

Comunica: a Modular SPARQL Query Engine for the Web

Ruben Taelman, Joachim Van Herwegen, Miel Vander Sande, Ruben Verborgh

Ghent University – imec – IDLab, Belgium

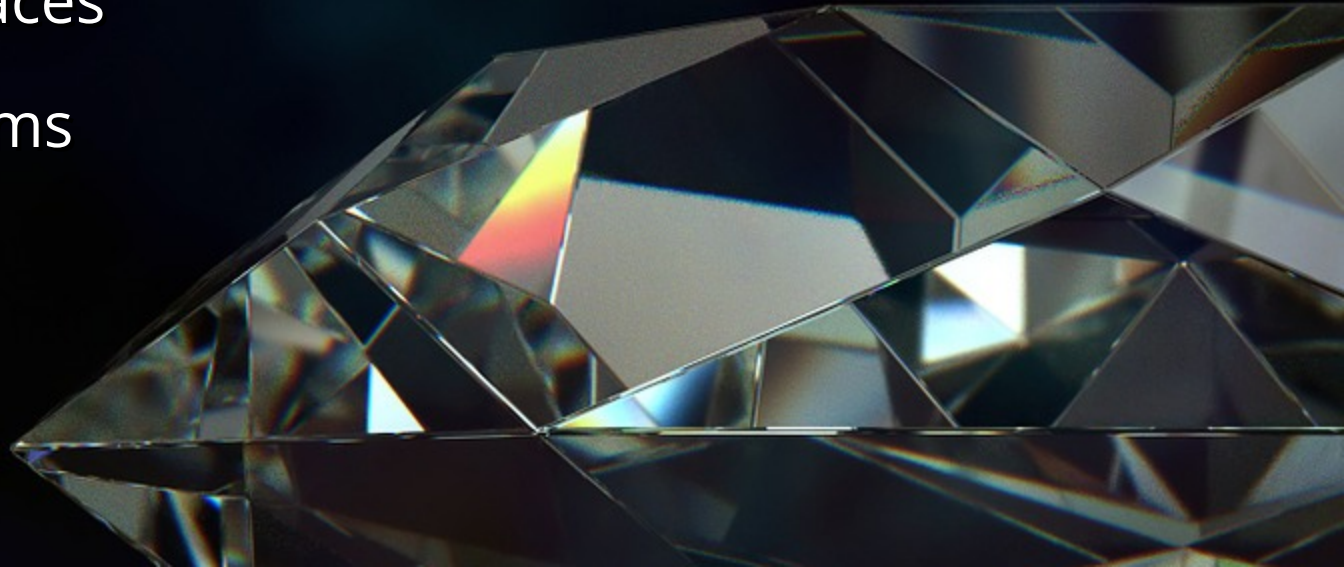
Querying Linked Data on the Web is a multifaceted story

Query languages and extensions

Publication interfaces

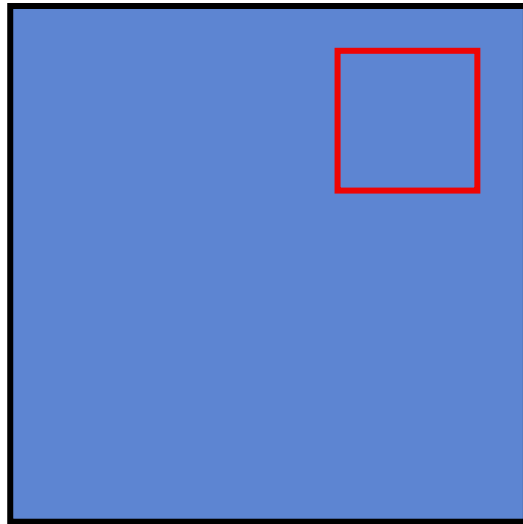
Querying algorithms

...

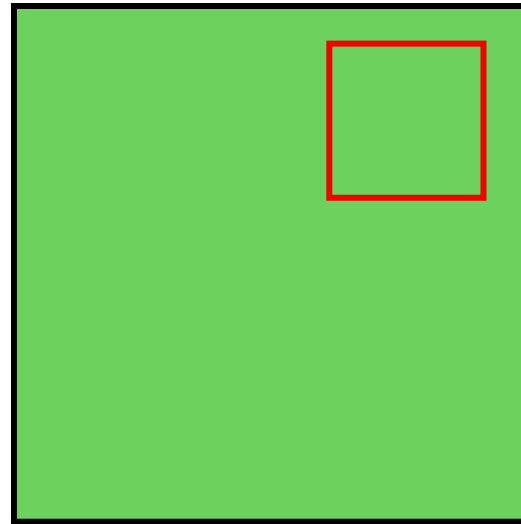


This variety of facets leads to *comparability* problems

Alternative algorithms are implemented in *different engines* → unfair!

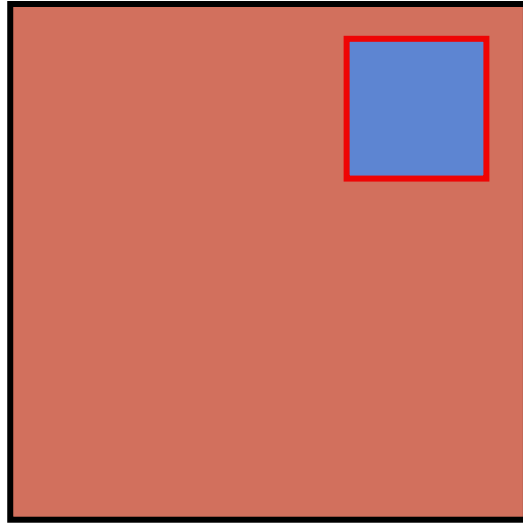


Engine A

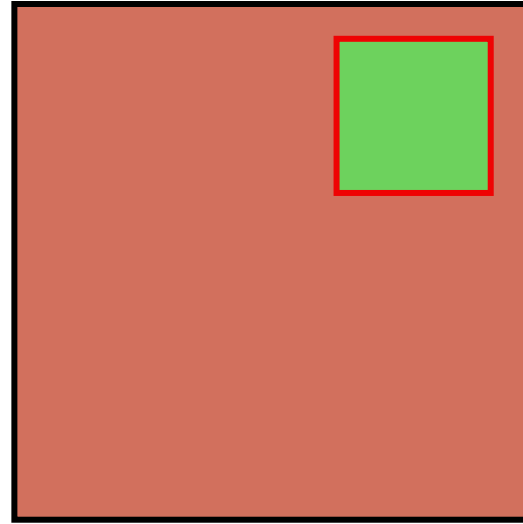


Engine B

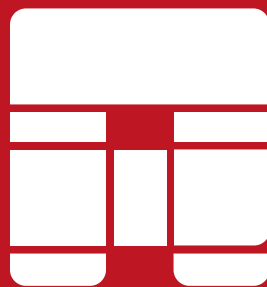
Need for a *flexible engine* to compare alternative algorithms



Algorithm A



Algorithm B



COMUNICA

Highly modular *meta* query engine

Flexible configuration system

Architecture

Implementation

Using Comunica

Architecture

Implementation

Using Comunica

Modularity using the *actor* model

Logic is separated into different *actors*.

Each actor *independently* performs a *specific task*.

JSON-LD parsing actor

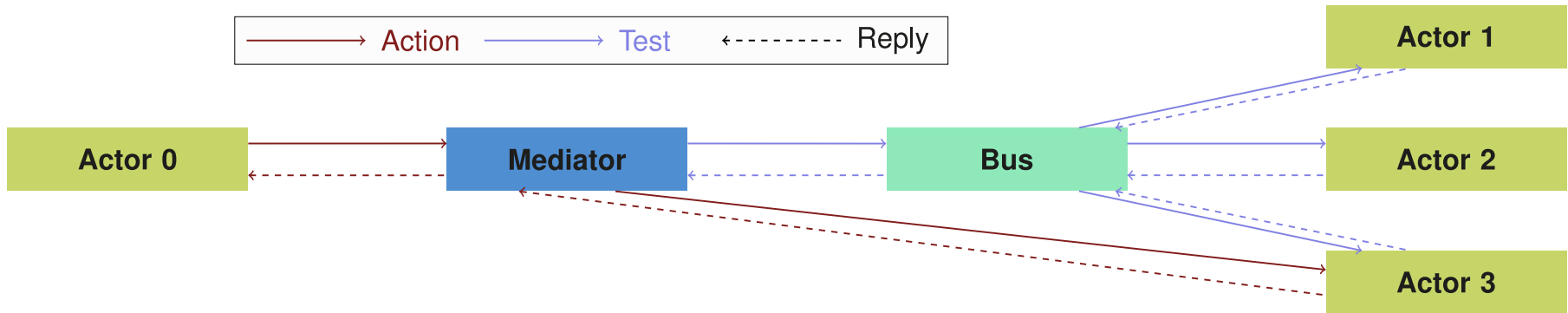
SPARQL union actor A

SPARQL union actor B

Actors can be *combined* for solving more *complex tasks*.

Actors subscribe to task-specific *buses*.

Mediators are responsible for selecting an actor to solve a certain task.



Example: Parse as fast as possible

Modules are wired together through semantic configuration files

Components.js: a semantic dependency injection framework.

Configuration files declare and parameterize actors, mediators and buses.

Architecture

Implementation

Using Comunica

Implementation details

Open-source under MIT license

Implemented in TypeScript/JavaScript

114 separate modules (*actors, buses, mediators*)

1619 unit tests with a coverage of 100%

Compatible with JavaScript API specification of RDFJS (W3C community group)

Current features

Nearly full SPARQL 1.1 support

GraphQL queries

Federated querying of heterogeneous sources:

- Linked Data document

- Triple Pattern Fragment

- SPARQL endpoint

- HDT files

- RDFJS Source

Handled with an internal SPARQL algebra

GraphQL is translated to SPARQL algebra

Any combination of sources

Federate over any set of sources
from your browser

Architecture

Implementation

Using Comunication

Using Comunica

in your browser:

<http://query.linkeddatafragments.org/>

in JavaScript applications:

Node.js: `npm install @comunica/actor-init-sparql`

Browser: <https://github.com/rdfjs/comunica-browser>

for development:

<https://github.com/comunica/comunica/>

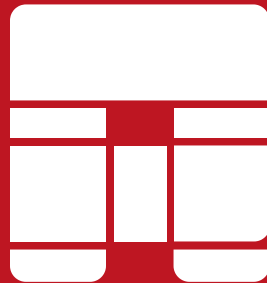
→ Documentation:

<https://comunica.readthedocs.io/en/latest/>

Fair.

Modular.

For everyone.



COMUNICA

<http://comunica.linkeddatafragments.org/>