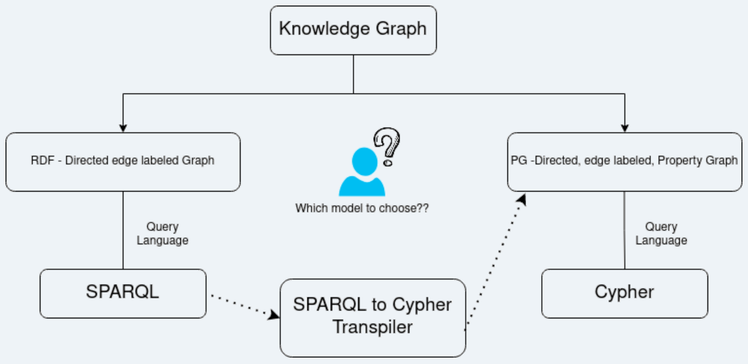
A SPARQL to Cypher Transpiler

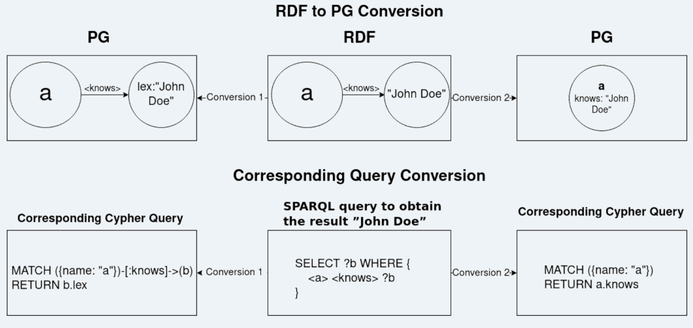
Nikunj Singhal Lakshya A Agrawal Dr. Raghava Mutharaju
KRaCR Lab, Indraprastha Institute of Information Technology Delhi

Motivation

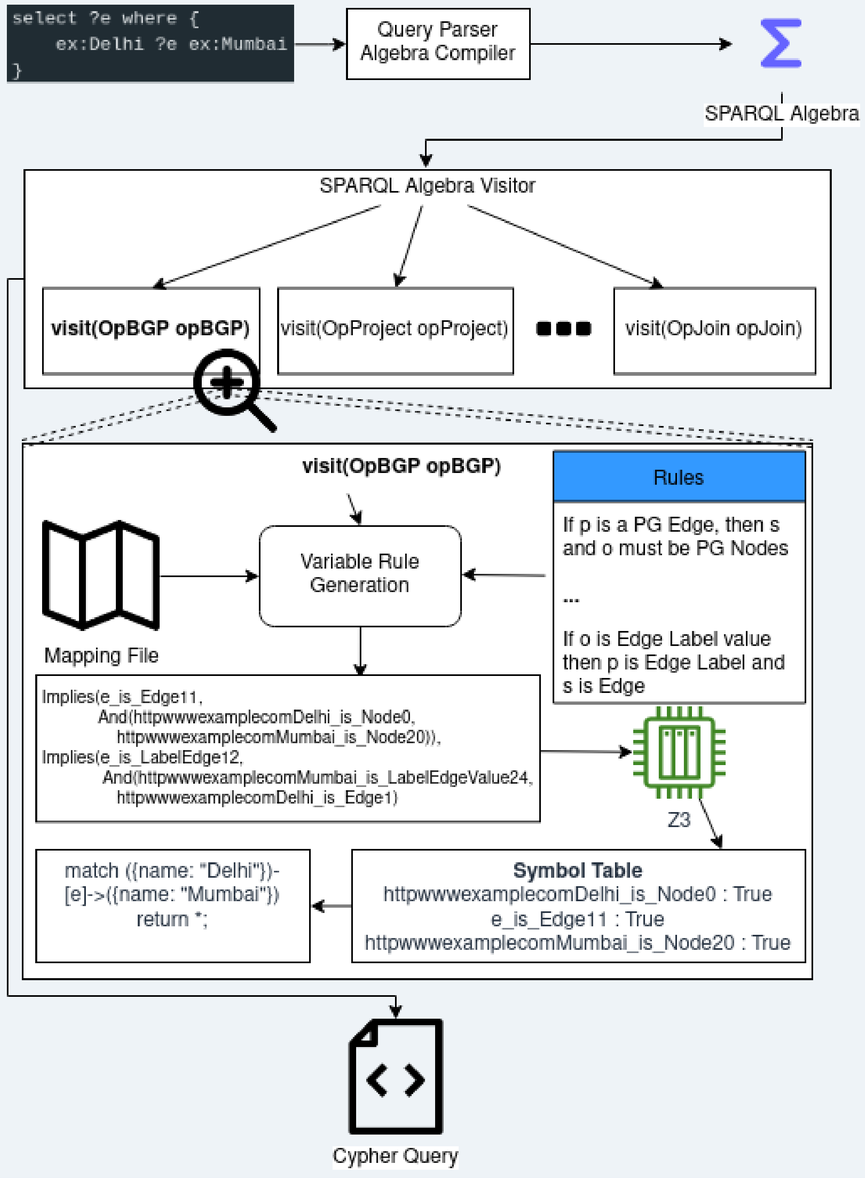
- Two most common Graph models:
  - Resource Description Framework(RDF)
    - W3C standardized graph model. Directed edge labelled graph. SPARQL as query language.
  - Property Graphs(PG):
    - Non standardised model - Directed, edge labeled graph with key-value properties as additional flexibility.
- Deciding between the 2 models complex due to considerations on data modeling and query language features.
- Approaches to interoperability have focused on converting RDF to PG directly, and new graph data models have been proposed that can be queried by both languages. Requires **very expensive data conversions,** especially for large datasets.
- We propose a **flexible SPARQL to Cypher Transpiler**, avoiding expensive data conversion and providing user flexibility by basing on a Mapping Language to allow different queries for different use cases and query sources.



Dependence of Query conversion on Schema conversion

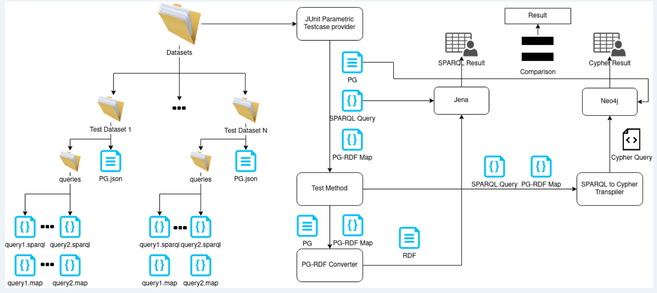


SPARQL to Cypher Transpiler



Testbench

Similar testing can be performed for originally RDF datasets imported to PG and queried using mapping language based SPARQL



Mapping Language

```
# Describe the conversion of nodes
nodes:
  # Convert nodes to IRI according to the following template
  convertTo: "http://www.example.com/@prop[name]"

  # Convert Node Labels to IRI or Literal
  labelValConvertTo: IRI
  labels:
    - http://www.example.com/type # The edge from node to label
    - http://www.example.com/node/@LabName # Template for Label IRI

  # Convert Node property values to Literal or IRI
  propertyValConvertTo: Literal
  properties:
    # Template for edge from node to property value
    - http://www.example.com/node/propName/@propKey
    - "@propVal" # Template for literal generation

# Describe the conversion of edges
edges:
  # Edges in PG are mapped to an edge in RDF according to the following template
  convertTo: edge
  # The edge template uses the edge label and one of its property values
  edgeIRI: "http://www.example.com/edge/@LabName/@prop[name]"

  # Convert Labels to Literal or IRI
  labelValConvertTo: Literal
  labels:
    - http://www.example.com/edge/type # Edge leading to the Literal
    - "@LabName" # Literal generation template

  # Convert properties to literal or IRI
  propertyValConvertTo: IRI
  properties:
    - http://www.example.com/edge/propName/@propKey # Edge template leading to the value
    - http://www.example.com/edge/propVal/@propVal # Template for value IRI
```

Future Work

- Extending Testbench to support query performance benchmarks
- Formal proof of correctness of query conversion
- Releasing the tool as Open Source public web application.

References

- Jesús Barrasa and Adam Cowley. [n.d.]. neosemantics (n10s): Neo4j RDF Semantics toolkit - Neo4j Labs. https://neo4j.com/labs/neosemantics/
- Olaf Hartig. 2014. Reconciliation of RDF* and Property Graphs. CoRR abs/1409.3288 (2014). arXiv:1409.3288
- Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia D'amato, Gerard De Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, and et al. 2021. Knowledge Graphs. Comput. Surveys 54, 4 (Jul 2021), 1–37. https://doi.org/10.1145/3447772
- Meenakshi Maindola and Raghava Mutharaju. 2020. PGBench: A Property Graph Benchmark for Knowledge Graphs. Master's thesis. Indraprastha Institute of Information Technology, Delhi.
- Michael Schmidt, Thomas Schallhorn, Georg Lausen, and Christoph Pinkel. 2009.
- SP2Bench: A SPARQL Performance Benchmark. Proceedings - International Conference on Data Engineering, 222 – 233. https://doi.org/10.1109/ICDE.2009.28