

```
<xs:complexType name="CategoryType">
<xs:sequence>
  <xs:element name="description" type="xs:string" />
  <xs:element name="category" type="CategoryType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="books">
      <xs:sequence>
        <xs:element name="link" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

Application of the Linked Data Visualization Model on Real World Data from the Czech LOD Cloud

Jakub Klímek

klimek@fit.cvut.cz

Faculty of Information Technology
Czech Technical University in Prague

Jiří Helmich

Martin Nečaský

<http://xrg.cz> | contact@xrg.cz

XML and Web Engineering Research Group

Faculty of Mathematics and Physics

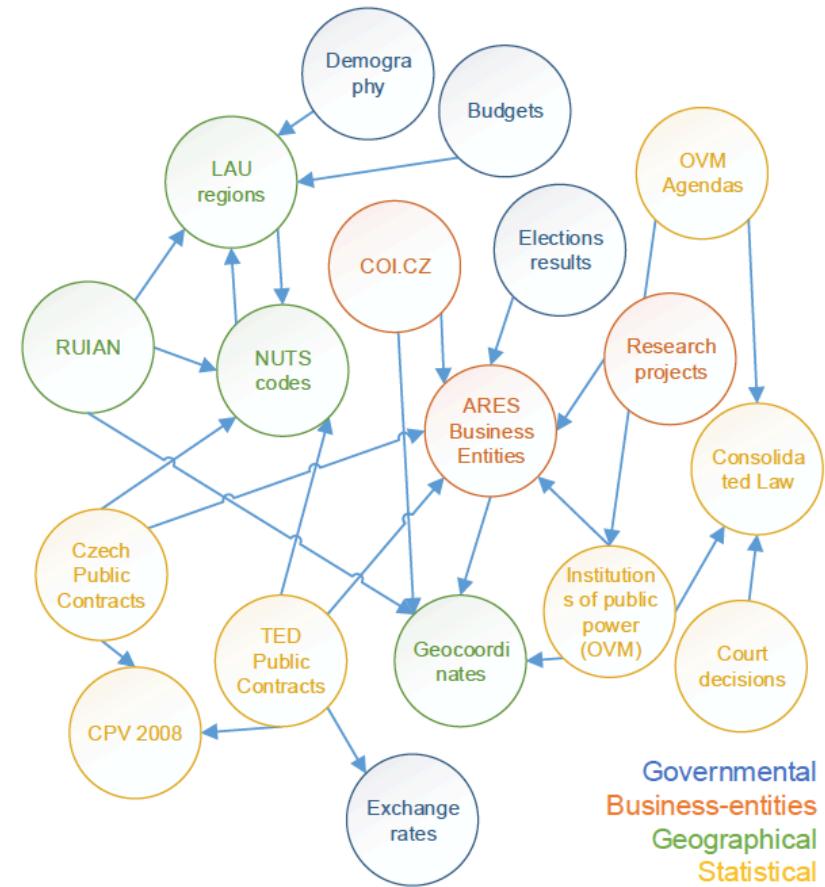
Charles University, Prague

Czech Republic



CzLOD cloud

- ❑ OpenData.cz since 2012
- ❑ 100M triples
- ❑ RUIAN 600M triples
- ❑ Virtuoso 7 instance



<http://linked.opendata.cz/sparql>

CzLOD cloud

- Business entities
 - ARES – Business registry
 - COI.CZ – Czech Trade Inspection Agency
 - Research projects
- Geographical data
 - RUIAN – territorial identification linked to LAU & NUTS
 - Geocoding results

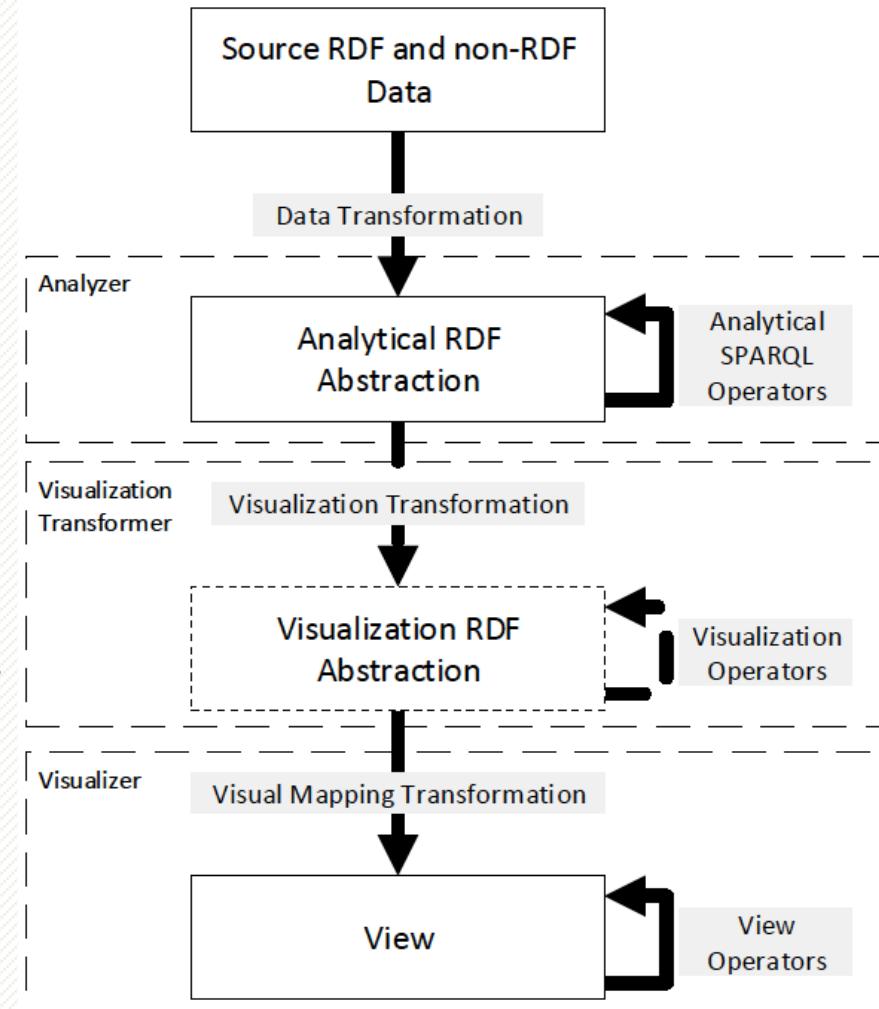
CzLOD cloud

- Governmental datasets
 - OVM – ministries, cities, notaries, ...
 - Agendas (opening hours)
 - Laws - decisions of the Czech Supreme court
- Statistical datasets
 - Demography
 - Budgets
 - Exchange rates – European Central Bank

LDVM

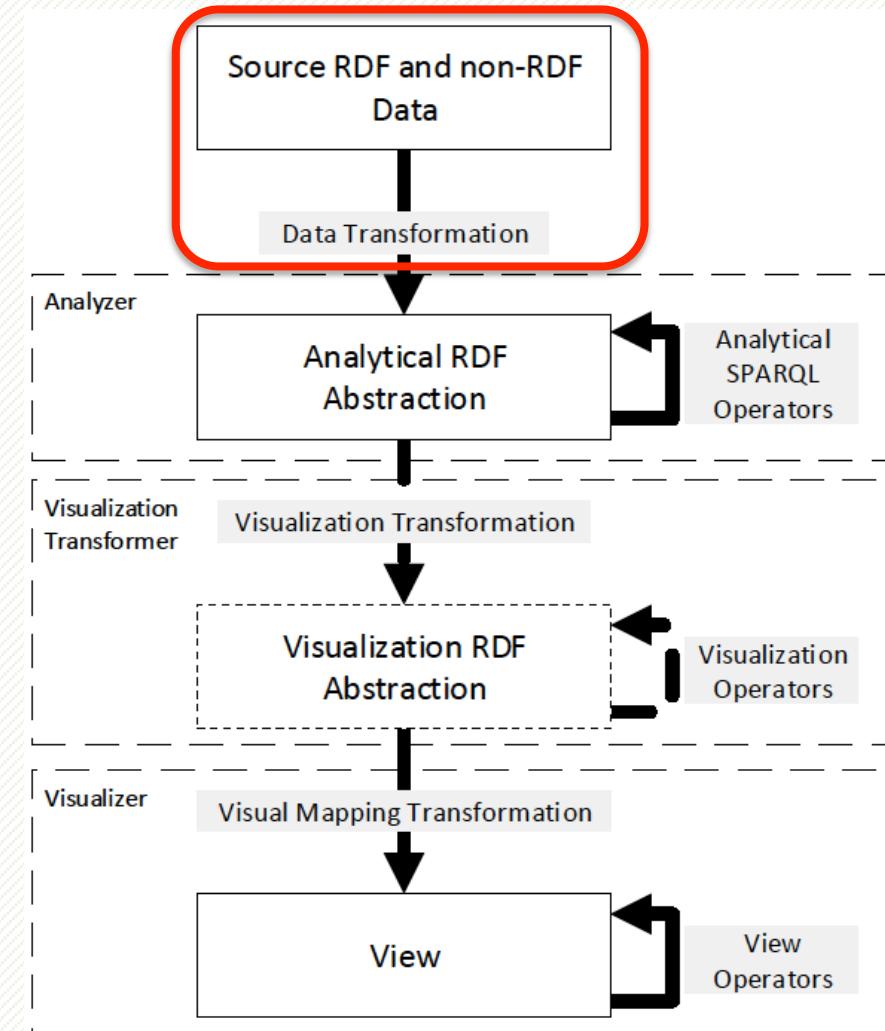
- ❑ Abstract process
 - 4 stages
 - 3 transformation types
 - 3 operator types

- ❑ Adapted for RDF
 - Input transformed to RDF
 - Internally works with RDF
 - Existing visualizers accept proprietary formats



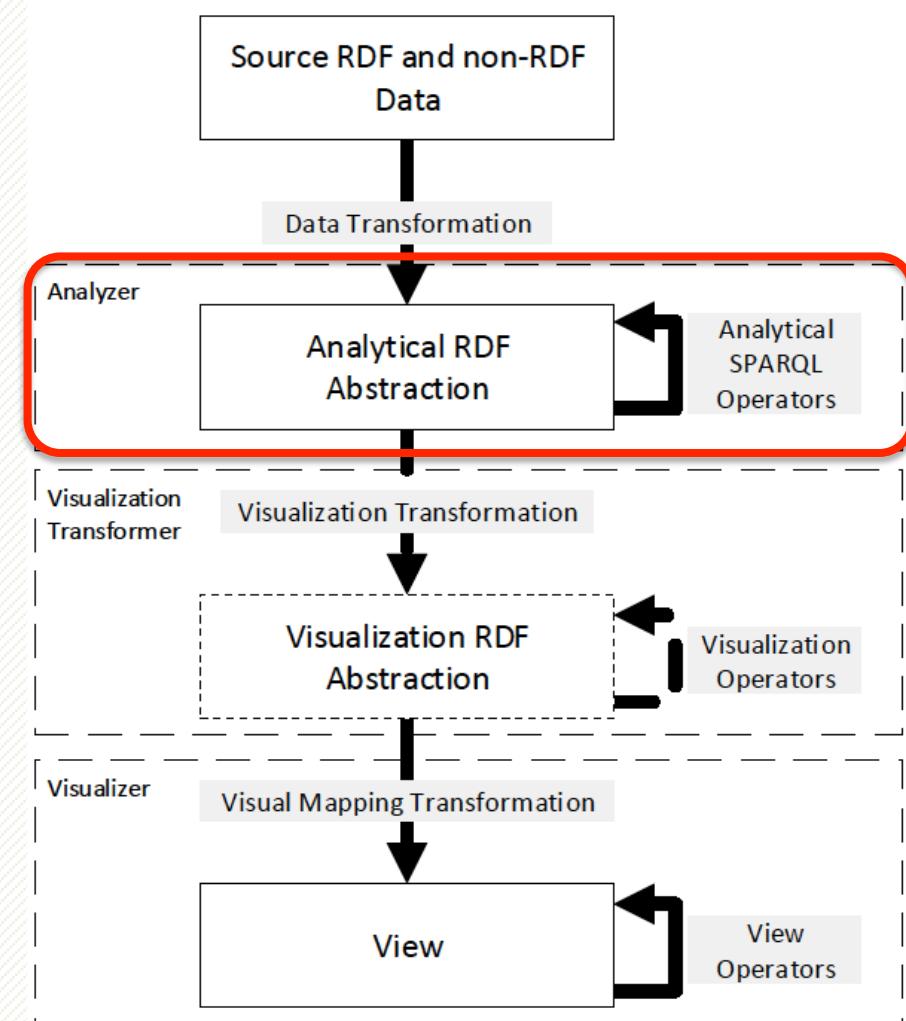
LDVM stages

- ❑ Source RDF and non-RDF data
 - Data transformation



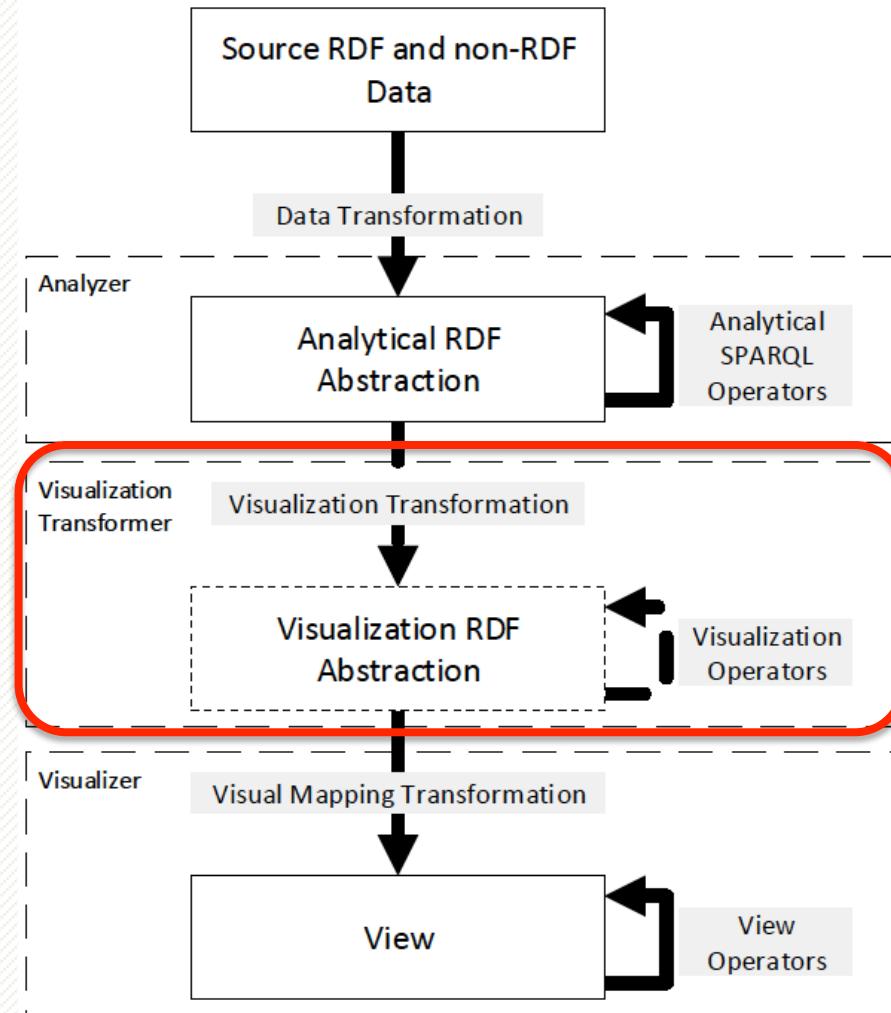
LDVM stages

- Analytical abstraction
 - Relevant data extraction using SPARQL operators
 - Analyzers
 - Reusability



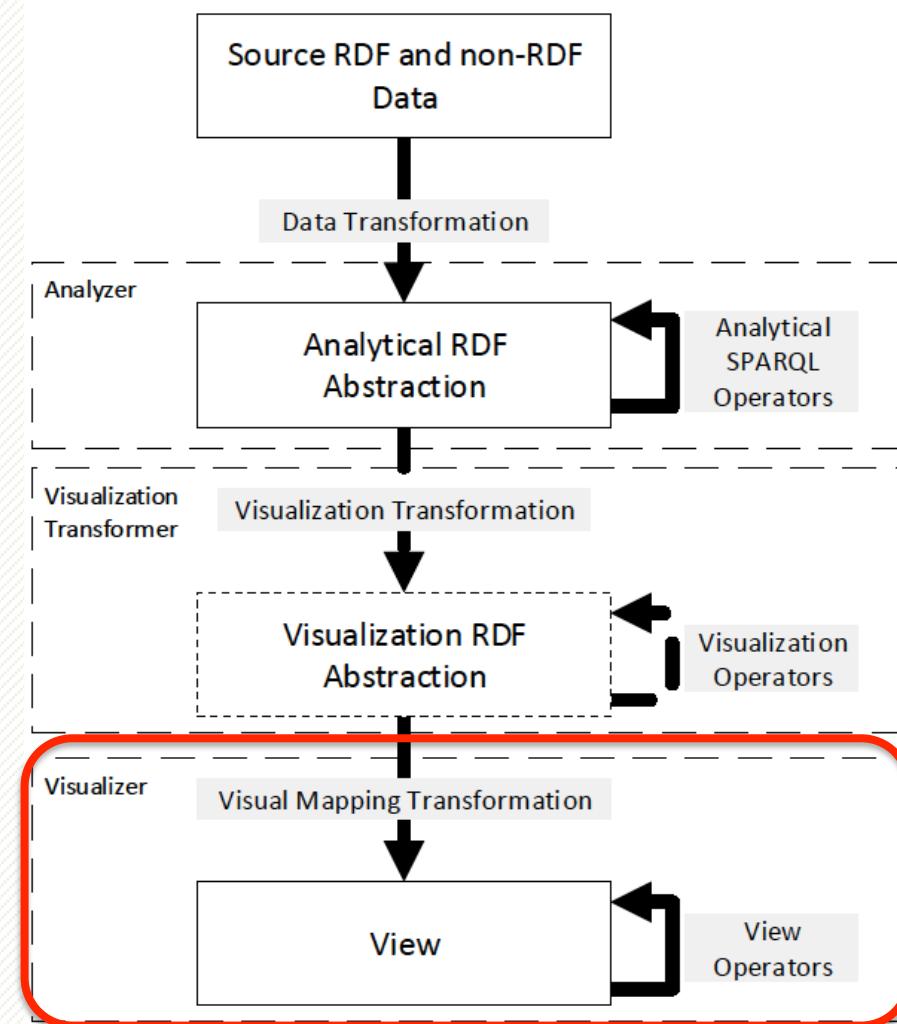
LDVM stages

- ❑ Visualization abstraction
 - Transformers
 - Reusing concepts
 - e.g. skos:broader



LDVM stages

- View
 - Visualizers
 - User configuration
(rotate, scale, zoom, etc.)
 - End-users



LDVM components compatibility

- ❑ Components compatibility
 - Input signature
 - SPARQL ASK
 - Output data sample

LDVM Demonstration

- ❑ Case: Inspections of COI.CZ
- ❑ Input signature of an analyzer:

```
# Q1 of A2
[] a s:CheckAction;
  s:location/s:location ?region;
  s:location/s:geo ?geo;
  s:object ?object;
  dcterms:date ?date ;
  s:result ?result.

?result a coicz:Sanction;
        s:result/gr:hasCurrencyValue [] .

?object gr:legalName [] .

?region a ec:LAURegion;
        ec:level 2 .

?geo s:latitude [];
     s:longitude [].

FILTER(datatype(?date) = xsd:date)
```

Compatibility Demonstration

```
# D of A2
<ca> a s:CheckAction;
      s:location <region> ;
      s:geo <geo>;
      s:title "title";
      s:description "description";
      dcterms:date "2014-02-16";
      rdf:value 2 .
<geo> s:latitude "50.088289";
      s:longitude "14.404446".
<region> a ec:LAURegion;
          ec:level 2 ;
          rdfs:label "label" ;
          ec:hasParentRegion <lau1>.
<lau1> rdfs:label "label" ;
          ec:hasParentRegion <nuts3> .
<nuts3> rdfs:label "label" ;
          ec:hasParentRegion <nuts2> .
<nuts2> rdfs:label "label" ;
          ec:hasParentRegion <nuts1> .
<nuts1> rdfs:label "label".
```

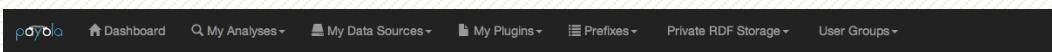
```
# Q of T1
[] a s:CheckAction;
  s:location ?region;
  s:title [] ;
  rdf:value [] .
?region a ec:LAURegion;
        ec:level 2 ;
        rdfs:label [] ;
        ec:hasParentRegion ?lau1 .
?lau1 rdfs:label [] ;
        ec:hasParentRegion ?nuts3 .
?nuts3 rdfs:label [] ;
        ec:hasParentRegion ?nuts2 .
?nuts2 rdfs:label [] ;
        ec:hasParentRegion ?nuts1 .
?nuts1 rdfs:label [] .
```

Compatibility demonstration II

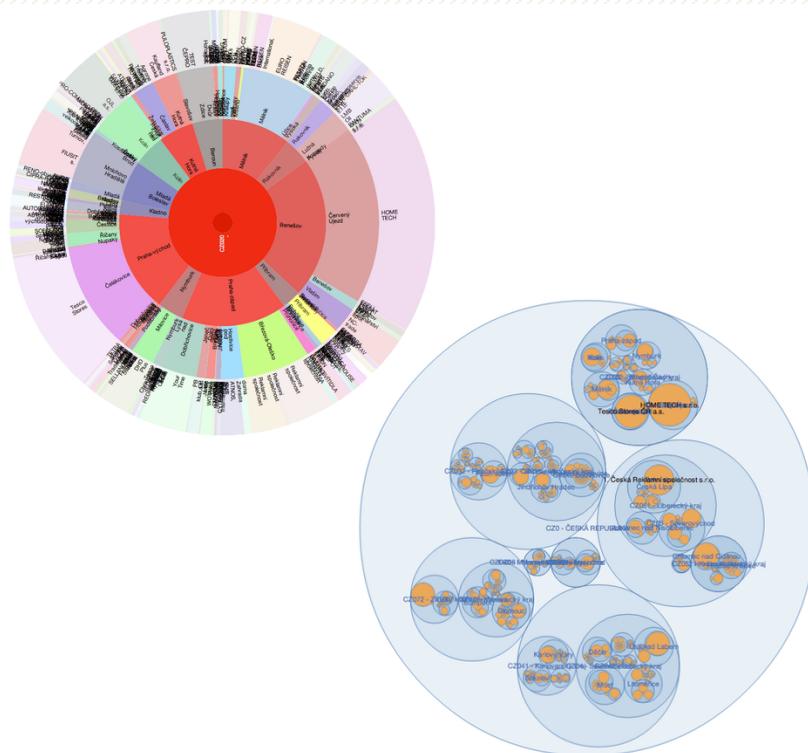
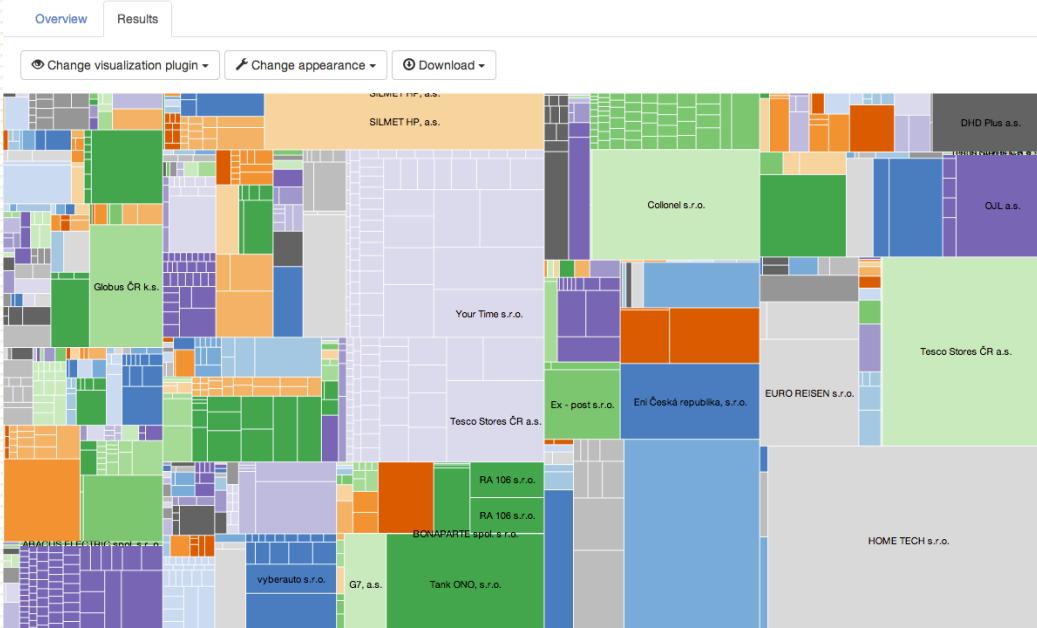
```
# D of T1                                     # Q of V1
<ca> a skos:Concept ;                      [] a skos:Concept ;
      skos:prefLabel "title" ;                skos:prefLabel [] ;
      rdf:value 100 ;                         rdf:value [] ;
      skos:broader <region> .                 skos:broader ?b .
<region> a skos:Concept ;                   ?b a skos:Concept ;
      skos:prefLabel "label" ;                  skos:prefLabel [] .
      skos:broader <lau1> .
<lau1> a skos:Concept ;                    <nuts3> a skos:Concept ;
      skos:prefLabel "label" ;                  skos:prefLabel [] ;
      skos:broader <nuts3> .
<nuts3> a skos:Concept ;                   <nuts2> a skos:Concept ;
      skos:prefLabel "label" ;                  skos:prefLabel "label" ;
      skos:broader <nuts2> .
<nuts2> a skos:Concept ;                   <nuts1> a skos:Concept ;
      skos:prefLabel "label" ;                  skos:prefLabel "label" ;
      skos:broader <nuts1> .
<nuts1> a skos:Concept ;
```

LDVM Demonstration

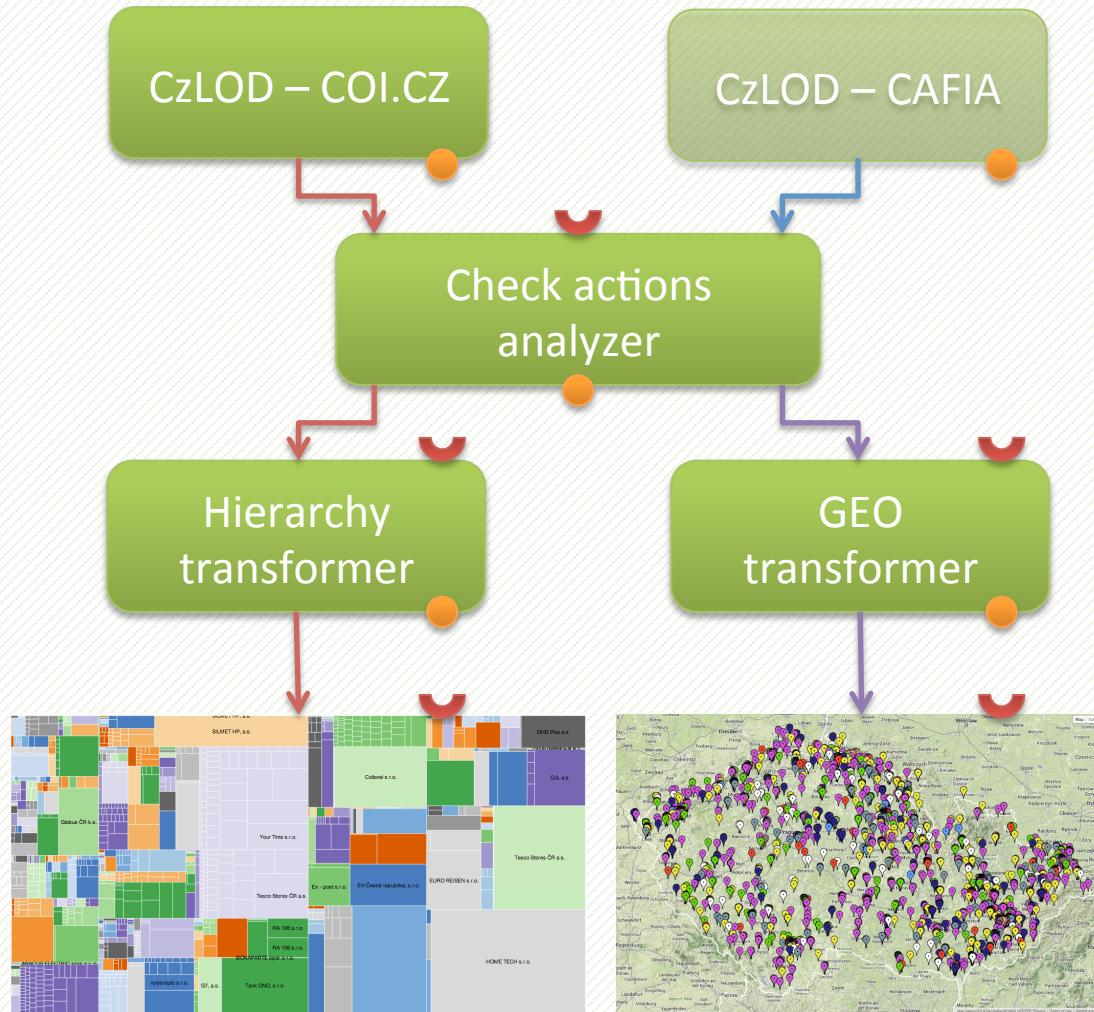
- Visualizers (e.g. d3js, jQuery flot, etc.)



Detail of analysis: COI.CZ inspections and sanctions by regions and san



Compatibility demonstration



Implementation: Payola

- ❑ Analyzer editor with analytical plugins
- ❑ Collaborative features
 - Inner analyzers
- ❑ Custom plugins
- ❑ One-click solution for non-expert users
 - Caching analyzer results



Links

- <http://vis.payola.cz/gmaps-ovm>
- <http://vis.payola.cz/coi-treemap>
- <https://github.com/payola/Payola>

```
<xs:complexType name="CategoryType">
<xs:sequence>
  <xs:element name="description" type="xs:string" />
  <xs:element name="category" type="CategoryType"
minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="books">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="book" type="BookType"
minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>
```

Thank you for your attention

Jakub Klímek

klimek@fit.cvut.cz

Faculty of Information Technology
Czech Technical University in Prague

Jiří Helmich

Martin Nečaský

<http://xrg.cz> | contact@xrg.cz

XML and Web Engineering Research Group

Faculty of Mathematics and Physics

Charles University, Prague

Czech Republic

