An Efficient RDF Converter and SPARQL Endpoint for the Complete OpenStreetMap Data

Hannah Bast, Patrick Brosi, Johannes Kalmbach, Axel Lehmann

University of Freiburg

ACM SIGSPATIAL 2021
November 2-5, 2021

Motivation - Working with OpenStreetMap data

Goal: Efficient and user-friendly querying of OpenStreetMap (OSM) data.

Motivation - Working with OpenStreetMap data

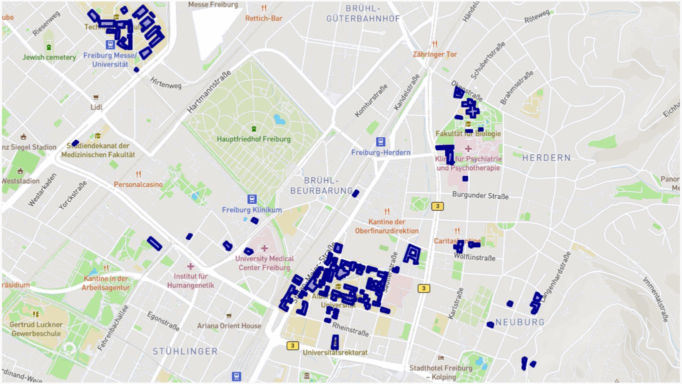
Goal: Efficient and user-friendly querying of OpenStreetMap (OSM) data.

Example: *All university buildings in Freiburg*.

Motivation - Working with OpenStreetMap data

Goal: Efficient and user-friendly querying of OpenStreetMap (OSM) data.

Example: *All university buildings in Freiburg.*



Existing Tools and Our Work

	Completeness	Efficiency	Queries	Interoperability
OSM Tools	+	○ pass over whole data	○ limited functionality	○ OSM format
Overpass	+	○ limited indexes	○ arcane query language	○ OSM format
PostGIS	○ limited number of columns	○ limited indexes	○ queries hard to formulate	○ instance-specific configuration
Sophox	○ no shape info	○ Blazegraph is slow	○ SPARQL w/o autocomplete	+ RDF
Ours (osm2rdf + QLever)	+	○ work in progress	+ SPARQL w. autocomplete	+ RDF

OSM Data

```
<node id="831225088" lat="48.002756" lon="7.848349"/>
<node id="831224996" lat="48.0026865" lon="7.848733"/>
<node id="1061500830" lat="48.0025018" lon="7.8482462"/>
[...]
<way id="69367089">
  <nd ref="831225088"/>
  <nd ref="831224996"/>
  [...]
  <nd ref="1061500830"/>
  <nd ref="831225088"/>
  <tag k="name" v="Mensa Institutsviertel"/>
  <tag k="building" v="university"/>
</way>
```

OSM Data

```
<node id="831225088" lat="48.002756" lon="7.848349"/>
<node id="831224996" lat="48.0026865" lon="7.848733"/>
<node id="1061500830" lat="48.0025018" lon="7.8482462"/>
[...]
<way id="69367089">
 <nd ref="831225088"/>
 <nd ref="831224996"/>
 [...]
 <nd ref="1061500830"/>
 <nd ref="831225088"/>
 <tag k="name" v="Mensa Institutsviertel"/>
 <tag k="building" v="university"/>
</way>
```

- Build of nodes (points), ways (polylines) and relations (groups thereof)

OSM Data

```
<node id="831225088" lat="48.002756" lon="7.848349"/>
<node id="831224996" lat="48.0026865" lon="7.848733"/>
<node id="1061500830" lat="48.0025018" lon="7.8482462"/>
[...]
<way id="69367089">
 <nd ref="831225088"/>
 <nd ref="831224996"/>
 [...]
 <nd ref="1061500830"/>
 <nd ref="831225088"/>
 <tag k="name" v="Mensa Institutsviertel"/>
 <tag k="building" v="university"/>
</way>
```

- Build of nodes (points), ways (polylines) and relations (groups thereof)
- Explicit spatial information only for nodes

OSM Data

```
<node id="831225088" lat="48.002756" lon="7.848349"/>
<node id="831224996" lat="48.0026865" lon="7.848733"/>
<node id="1061500830" lat="48.0025018" lon="7.8482462"/>
[...]
<way id="69367089">
 <nd ref="831225088"/>
 <nd ref="831224996"/>
 [...]
 <nd ref="1061500830"/>
 <nd ref="831225088"/>
 <tag k="name" v="Mensa Institutsviertel"/>
 <tag k="building" v="university"/>
</way>
```

- Build of nodes (points), ways (polylines) and relations (groups thereof)
- Explicit spatial information only for nodes
- Objects can have arbitrary attributes

OSM Data

```
<node id="831225088" lat="48.002756" lon="7.848349"/>
<node id="831224996" lat="48.0026865" lon="7.848733"/>
<node id="1061500830" lat="48.0025018" lon="7.8482462"/>
[...]
<way id="69367089">
 <nd ref="831225088"/>
 <nd ref="831224996"/>
 [...]
 <nd ref="1061500830"/>
 <nd ref="831225088"/>
 <tag k="name" v="Mensa Institutsviertel"/>
 <tag k="building" v="university"/>
</way>
```

- Build of nodes (points), ways (polylines) and relations (groups thereof)
- Explicit spatial information only for nodes
- Objects can have arbitrary attributes
- Ways may be interpreted as polygons (areas)

OSM Data

```
<node id="831225088" lat="48.002756" lon="7.848349"/>
<node id="831224996" lat="48.0026865" lon="7.848733"/>
<node id="1061500830" lat="48.0025018" lon="7.8482462"/>
[...]
<way id="69367089">
 <nd ref="831225088"/>
 <nd ref="831224996"/>
 [...]
 <nd ref="1061500830"/>
 <nd ref="831225088"/>
 <tag k="name" v="Mensa Institutsviertel"/>
 <tag k="building" v="university"/>
</way>
```

- Build of nodes (points), ways (polylines) and relations (groups thereof)
- Explicit spatial information only for nodes
- Objects can have arbitrary attributes
- Ways may be interpreted as polygons (areas)
- Historically grown

RDF Data

```
osmway:69367089  geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                 osmkey:name       "Mensa Institutsviertel";
                 osmkey:building   "university".
osmway:91332395  geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                 osmkey:name       "Rechenzentrum";
                 osmkey:building   "university".
```

RDF Data

```
osmway:69367089   geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                  osmkey:name       "Mensa Institutsviertel";
                  osmkey:building   "university".
osmway:91332395   geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                  osmkey:name       "Rechenzentrum";
                  osmkey:building   "university".
```

- Data represented as subject-predicate-object triples

RDF Data

```
osmway:69367089  geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                 osmkey:name       "Mensa Institutsviertel";
                 osmkey:building   "university".
osmway:91332395  geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                 osmkey:name       "Rechenzentrum";
                 osmkey:building   "university".
```

- Data represented as subject-predicate-object triples
- All attribute keys converted into predicates, explicit shapes for each object

RDF Data

```
osmway:69367089  geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                 osmkey:name       "Mensa Institutsviertel";
                 osmkey:building   "university".
osmway:91332395  geo:hasGeometry   "MULTIPOLYGON(...)"^^geo:wktLiteral;
                 osmkey:name       "Rechenzentrum";
                 osmkey:building   "university".
```

- Data represented as subject-predicate-object triples
- All attribute keys converted into predicates, explicit shapes for each object
- Easy representation and interoperability (just concatenate)

Overpass API

Find all OSM relations with identical `wikidata=` tags:

```
rel[wikidata]->.a;
foreach .a -> .b (
  rel.a(if:t["wikidata"] == b.u(t["wikidata"]))->.d;
  rel.d(if: id() == d.min(id()) && d.count(relations) > 1 );
  convert rel ::id = _.u(id()),
          wikidata = _.u(t["wikidata"]),
          all_relation_ids = d.set( id() );
  (._ ; .result; ) -> .result;
);

.result out;
```

Overpass API

Find all OSM relations with identical `wikidata=` tags:

```
rel[wikidata]->.a;
foreach .a -> .b (
  rel.a(if:t["wikidata"] == b.u(t["wikidata"]))->.d;
  rel.d(if: id() == d.min(id()) && d.count(relations) > 1 );
  convert rel ::id = _.u(id()),
          wikidata = _.u(t["wikidata"]),
          all_relation_ids = d.set( id() );
  (._ ; .result; ) -> .result;
);

.result out;
```

- Script-like query language

Overpass API

Find all OSM relations with identical `wikidata=` tags:

```
rel[wikidata]->.a;
foreach .a -> .b (
  rel.a(if:t["wikidata"] == b.u(t["wikidata"]))->.d;
  rel.d(if: id() == d.min(id()) && d.count(relations) > 1 );
  convert rel ::id = _.u(id()),
          wikidata = _.u(t["wikidata"]),
          all_relation_ids = d.set( id() );
  (._ ; .result; ) -> .result;
);

.result out;
```

- Script-like query language
- Arcane syntax, difficult to read and learn

Overpass API

Find all OSM relations with identical `wikidata=` tags:

```
rel[wikidata]->.a;
foreach .a -> .b (
  rel.a(if:t["wikidata"] == b.u(t["wikidata"]))->.d;
  rel.d(if: id() == d.min(id()) && d.count(relations) > 1 );
  convert rel ::id = _.u(id()),
          wikidata = _.u(t["wikidata"]),
          all_relation_ids = d.set( id() );
  (._ ; .result; ) -> .result;
);

.result out;
```

- Script-like query language
- Arcane syntax, difficult to read and learn
- Query times out after 3 minutes

Query Performance

Q1 All university buildings

Q2 All university buildings in the bounding box of Germany

Q3 All university buildings in the bounding box of Freiburg

Q4 All university buildings in Freiburg

Query Performance

Q1 All university buildings

Q2 All university buildings in the bounding box of Germany

Q3 All university buildings in the bounding box of Freiburg

Q4 All university buildings in Freiburg

	RDF + QLever	Overpass API	PostGIS	Sophox
Q1	61ms	—	7m	8s
Q2	390ms	—	6m	—
Q3	391ms	169ms	100ms	4.5s*
Q4	134ms	300ms	188ms	n/a

Table 1: Running time of the example queries. A — means a timeout. Query Q3 on Sophox only considers the centroids (*).

URLs

Weekly updated RDF datasets, evaluation queries and SPARQL endpoint:

`https://osm2rdf.cs.uni-freiburg.de/`

Code:

`https://github.com/ad-freiburg/osm2rdf`

`https://github.com/ad-freiburg/qlever`

Thank you!

Thank you!