Towards a Visual SPARQL-DL Query Builder

Christian Gimenez[1] Germán Braun[1,3] Laura Cecchi[1]
Pablo Fillottrani[2,4]

[1]Universidad Nacional del Comahue
[2]Universidad Nacional del Sur
[3]Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
[4]Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC)

XXIV Congreso Argentino de Ciencias de la Computación
Octubre 2018

Contents

Outline

Outline

Objective

- We have a big ontology.

Objective

- We have a big ontology.
- How can we query it?.

Objective

- We have a big ontology.
- How can we query it?.
- But OWLlink/DIG is difficult to use!

Objective

- We have a big ontology.
- How can we query it?.
- But OWLlink/DIG is difficult to use!
- We want to query whatever we want!

Objective

- We have a big ontology.
- How can we query it?.
- But OWLlink/DIG is difficult to use!
- We want to query whatever we want!
  - Not only data, structure too!

Objective

- We have a big ontology.
- How can we query it?.
- But OWLlink/DIG is difficult to use!
- We want to query whatever we want!
  - Not only data, structure too!
- It would be awesome if it is a graphical language!

Objective

- We have a big ontology.
- How can we query it?.
- But OWLlink/DIG is difficult to use!
- We want to query whatever we want!
  - Not only data, structure too!
- It would be awesome if it is a graphical language!

Then, we want to:

- Make queries.

Objective

- We have a big ontology.
- How can we query it?.
- But OWLlink/DIG is difficult to use!
- We want to query whatever we want!
  - Not only data, structure too!
- It would be awesome if it is a graphical language!

Then, we want to:

- Make queries.
- With great expressivenes.

Objective

- We have a big ontology.
- How can we query it?.
- But OWLlink/DIG is difficult to use!
- We want to query whatever we want!
  - Not only data, structure too!
- It would be awesome if it is a graphical language!

Then, we want to:

- Make queries.
- With great expressivenes.
- Easy to read, write and understand.

Outline

What is *crowd*?

Web tool for graphical ontology modelling

crowd

What is *crowd*?

Web tool for graphical ontology modelling

Why *crowd*?

- Web tool.

crowd

What is *crowd*?

Web tool for graphical ontology modelling

Why *crowd*?

- Web tool.
- Adaptable and expandable.

crowd

What is *crowd*?

Web tool for graphical ontology modelling

Why *crowd*?

- Web tool.
- Adaptable and expandable.
- Supports UML as graphical language.

crowd

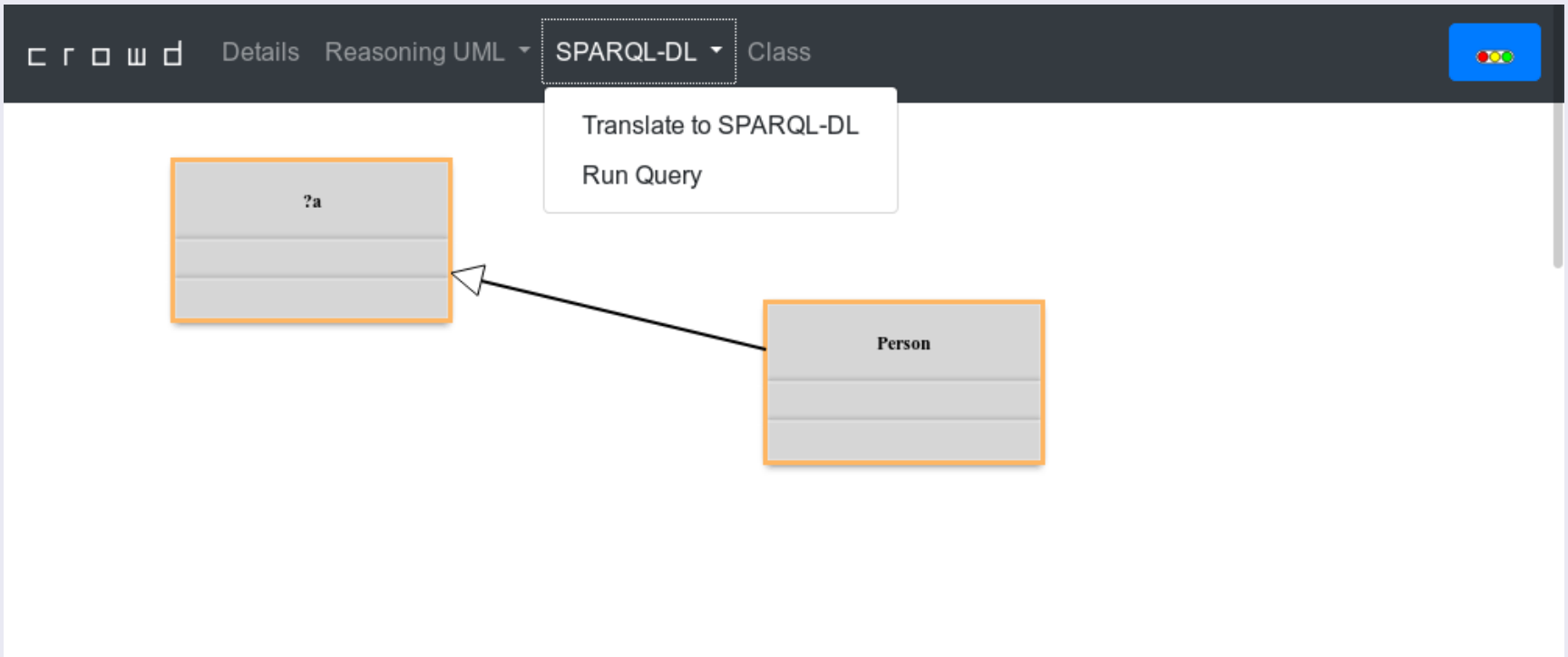
What is *crowd*?

Web tool for graphical ontology modelling

Why *crowd*?

- Web tool.
- Adaptable and expandable.
- Supports UML as graphical language.
- Berardi *et al.* encoding to OWL 2 and reasoning support.

A *crowd* screeshot



Outline

SPARQL-DL

What is SPARQL-DL?

A Query Language for OWL-DL ontologies significantly more expressive than existing DL QL.

SPARQL-DL

What is SPARQL-DL?

A Query Language for OWL-DL ontologies significantly more expressive than existing DL QL.

Why SPARQL-DL?

- Allows combined ABox, RBox and TBox queries.

SPARQL-DL

What is SPARQL-DL?

A Query Language for OWL-DL ontologies significantly more expressive than existing DL QL.

Why SPARQL-DL?

- Allows combined ABox, RBox and TBox queries.
- Aligned with SPARQL.

SPARQL-DL

What is SPARQL-DL?

A Query Language for OWL-DL ontologies significantly more expressive than existing DL QL.

Why SPARQL-DL?

- Allows combined ABox, RBox and TBox queries.
- Aligned with SPARQL.
- Can be used on top of OWL-DL reasoners.

SPARQL-DL

What is SPARQL-DL?

A Query Language for OWL-DL ontologies significantly more expressive than existing DL QL.

Why SPARQL-DL?

- Allows combined ABox, RBox and TBox queries.
- Aligned with SPARQL.
- Can be used on top of OWL-DL reasoners.

Example

```
SELECT * WHERE {Class(?x), Class(?y),
    DirectSubclassOf(?y, ?x)}
```

Outline

Outline

UML-like VQL

UML-like VQL

Let's define a UML-like Visual Query Language.

UML-like VQL

UML-like VQL

Let's define a UML-like Visual Query Language.

- *crowd* has UML support.

UML-like VQL

UML-like VQL

Let's define a UML-like Visual Query Language.

- *crowd* has UML support.
- UML is well-known.

UML-like VQL

Let's define a UML-like Visual Query Language.

- *crowd* has UML support.
- UML is well-known.
- We allow to use variables as classifier names.
  For example: `?person`

UML-like VQL

UML-like VQL

Let's define a UML-like Visual Query Language.

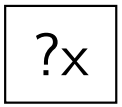
- *crowd* has UML support.
- UML is well-known.
- We allow to use variables as classifier names.
  For example: `?person`

UML	DL	OWL 2
TBox		
Classes Associations Generalizations	Concepts Roles Inclusion	Classes Object Properties `SubClassOf`
ABox		
Objects	Instances	`Individuals`/Instances

Outline

SPARQL-DL Encoding

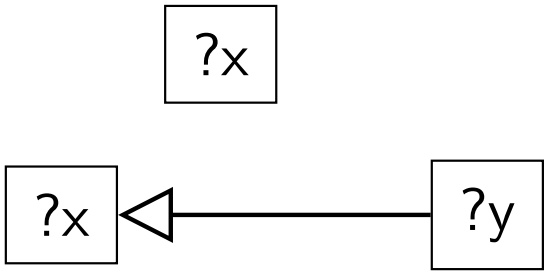
Now we need to encode the UML-like primitives into SPARQL-DL.




```
SELECT * WHERE {Class(?x)}
```

SPARQL-DL Encoding

Now we need to encode the UML-like primitives into SPARQL-DL.

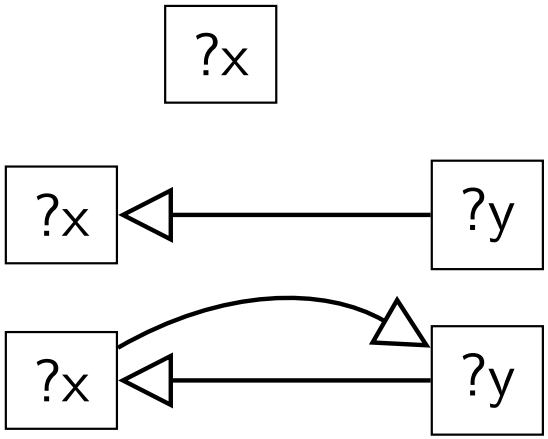



```
SELECT * WHERE {Class(?x)}
```

```
SELECT * WHERE {Class(?x), Class(?y),
  DirectSubclassOf(?y, ?x)}
```

SPARQL-DL Encoding

Now we need to encode the UML-like primitives into SPARQL-DL.



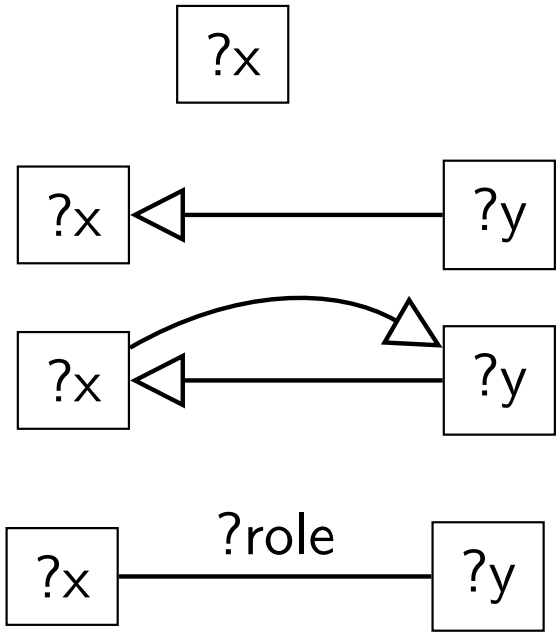

```
SELECT * WHERE {Class(?x)}
```

```
SELECT * WHERE {Class(?x), Class(?y),
  DirectSubclassOf(?y, ?x)}
```

```
SELECT * WHERE {Class(?x), Class(?y),
  EquivalentClass(?x, ?y)}
```

SPARQL-DL Encoding

Now we need to encode the UML-like primitives into SPARQL-DL.




```
SELECT * WHERE {Class(?x)}
```

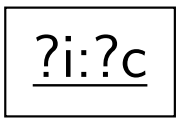
```
SELECT * WHERE {Class(?x), Class(?y),
  DirectSubclassOf(?y, ?x)}
```

```
SELECT * WHERE {Class(?x), Class(?y),
  EquivalentClass(?x, ?y)}
```

```
SELECT * WHERE {Class(?x), Class(?y),
  Domain(?role, ?x), Range(?role, ?y)}
```

SPARQL-DL Encoding (instances)

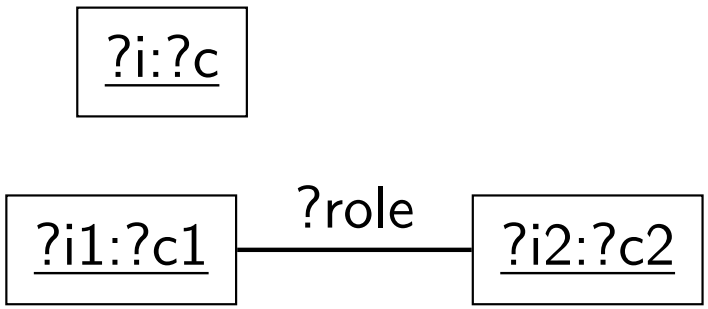
Now we need to encode the UML-like primitives into SPARQL-DL.




```
SELECT * WHERE {Type(?i, ?c)}
```

SPARQL-DL Encoding (instances)

Now we need to encode the UML-like primitives into SPARQL-DL.

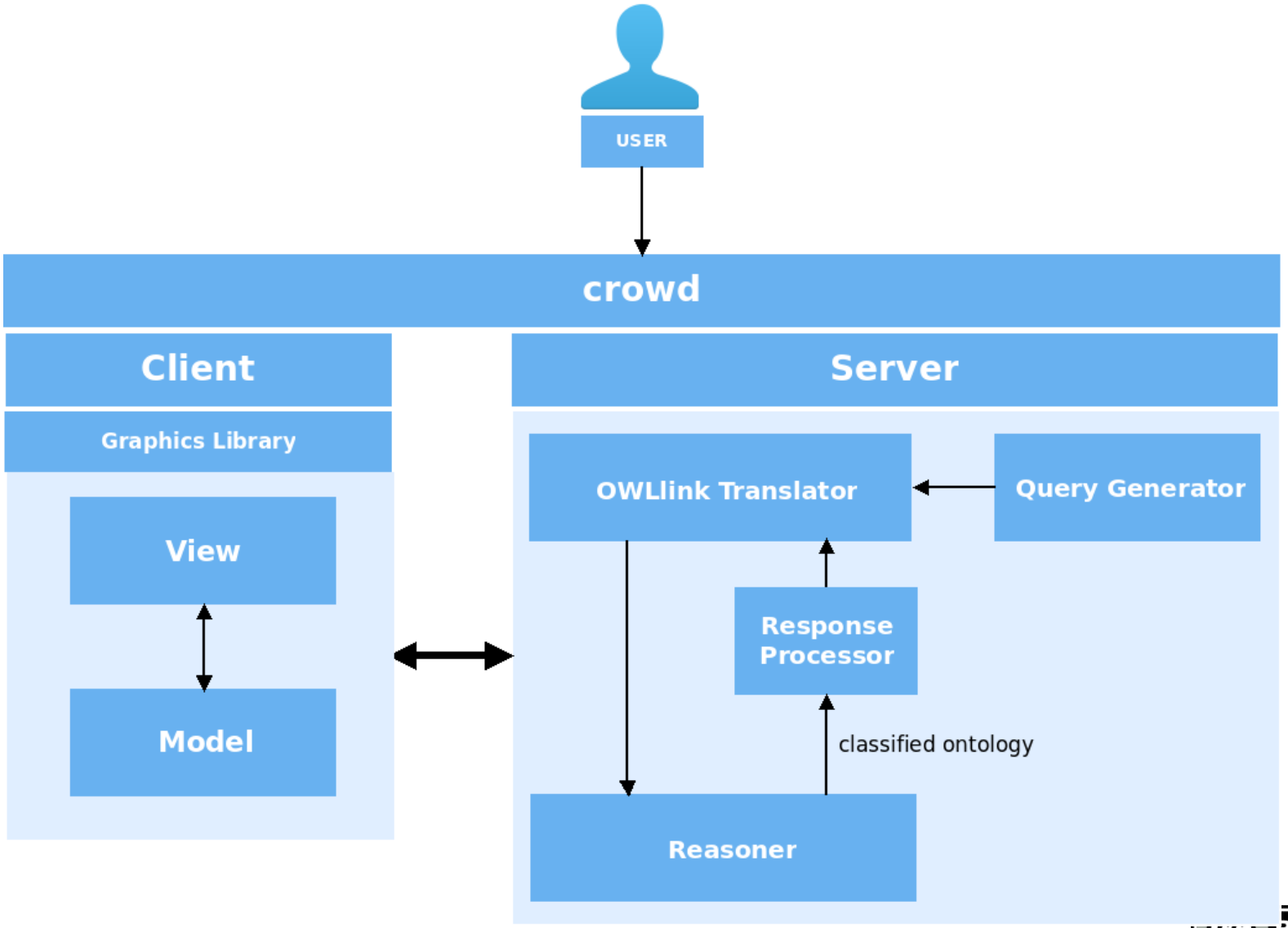



```
SELECT * WHERE {Type(?i, ?c)}
```

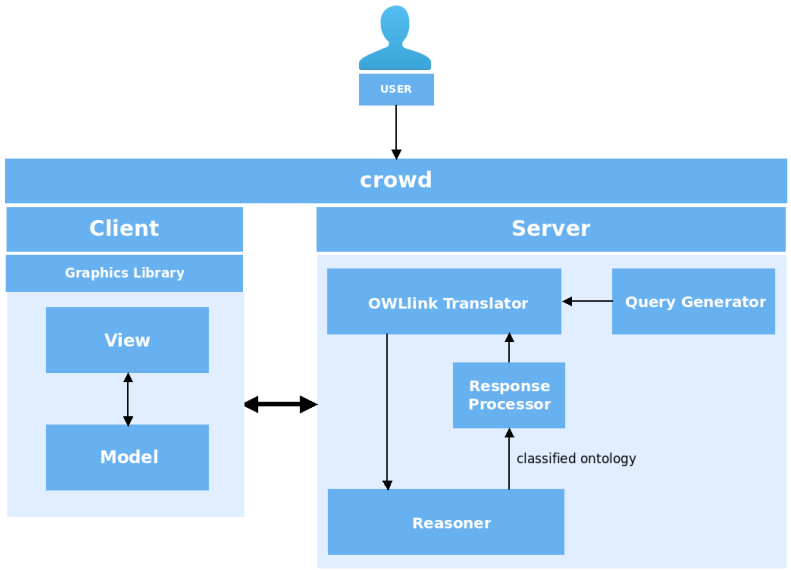
```
SELECT * WHERE {
  PropertyValue(?c1, ?role, ?c2),
  Type(?i1, ?c1), Type(?i2, ?c2)}
```

Outline

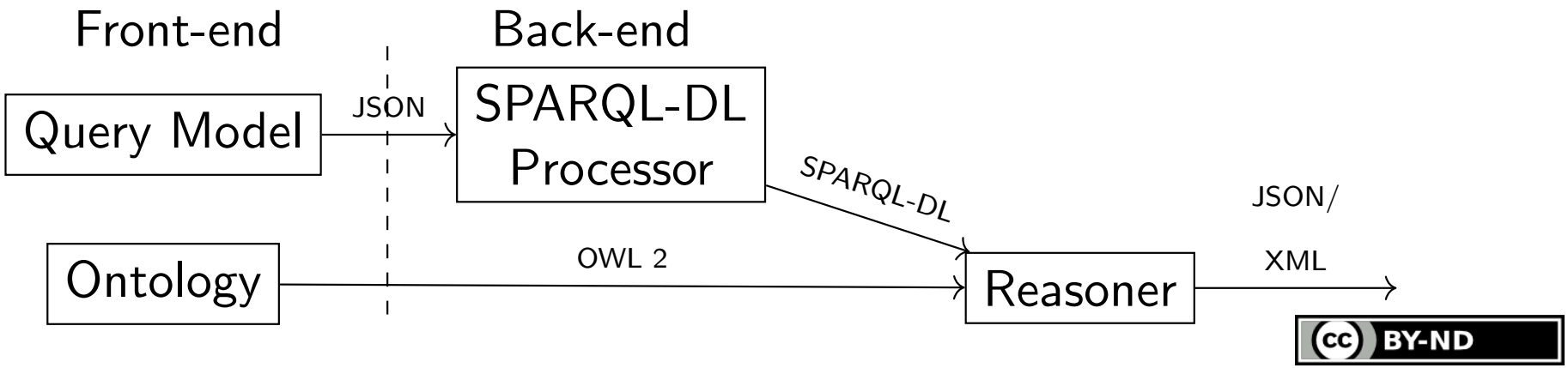
crowd Architecture



crowd Architecture



Back-end steps for processing the query modelled by the user.

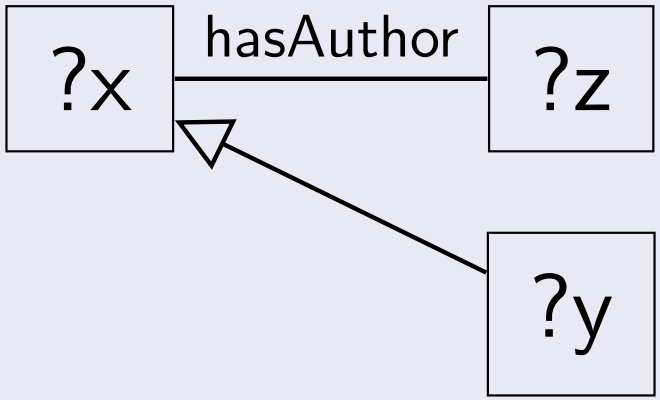


Outline

Example

UML-like Query

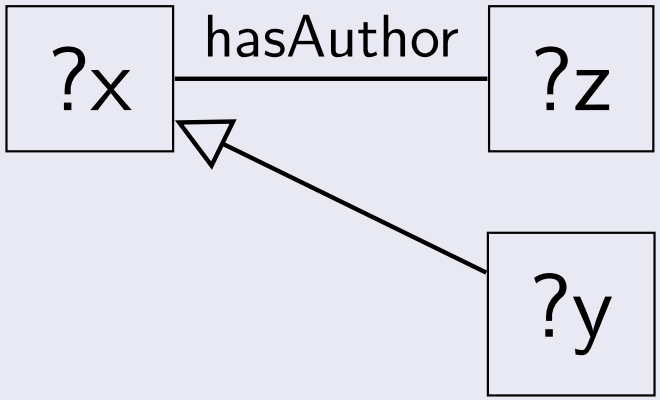
A query expressed in UML-like class diagram model.



Example

UML-like Query

A query expressed in UML-like class diagram model.



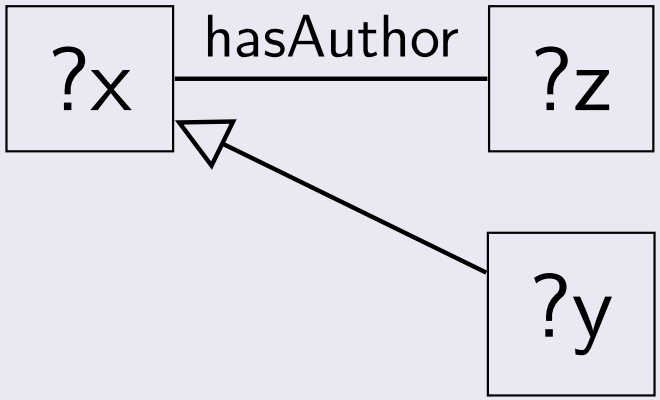
SPARQL-DL Encoding

```
SELECT ?x,?z,?y WHERE
{Class(?x), Class(?z), Class(?y),
```

Example

UML-like Query

A query expressed in UML-like class diagram model.



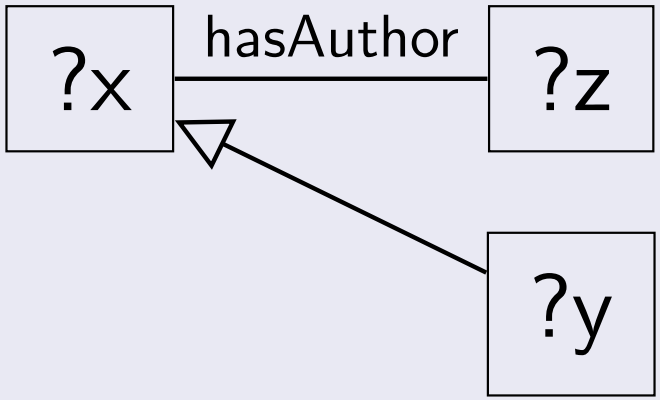
SPARQL-DL Encoding

```
SELECT ?x,?z,?y WHERE
{Class(?x), Class(?z), Class(?y),
  DirectSubclassOf(?y,?x),
```

Example

UML-like Query

A query expressed in UML-like class diagram model.



SPARQL-DL Encoding

```
SELECT ?x,?z,?y WHERE
{Class(?x), Class(?z), Class(?y),
  DirectSubclassOf(?y,?x),
  Domain(?x,:hasAuthor), Range(?z,:hasAuthor)
}
```

Thank you!

Questions?

License

Unless where otherwise stated:

CC-By-ND