

# Last.fm Events

MI-SWE Project



ZS 2011/2012

Andrey Chervinka

[chervand@fit.cvut.cz](mailto:chervand@fit.cvut.cz)

# Purpose

- Get XML data from Last.fm API
- Parse XML data to suitable format
- Serialize data to RDF/N3 ontology
- Execute some queries on RDF
- Implementation in Ruby
- Sources: <https://github.com/chervand/LastEvents>

# Last.fm API

- <http://www.last.fm/api/>
- Allows to call methods that respond in REST style xml
- Authenticating with the API key
- Request example:

[http://ws.audioscrobbler.com/2.0/?method=artist.getSimilar&api\\_key=xxx...](http://ws.audioscrobbler.com/2.0/?method=artist.getSimilar&api_key=xxx...)

# geo.getEvents method

- <http://www.last.fm/api/show?service=270>
- Get all events in a specific location by country or city name
- Params: location, lat, long, distance, limit, page, api\_key
- Request example:

`http://ws.audioscrobbler.com/2.0/?  
method=geo.getevents&location=prague&api_key=xxx`

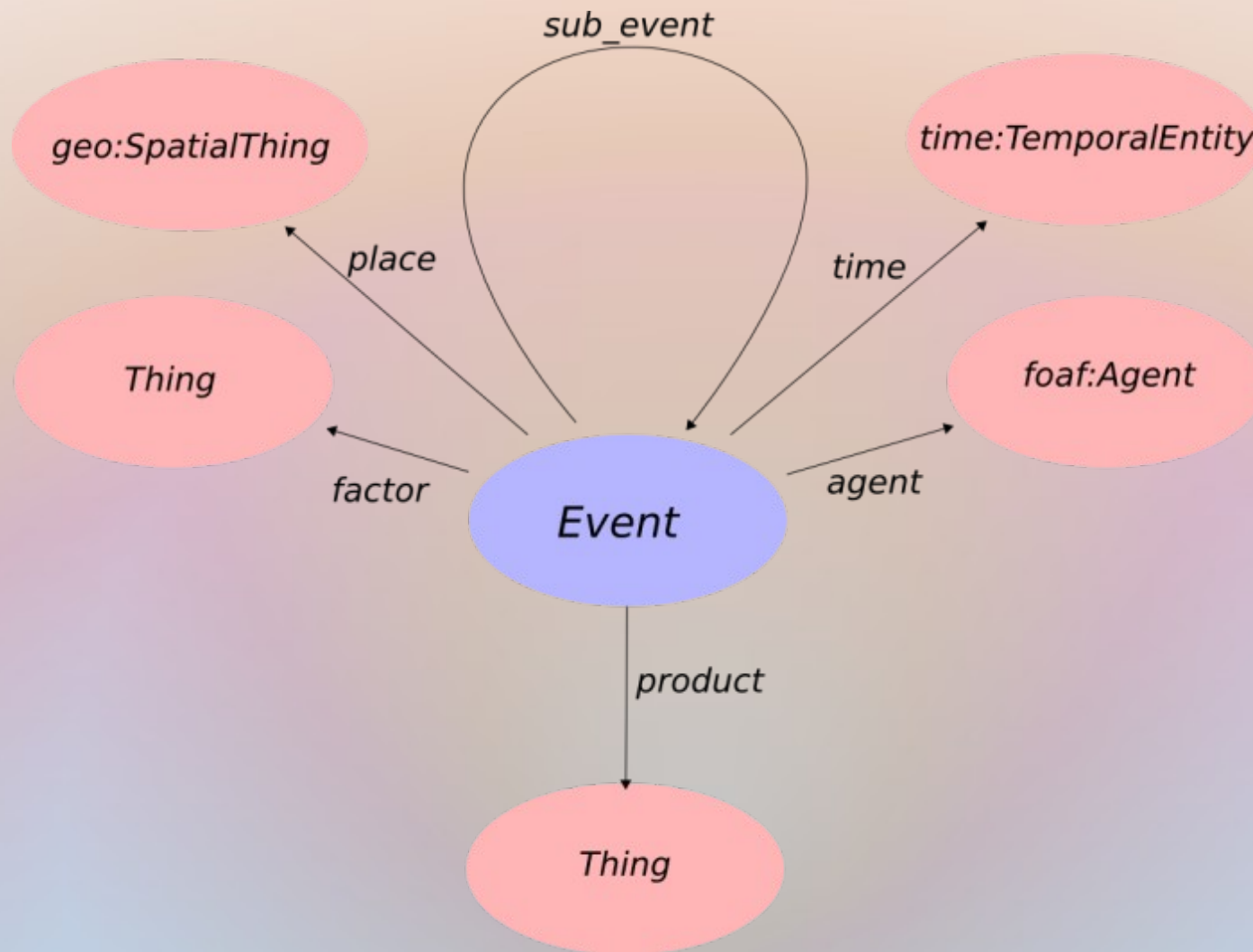
# geo.getEvents Sample Response

[Link](#)

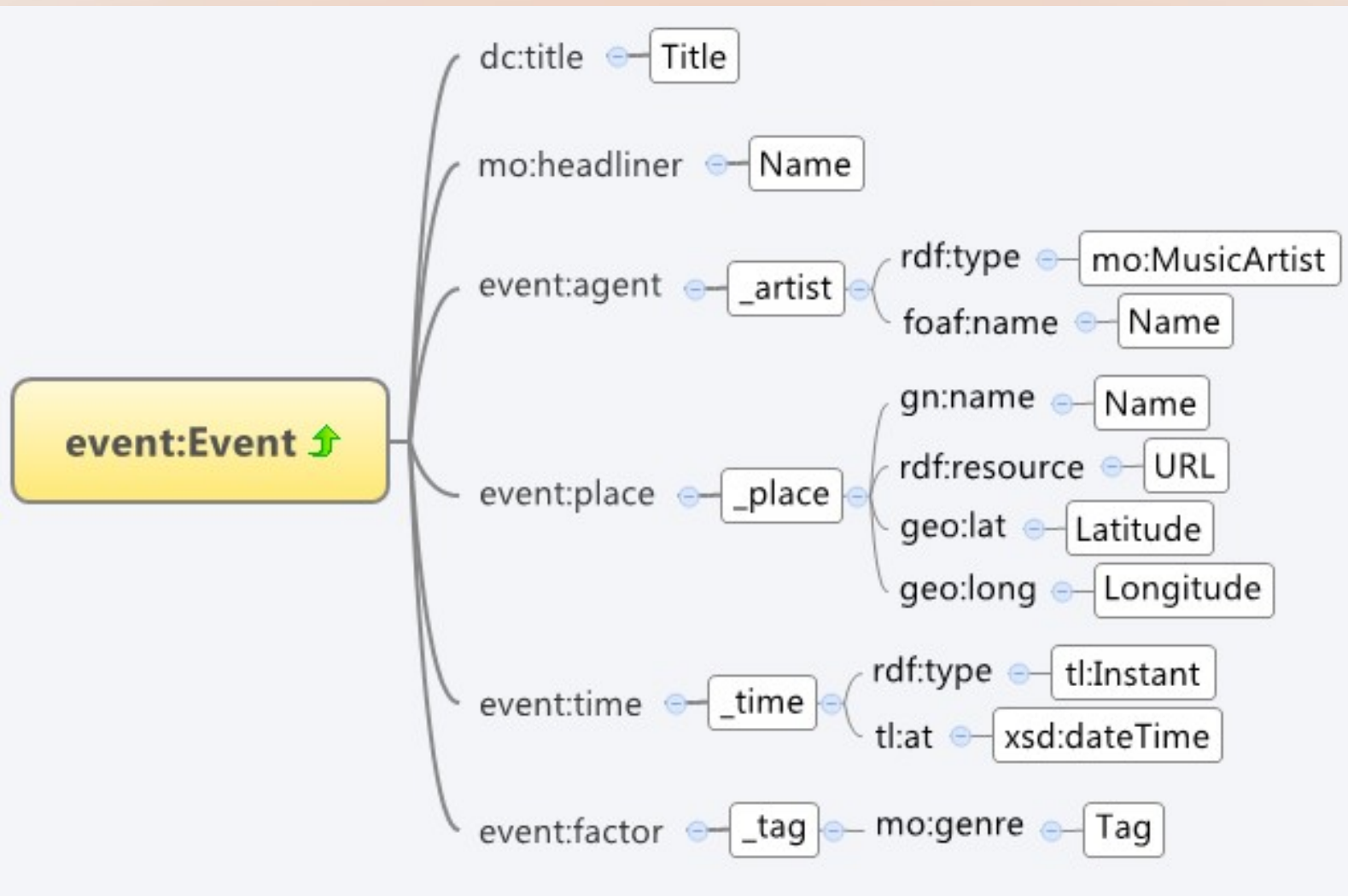
# Event Ontology

- <http://motools.sourceforge.net/event/event.html>
- Notion of event
- An event may have a location, a time, active agents, factors and products
- Agent - a person, a computer, ...
- Factor - a tool, an instrument, an abstract cause...
- Product - something produced during the event

# Event Ontology



# Final ontology





# RDF/N3 Example

more at: <https://github.com/chervand/LastEvents/tree/master/data/rdf>

```
<http://www.last.fm/event/1962280> a event:Event;
  dc:title "Metronomy";
  event:agent [ a mo:MusicArtist;
    foaf:name "Metronomy"],
    [ a mo:MusicArtist;
    foaf:name "Gross Magic"];
  event:factor [ mo:genre "new rave"],
    [ mo:genre "indie"],
    [ mo:genre "electronic"],
    [ mo:genre "experimental"],
    [ mo:genre "electro"],
    [ mo:genre "british"],
    [ mo:genre "indie rock"],
    [ mo:genre "indie pop"];
  event:place [ gn:name "Lucerna Music Bar";
    rdf:resource "http://www.musicbar.cz";
    geo:lat "50.081417";
    geo:long "14.425301"];
  event:time [ a tl:Instant;
    tl:at "2011-12-05T20:00:00"^^xsd:dateTime];
  mo:headliner "Metronomy" .
```

# Using the application

- Command line executing
- Run LastEvents ruby script from /bin folder

```
ruby LastEvents -h
Usage:  LastEvents [options]
    -l, --location          Location name (ex.'-l Prague')
    -t, --titles            Print titles
    -h, --help              Show this message
```

# Using the application

- Getting today's events in Prague
- XML data saved in data/xml/<location>-yyyy-mm-dd.xml file
- RDF/N3 data saved in data/rdf/<location>-yyyy-mm-dd.n3 file

```
ruby LastEvents -l Prague  
Requesting Last.FM for events in Prague  
Saving XML data  
XML data saved to Prague-2011-12-5.xml  
Serializing RDF/N3 data  
RDF/N3 data saved to Prague-2011-12-5.n3
```

# Using the application

- Performing a query to the RDF ontology
- Printing titles of the events using -t parameter

```
ruby LastEvents -l Prague -t
Requesting Last.FM for events in Prague
Saving XML data
XML data saved to Prague-2011-12-5.xml
Serializing RDF/N3 data
RDF/N3 data saved to Prague-2011-12-5.n3

TITLES
Free Mondays w/ Evelynne
Hudba Praha
Jana Lota
Kvelertak
Metronomy
Raindance (USA) ~ Hardcore Live
```

# Using the application

- Performing a query to the RDF ontology
- Printing titles of the events using -t parameter

```
ruby LastEvents -l Prague -t
Requesting Last.FM for events in Prague
Saving XML data
XML data saved to Prague-2011-12-5.xml
Serializing RDF/N3 data
RDF/N3 data saved to Prague-2011-12-5.n3

TITLES
Free Mondays w/ Evelynne
Hudba Praha
Jana Lota
Kvelertak
Metronomy
Raindance (USA) ~ Hardcore Live
```

# RDF::Query

- RDF.rb Class
- Usage example:

```
query = RDF::Query.new do
  pattern [:subject, RDF::DC.title, :object]
end
```

# Used gems

- Nokogiri XML Parser
- RDF.rb