

## **GraphDB**

## **Esercizi Reasoning**

**Manuel Fiorelli**

**[fiorelli@info.uniroma2.it](mailto:fiorelli@info.uniroma2.it)**

Per prima cosa dobbiamo creare un repository:

- **RepositoryID:** *TestOWL*
- **Ruleset:** *OWL2-RL*
  - Si noti che dopo aver selezionato questo ruleset, *la spunta sull'opzione disable owl:sameAs viene tolta automaticamente* (cioè l'ottimizzazione viene attivata)

# Creazione di un repository (2/6)

GraphDB Workbench

localhost:7200

120%

Choose repository

**Warning:** Some functionality is not available because you are not connected to any repository. Click one of the repositories below to connect to it or create a new repository.

DBpedia_Ontology_cor...	EuroVoc_core • Core reposi...
GDB_TEST1	OpenMultilingualWordne...
RemoteHistory_core • C...	RemoteHistory_suppor...
Test	TestOWL2

**Create new repository**

**License**

# Creazione di un repository (3/6)

GraphDB FREE

Choose repository ▾

## Create Repository

Repository properties

Repository ID\*

Repository title

Type 

Storage folder

Ruleset  Upload custom ruleset

☒ Disable owl:sameAs

Base URL

# Creazione di un repository (4/6)

GraphDB FREE

Import

Explore

SPARQL

Monitor

Setup

Help

## Create Repository

Repository ID\*

Repository title

Type

Storage folder

Ruleset

Base URL

Choose repository

Upload custom ruleset

Disable owl:sameAs

http://example.org/owl

localhost:7200/repository/create?previous=home

120%

# Creazione di un repository (5/6)

Create Repository | GraphDB Web Console

localhost:7200/repository/create?previous=home

Choose repository ▾

## Create Repository

### Repository properties

Repository ID\*

Repository title

Type

Storage folder

Ruleset

☐ Disable owl:sameAs

Base URL

Upload custom ruleset

# Creazione di un repository (6/6)

Create Repository | GraphDB Web Console

localhost:7200/repository/create?previous=home

GraphDB FREE

- Import
- Explore
- SPARQL
- Monitor
- Setup
- Help

indices

Choose repository

☐ Use context index

☒ Enable literal index

☐ Check for inconsistencies

☐ Throw exception on query time-out

☐ Read-only

Entity ID bit-size: 32

Query time-out (seconds): 0

Limit query results: 0

Create Cancel



# Selezione del repository

The screenshot shows the GraphDB Workbench interface. On the left is a sidebar with navigation options: Import, Explore, SPARQL, Monitor, Setup, and Help. The main area displays a message: "Some functionality is not available because connected to any repository. Click one of the repositories below to connect to it or create a new repository." Below this message is a table of available repositories. A hand cursor is pointing at the "TestOWL" repository. At the bottom right, there is a button to "Create new repository".

DBpedia_Ontology_core...	EuroVoc_core • Core reposit...
GDB_TEST1	OpenMultilingualWordnet...
RemoteHistory_core • Cor...	RemoteHistory_support ...
Test	TestOWL
TestOWL2	

License



# Aggiunta di triple RDF con INSERT DATA (1/2)

The screenshot shows the GraphDB SPARQL Query & Update interface. The left sidebar contains navigation options: Import, Explore, SPARQL (highlighted), Monitor, Setup, and Help. The main area displays a SPARQL query in a text editor with line numbers 1 through 15. The query is as follows:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 prefix : <http://example.org/>
3 INSERT DATA {
4     :ClassA a owl:Class .
5     :ClassB a owl:Class .
6
7     :instance1 a :ClassA .
8     :instance2 a :ClassB .
9
10    :Intersection a owl:Class ;
11        owl:intersectionOf (:ClassA :ClassB) .
12
13    :instance1 owl:sameAs :instance2 .
14 }
15

```

On the right side of the query editor, there are several icons: a save icon, a folder icon, a link icon, a double arrow icon, and a graph icon. A red 'Run' button is located at the bottom right of the query editor, with a hand cursor pointing at it. Below the query editor, a light blue banner states: "No results from previous run. Click Run or press Ctrl/Cmd-Enter to execute keyboard shortcuts".

# Aggiunta di triple RDF con INSERT DATA (2/2)

The screenshot shows the GraphDB SPARQL Query & Update interface. The left sidebar contains navigation options: Import, Explore, SPARQL (selected), Monitor, Setup, and Help. The main area displays a SPARQL query in a text editor with line numbers 1 through 15. The query is as follows:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 prefix : <http://example.org/>
3 INSERT DATA {
4     :ClassA a owl:Class .
5     :ClassB a owl:Class .
6
7     :instance1 a :ClassA .
8     :instance2 a :ClassB .
9
10    :Intersection a owl:Class ;
11        owl:intersectionOf (:ClassA :ClassB) .
12
13    :instance1 owl:sameAs :instance2 .
14 }
15
  
```

Below the query editor, a status bar indicates: "Added 11 statements. Update took 0.2s, moments ago." This message is circled in purple. To the right of the query editor, there are icons for saving, opening, linking, and navigating, along with a "Run" button and a hint "Press Alt+Enter to autocomplete".

# Spiegazione dei dati appena aggiunti

- Abbiamo definito due classi (*ClassA* e *ClassB*) e fornito un'istanza per ciascuna di esse (rispettivamente, *instance1* e *instance2*).
- Abbiamo quindi definito la classe *Intersection* come l'intersezione delle due classi citate di sopra
- Abbiamo asserito che le due istanze sono in effetti la stessa cosa (usando il predicato *owl:sameAs*)

# Istanze di *Intersection* senza inferenza (1/2)

SPARQL Query & Update | GraphDB

localhost:7200/sparql

GraphDB FREE

Import

Explore

Unnamed × Unnamed × Unnamed × Unnamed TestOWL v

```

1 prefix : <http://example.org/>
2 select ?s where {
3     ?s a :Intersection
4 }
5

```

Table Raw Response Pivot Table Google Chart

Download as ✓

Filter query results

No results. Query took 0.1s, minutes ago.

s	
No data available in table	

Run

keyboard shortcuts

Nessuna risorsa è *esplicitamente dichiarata* essere un'istanza di Intersection.

# Istanze di *Intersection* con inferenza e espansione owl:sameAs (1/2)

The screenshot shows the GraphDB SPARQL Query & Update interface. The query editor contains the following SPARQL query:

```
prefix : <http://example.org/>
select ?s where {
  ?s a :Intersection
}
```

The results are displayed in a table with the following data:

	s
1	<a href="http://example.org/instance1">http://example.org/instance1</a>
2	<a href="http://example.org/instance2">http://example.org/instance2</a>

A red box labeled "Download as" is highlighted, and a purple arrow points from it to a red "Run" button. A hand icon is shown clicking the "Run" button.

# Istanze di *Intersection* con inferenza e espansione owl:sameAs (2/2)

*instance1* è dichiarata esplicitamente di tipo *ClassA*, ma essendo stata dichiarata coincidente con *instance2*, che è di tipo *ClassB*, possiamo inferire che *instance1* è anche di tipo *ClassB*. Pertanto, si può inferire che *instance1* appartiene alla classe *Intersection*.

Un ragionamento analogo ci permette di inferire che *instance2* appartiene alla classe *Intersection*.

# Istanze di *Intersection* con inferenza e senza espansione owl:sameAs (1/2)

The screenshot shows the GraphDB SPARQL Query & Update interface. The query editor contains the following SPARQL query:

```
prefix : <http://example.org/>
select ?s where {
  ?s a :Intersection
}
```

The results are displayed in a table view. The table has one column labeled 's' and one row with the value 'http://example.org/instance1'. A red box labeled 'Download as' is highlighted, and a purple arrow points to it. A red box labeled 'Run' is also highlighted, with a black hand icon pointing to it. The interface includes a sidebar with 'Import' and 'Explore' buttons, and a top bar with 'TestOWL' and 'keyboard shortcuts' links.

	s
1	<a href="http://example.org/instance1">http://example.org/instance1</a>



# Istanze di *Intersection* con inferenza e senza espansione owl:sameAs (2/2)

GraphDB possiede un'implementazione non rule-based del predicato owl:sameAs.

- Tiene traccia delle classi di equivalenza generate da owl:sameAs, associando a ciascuna di esse un *rappresentante* (tra le risorse che vi appartengono)
- Rappresenta ciascuno statement una sola volta sostituendo a ciascun componente i rappresentati delle classi di equivalenza cui appartengono
- Durante la valutazione di una query, GraphDB utilizza una sorta di backward-chaining enumerando tutti gli IRI equivalenti, garantendo la completezza dell'inferenza e dei risultati.
- L'implementazione di questa feature è tale da permettere di distinguere gli statement espliciti da quelli impliciti.

Per maggiori dettagli si consulti:

<http://graphdb.ontotext.com/documentation/free/sameas-optimisation.html>

Da notare l'uso della clausola *FROM onto:disable-sameAs* per disabilitare da dentro la query SPARQL l'espansione di owl:sameAs (utile quando si sottomette la query tramite API REST)

# Aggiunta di altre triple (vedi esercizi OWL 9 e 10)

The screenshot shows the GraphDB SPARQL Query & Update interface. The left sidebar contains navigation options: Import, Explore, SPARQL (selected), Monitor, Setup, and Help. The main area displays a SPARQL query in a text editor with line numbers 1 through 15. The query is as follows:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 prefix : <http://example.org/>
3 INSERT DATA {
4     :grandparent a owl:ObjectProperty ;
5         owl:propertyChainAxiom (:parent :parent ) .
6     :parent a owl:ObjectProperty .
7     :philipp a owl:Thing ;
8     :parent :john , :mary .
9     :alice a owl:Thing ;
10        :parent :luke , :janet .
11    :mary a owl:Thing ;
12        :parent :frank , :jessica .
13    :luke a owl:Thing ;
14        :parent :frank , :jessica .
15 }

```

On the right side of the editor, there are icons for saving, opening, linking, and running the query. A red 'Run' button is located at the bottom right of the query area. Below the query editor, a status bar indicates 'No results from previous run. Click Run or press Ctrl/Cmd-Enter to execute keyboard shortcuts'.

# Query su property chain (senza inferenza)

SPARQL Query & Update | GraphDB

localhost:7200/sparql

1 prefix : <http://example.org/>  
2  
3 select \* where {  
4 ?s :grandparent ?o .  
5 }

Table Raw Response Pivot Table Google Chart

Download as

Filter query results

No results. Query took 0.1s, moments ago.

s	o
No data available in table	

Run

keyboard shortcuts

# Query su property chain (con inferenza)

The screenshot shows the GraphDB SPARQL Query & Update interface. The query executed is:

```
prefix : <http://example.org/>
select * where {
  ?s :grandparent ?o .
}
```

The results table shows 4 rows of data:

	s	o
1	http://example.org/philipp	http://example.org/frank
2	http://example.org/philipp	http://example.org/jessica
3	http://example.org/alice	http://example.org/frank
4	http://example.org/alice	http://example.org/jessica

A purple arrow points from the 'Run' button in the right sidebar to the results table. The 'Run' button is a red rectangle with the text 'Run' in white. The right sidebar also contains icons for Import, Explore, and other database functions.

# Query *DESCRIBE* su *grandparent* (1/2)

The screenshot shows the GraphDB SPARQL Query & Update interface. The browser address bar indicates the URL is `localhost:7200/sparql`. The left sidebar contains navigation options: Import, Explore, SPARQL (selected), Monitor, Setup, and Help. The main query editor displays the following SPARQL query:

```
1 prefix : <http://example.org/>
2
3 describe :grandparent
```

Below the query editor, there are icons for saving, opening, linking, navigating, and a graph view. A red 'Run' button is visible, with a hand cursor pointing at it. At the bottom, there are tabs for 'Table', 'Raw Response', 'Pivot Table', and 'Google Chart', along with a 'keyboard shortcuts' link.

# Query *DESCRIBE* su *grandparent* (2/2)

Filter query results

Showing results from 1 to 6 of 6. Query took 0.1s, minutes ago.

	subject	predicate	object
1	<code>_:node10</code>	<code>rdf:first</code>	<code>http://example.org/parent</code>
2	<code>_:node10</code>	<code>rdf:rest</code>	<code>rdf:nil</code>
3	<code>_:node9</code>	<code>rdf:first</code>	<code>http://example.org/parent</code>
4	<code>_:node9</code>	<code>rdf:rest</code>	<code>_:node10</code>
5	<code>http://example.org/grandparent</code>	<code>rdf:type</code>	<code>owl:ObjectProperty</code>
6	<code>http://example.org/grandparent</code>	<code>owl:propertyChainAxiom</code>	<code>_:node9</code>

L'oggetto della tripla col predicato *owl:propertyChainAxiom* è un blank node che rappresenta una RDF Collection attraverso un approccio molto simile a quello usato da Prolog: una lista contiene un elemento (*rdf:first*) e un riferimento al resto della lista (*rdf:rest*) (si noti che *rdf:nil* è la lista vuota)