

Description

This directory contains the ontology version descriptions for the domain Auto. Following are the descriptions of the files that are contained.

Ontology Version	Ontology File	Data File
1	ExampleO21.owl	Datao21.owl
2	ExampleO22.owl	Datao22.owl
3	ExampleO23.owl	Datao23.owl
4	ExampleO24.owl	Datao24.owl

Backward Compatibility

Version 2 is backward compatible with Version 1.

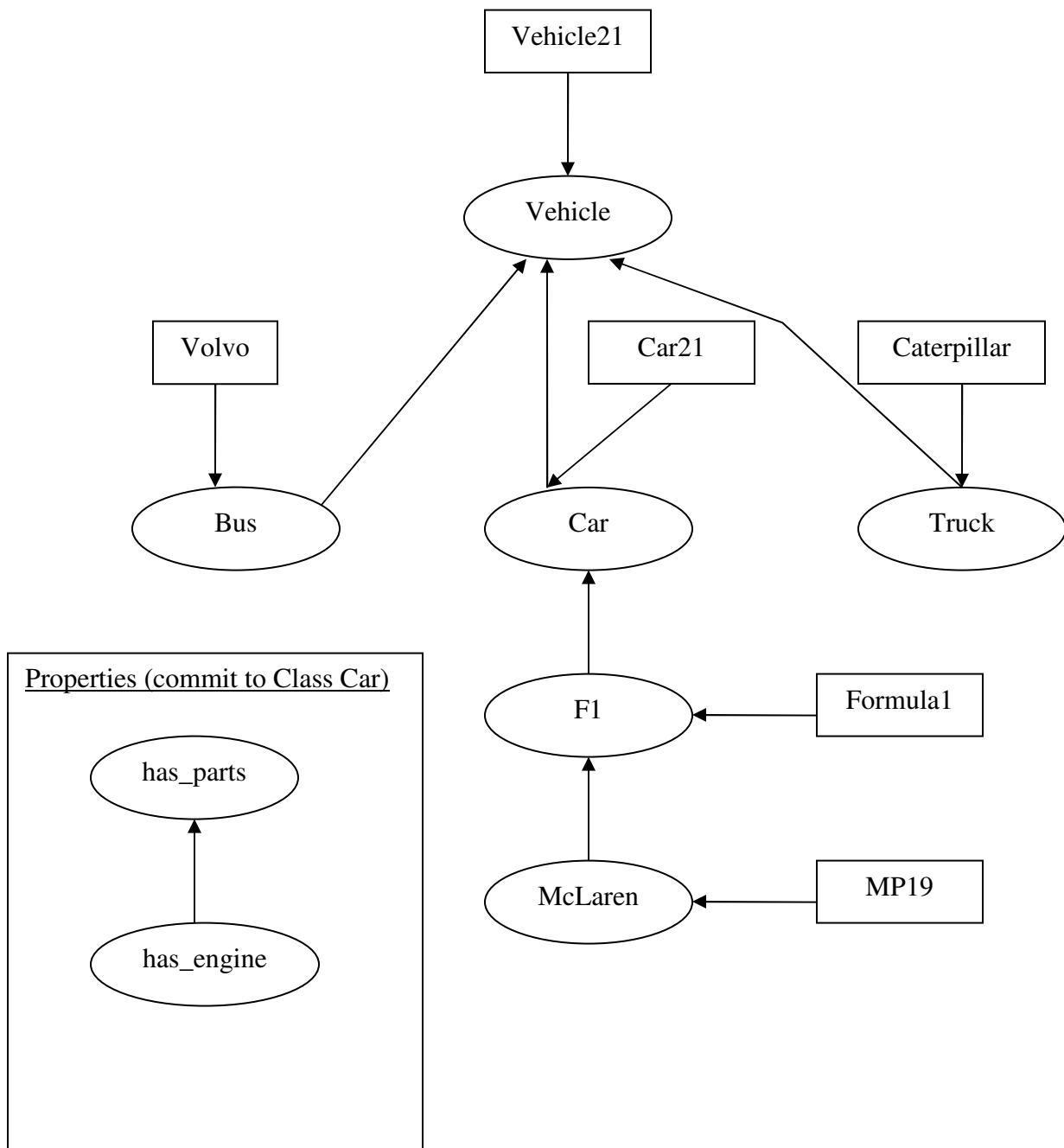
Version 3 is backward compatible with Version 2.

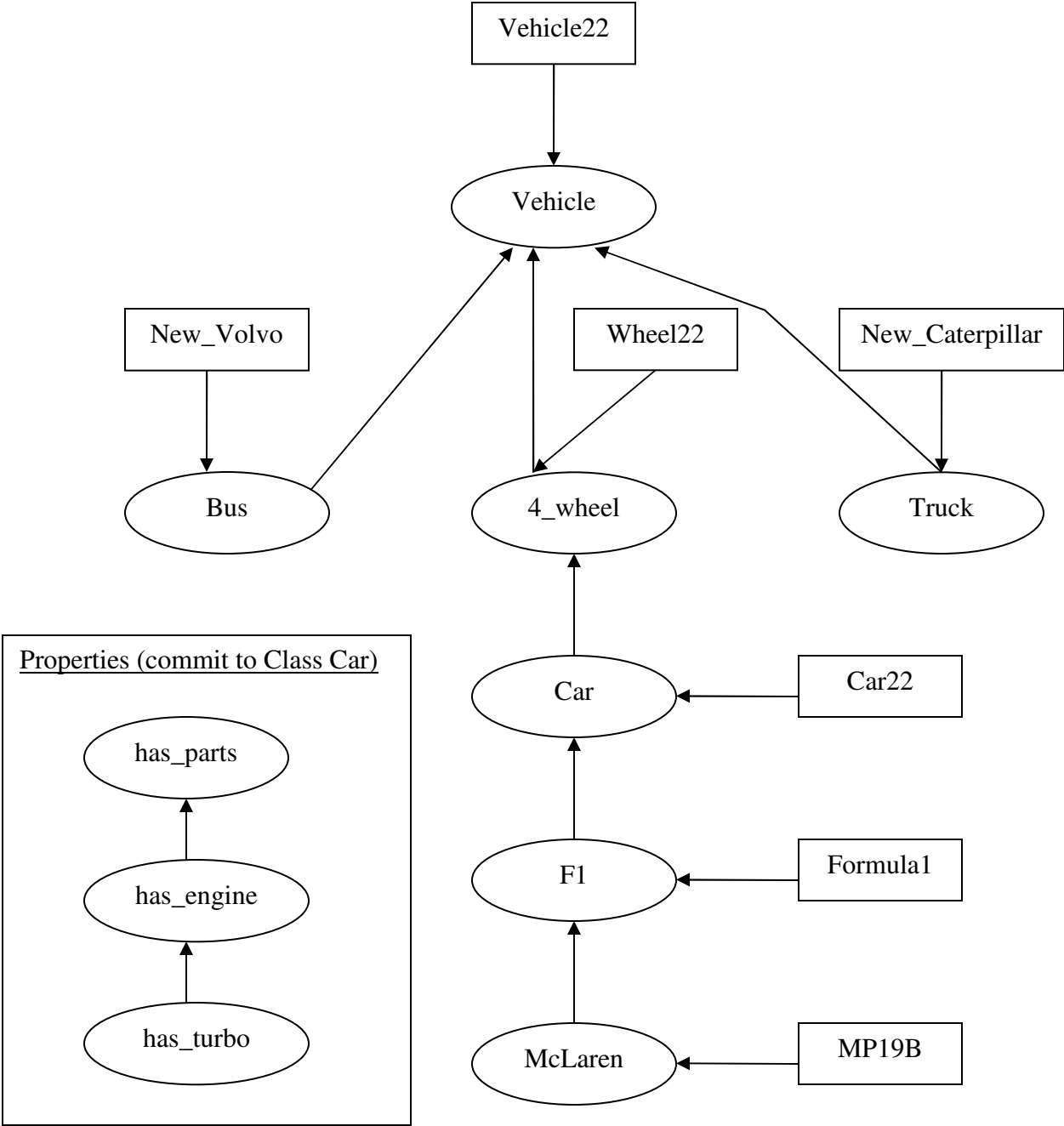
Version 4 is backward compatible with Version 1.

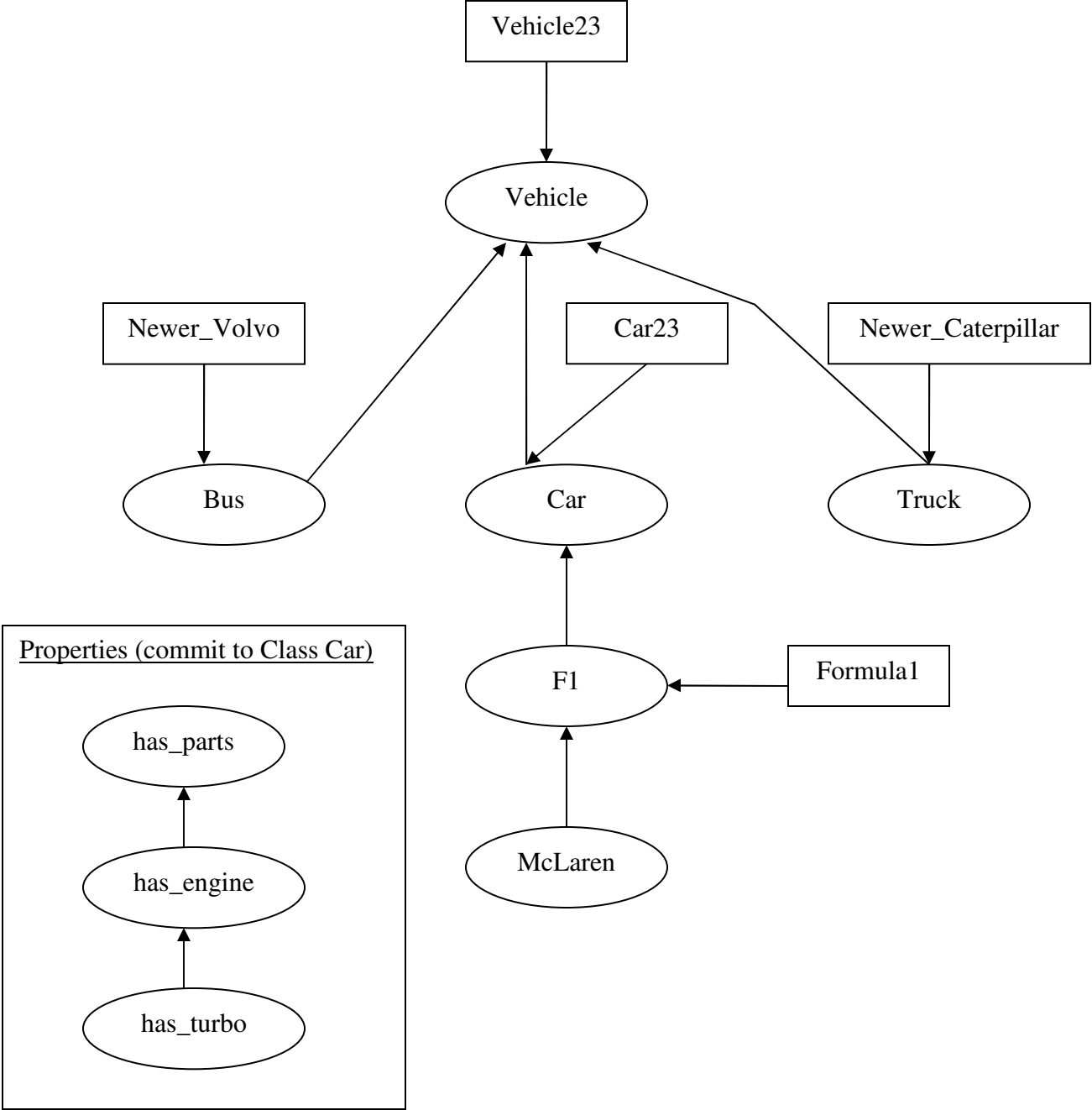
This document further contains the Diagrammatic representation of the Ontology version and some sample query results obtained through the designed Type Graph model. Sample #2 is a query on the property relationships and hence require a slight modification in the query. Hence I have included a step by step querying process for that one.

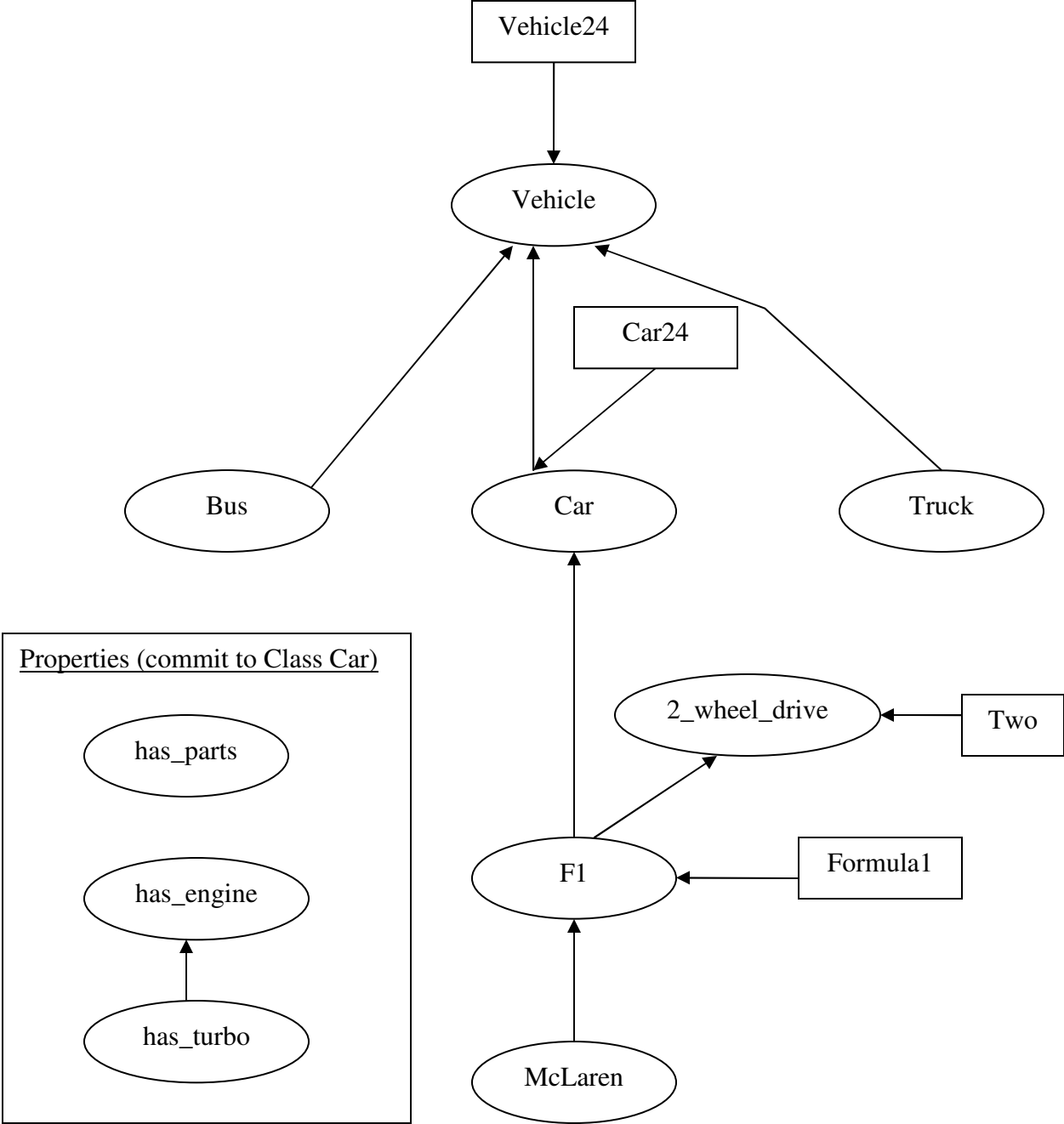
Diagrammatic Representation of AutoOntology (Directory – Auto)

Ontology Version-1 O_{V1}









Sample Query Results for Auto Ontology with different Perspectives

1) Querying for “Car” with Perspective “O_{v2}”

Results:

(x, file:///c:/dione/auto/datao22.owl#Formula1)
(x, file:///c:/dione/auto/datao22.owl#Car22)
(x, file:///c:/dione/auto/datao22.owl#MP19B)
(x, file:///c:/dione/auto/datao21.owl#MP19)
(x, file:///c:/dione/auto/datao21.owl#Car21)
(x, file:///c:/dione/auto/datao21.owl#Formula1)
6 answers

2) Querying for “has_parts”(property) with Perspective “O_{v2}”

In the GUI, where it says “type the url here”, give

<file:///c:/dione/auto/ExampleO22.owl> (or wherever you have stored the owl file)

On pressing load you should see all the documents that are loaded automatically, for example the owl files for the ontology that are backward compatible with ExampleO12
You should see something like this in the loaded documents window,

- file:///c:/dione/auto/ExampleO12.owl
- file:///c:/dione/auto/ExampleO11.owl

Now load all the data files linked to the owl files, for example

file:///c:/dione/auto/datao12.owl

file:///c:/dione/auto/datao11.owl

Note: this is done manually for all the data files. System will not automatically load data files for backward compatible files as it did in the case of the ontology files above

In the section Input you RDQL query paste the following query,

```
SELECT ?x ?y WHERE (?x,<file:///c:/dione/auto/ExampleO22.owl#has_parts>,<?y>
USING rdf FOR <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
rdfs FOR <http://www.w3.org/2000/01/rdf-schema#>
owl FOR <http://www.w3.org/2002/07/owl>
O1 for <file:///c:/dione/auto/ExampleO22.owl#>
```

Press “Query using VPI”. It will give the following results,

(x, file:///c:/dione/auto/datao22.owl#Car22) (y,
valueCar22has_turbo:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao22.owl#Car22) (y,
valueCar22has_parts:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao22.owl#Car22) (y,
valueCar22has_engine:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao21.owl#Car21) (y,
valueCar21has_parts:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao21.owl#Car21) (y,
valueCar21has_engine:http://www.w3.org/2001/XMLSchema#string)

5 answers

Press “Query without using VPI”. It will give the following results,

(x, file:///c:/dione/auto/datao22.owl#Car22) (y,
valueCar22has_parts:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao22.owl#Car22) (y,
valueCar22has_engine:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao22.owl#Car22) (y,
valueCar22has_turbo:http://www.w3.org/2001/XMLSchema#string)

3 answers

3) Querying for “Car” with Perspective “O_{V3}”

Results:

(x, file:///c:/dione/auto/datao21.owl#Car21)
(x, file:///c:/dione/auto/datao23.owl#Car23)
(x, file:///c:/dione/auto/datao22.owl#Car22)

3 answers

4) Querying for “F1” with Perspective “O_{V3}”

Results:

(x, file:///c:/dione/auto/datao22.owl#MP19B)
(x, file:///c:/dione/auto/datao21.owl#Formula1)
(x, file:///c:/dione/auto/datao21.owl#MP19)
(x, file:///c:/dione/auto/datao23.owl#Formula1)
(x, file:///c:/dione/auto/datao22.owl#Formula1)
5 answers

5) Querying for “Car” with Perspective “ O_{V4} ”

Results:

(x, file:///c:/dione/auto/datao24.owl#Formula1)
(x, file:///c:/dione/auto/datao21.owl#Car21)
(x, file:///c:/dione/auto/datao21.owl#MP19)
(x, file:///c:/dione/auto/datao21.owl#Formula1)
(x, file:///c:/dione/auto/datao24.owl#Car24)
5 answers

6) Querying for “Two_wheel_drive” with Perspective “ O_{V4} ”

Results:

(x, file:///c:/dione/auto/datao24.owl#Formula1)
(x, file:///c:/dione/auto/datao24.owl#TWO)
(x, file:///c:/dione/auto/datao21.owl#MP19)
(x, file:///c:/dione/auto/datao21.owl#Formula1)
4 answers

7) Querying for “has_parts”(property) with Perspective “ O_{V4} ”

Results:

(x, file:///c:/dione/auto/datao21.owl#Car21) (y,
valueCar21has_parts:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao24.owl#Car24) (y,
valueCar24has_parts:http://www.w3.org/2001/XMLSchema#string)

2 answers

8) Querying for “has_engine”(property) with Perspective “O_v4”

Results:

(x, file:///c:/dione/auto/datao24.owl#Car24) (y,
valueCar24has_turbo:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao24.owl#Car24) (y,
valueCar24has_engine:http://www.w3.org/2001/XMLSchema#string)

(x, file:///c:/dione/auto/datao21.owl#Car21) (y,
valueCar21has_engine:http://www.w3.org/2001/XMLSchema#string)

3 answers

NOTE: The type, vpi and global models of Query 3 and 8 are provided in the directory /type_vpi_global.