion of the linked objects. When shifting a geometry, a global offset for each of the dimensions is calculated. These offsets can be seen in the '**Schema Matching**' and if desirable, can be used for offsetting every linked pair with the same values.

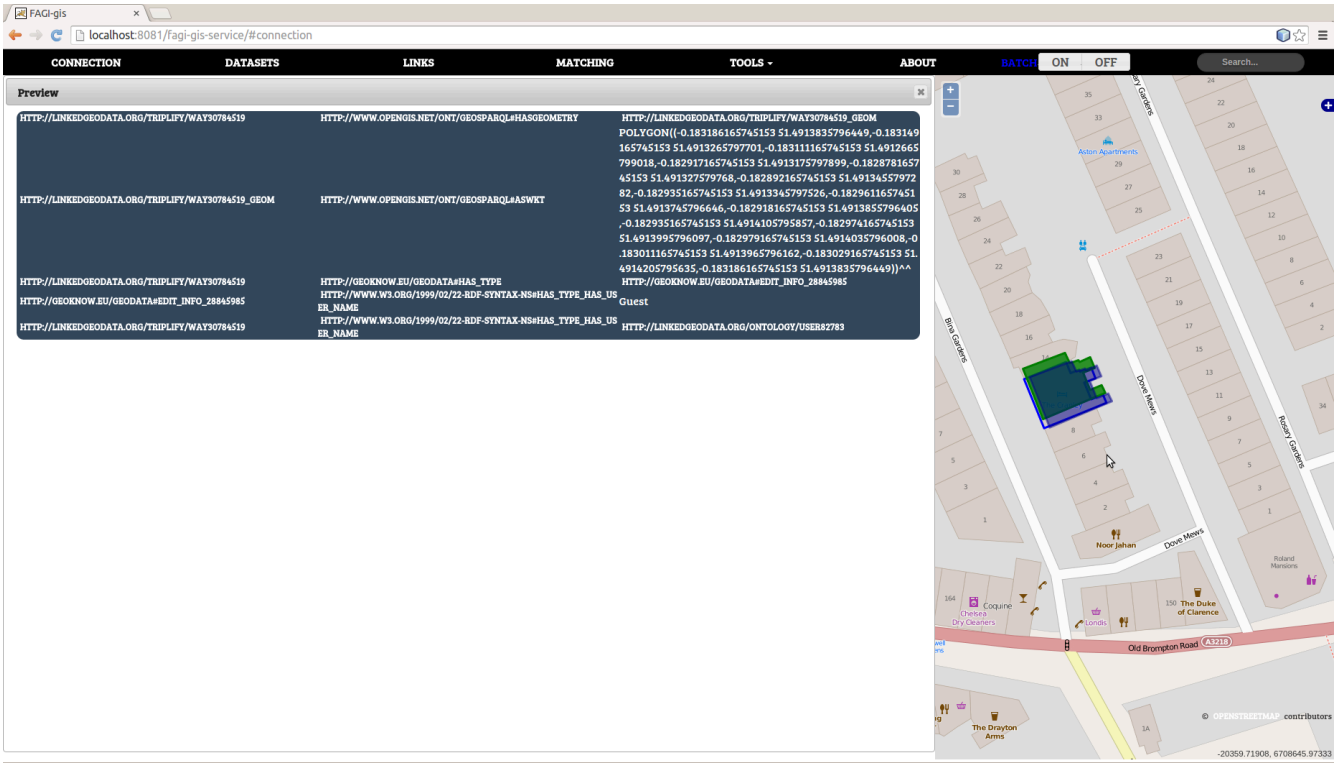


Clicking on the connecting line will bring the per-link fusion panel to the front. Similarly to the '**Schema Matching**' panel, the list of all the properties of each entity is provided and one can include more specific pairs of properties for fusion in addition to the ones selected before. The table is always present and now the value of each selected property is shown.



Clicking the Fuse button will execute the selected fusion actions and store them in the “target” graph of the previously specified SPARQL service. The process can be repeated similarly for every other linked entity pair. If the provided graph and endpoint resides in a local instance, FAGI can greatly optimize the creation of the new graph without the need to perform insertions and deletions through compliant SPARQL queries.

Clicking on the resulting fused entity, will open a panel previewing the triples of the new entity. The source geometries are also previewed as highly transparent.



2.5.1. Implemented Fusion Actions

The table below describes the available fusion actions:

Action	Type	Description
Concatenation	Geometric	Create a GEOMETRYCOLLECTION Well-Known-Text from the two geometries
ShiftAtoB/BtoA	Geometric	Shift one geometry in the direction of the other' s center
Keep A/B	Geometric	Keep only the geometry of one of the datasets
Keep most points	Geometric	Keep most complicated

		geometry
Keep Both	Geometric	Keep both geometries seperately
Keep Concatenated A/B	Metadata	Keep a concatenation of the values corresponding to each property in one single property
Concatenation	Metadata	Perform the above action on both properties and keep a concatenation of the results
Keep Concatenated Both	Metadata	Perform the above action on both properties and keep both results
Keep Flattened A/B	Metadata	With properties that are part of a long RDF triple chain this action allows for creating triples that move the values of the selected property one depth level less in the chain.
Keep Flattened Both	Metadata	Perform the above action on both properties and keep both results
Keep A/B	Metadata	Keep only the value from one of the datasets
Keep Both	Metadata	Keep both separate values

Table 1. Fusion Action Table

Refer to the **Fusion Action Examples** section for a more comprehensive explanation of the actions.

2.5.2. Additional FAGI-gis Tools

Currently, FAGI-gis supports the following additional actions

- **Clustering**. The tool offers the ability to perform clustering on the links based on a selection among the three attributes:
  - **Vector Length.** Euclidean distance between the centroids of the two geometries normalized by the max distance.
  - **Vector Direction.** Coordinates of the vector connecting the centroids of the two geometries, normalized by its length
  - **Coverage.** Binary attribute indicating whether the polygons overlap.

  The number of clusters is either calculated automatically or selected by the user using the slider (max = 10). The calculated clusters are previewed on the map using different colors for links that belong to different clusters. Going back to the **'Schema Matching'** panel after assigning clusters to the links, allows you to perform batch actions on one of the specific

clusters.

- **Multiple Select**. This facility allows the creation of a custom group of linked entities for batch fusion. It allows to create the group either by selecting individual links or by bounding box selection.
- **Link Finding/Creation**. Provided that the remote endpoints are GeoSPARQL compliant, FAGI-gis allows the user to perform additional actions by fetching geometric data from entities of the two datasets that are *not interlinked*. Firstly, this allows FAGI to function as a general geometry visualization tool for data that are stored in remote endpoints. It also allows the user to visually discover and create links between previously unlinked entities of the two datasets. This operation is referred to as **Link Creation** and provides the ability to perform fusion on new pairs of entities. To supplement link creation, FAGI offers **Link Finding** by searching for links in a bounded area provided by the user. Using various metrics and methods the tool visually proposes some links to the user and gives him the ability to individually validate one or all of them. FAGI can also search the neighborhood of any individual unlinked entity and propose links.

2.5.3. Fusion Action Examples

Currently, FAGI-gis only supports geometries that are in WKT. The triples of the datset must be in the form

<sub> <geoProperty> <subGeom> <subGeom> <http://www.opengis.net/ont/geosparql#asWKT> “WKT geometry”

Geometry Triple

The provided links either through a file or an endpoint should be provided in the following form.

<subA> <http://www.w3.org/2002/07/owl#sameAs> <subB>

Link Triple

Declaration	Definition	Example

Property Chain	A property chain consists of the *predicate* values of the triple patterns that lead to an object(URI or Literal). They help in understanding the structure of an RDF graph and optimizing queries	Taking the supported Geometry Triple for FAGI as an example, the property chain, leading to the geometric data is geoProperty,http://www.opengis.net/ont/geosparql#asWKT

- Consider Dataset i

Dataset A	**Link**	**Dataset B**
<subA> <propA> “objA”	<subA> <same> <subB>	<subB> <propB> “objB”

Dataset i

Assuming the user has matched <propA> with <propB>, **Keep A** action will keep the first triple and **Keep B** will keep the second triple. **Keep Both** will keep both of them. The important thing is that the resulting triple will have the subject of the dominant dataset as selected by the user.

Using the above dataset, selecting **Concatenation** will produce a triple of the form

<subA> <newProperty> “objA objB”

The newProperty will be a concatenation of name of the property from the dominant dataset, (as specified earlier) and the value the user provided during matching of the properties in the ‘**Schema Matching**’ panel.

- Consider Dataset ii

Dataset A	**Link**	**Dataset B**
<subA> <propobjA1” <subA> <propA> “objA2” <subA> <propA> “objA3”	<subA> <same> <subB>	<subB> <propB> “objB1” <subB> <propC> “objB2”

Dataset ii

_____If someone matches <propA> with <propB> then a **Keep Concatenated A** action will produce

the following triple

<subA> <newProperty> “objA1 objA2 objA3”

whereas, **Keep Concatenated B** will produce

<subA> <newProperty> “objB1”

Concatenated Both will insert both these new triples.

Lastly, going back to **Concatenation,** the resulting triple will be

<subA> <newProperty> “objA1 objA2 objA3 objB1”

Notice that the second dataset has two distinct properties (<propB>, <propC>). As mentioned, FAGI supports 1-to-1 mapping of properties as well as m-to-n mapping. Therefore, one could choose to fuse <propA> with {<propB>,<propC>}.

This would produce the following triples for **Keep Concatenated B**

<subA> <newProperty> “objB2 objB1”

and **Concatenation**

<subA> <newProperty> “objA1 objA2 objA3 objB1 objB2”

- Consider Dataset iii

Dataset A	**Link**	**Dataset B**
<subA> <propA1> <objA> <objA> <propA2> “objA2” <objA> <propA3> “objA3”	<subA> <same> <subB>	<subB> <propB1> <objB1> <objB1> <propB2> <objB2> <objB2> <propB3> “objB3”

Dataset iii

With <propA1> and <propB1> matched, this is where property chains come into play. Observe that two second level chains are formed for <subA> and one third level chain for <subB>. Inside FAGI, these are represented as

- propA1,propA2 and propA1,propA3
- propB1,propB2,propB3

Using the above, **Keep Flattened A** will create the following triples

<subA> <newPropertyA> “objA2”
<subA> <newPropertyB> “objA3”

<newPropertyA> and <newPropertyB> keep the same naming convention as before but, also include the *suffix* of the previous property chain. In most cases this the part after the ‘#’ or ‘/’ characters in the URI. This is a useful operation for lowering the depth of a property chain.

Keep Flattened B will create the following triple

<subA> <newProperty> “objB3”

Similarly, <newProperty> will contain the new property value including the ***name*** part of the final property in the chain (<propB3>).

In **Concatenation** and **Flattening** actions, only RDF Literals are considered.