

Indice

<i>Abstract</i>	2
<i>Ontologia</i>	2
<i>Triplificazione di una tabella relazionale</i>	3
<i>GraphDB</i>	6
<i>Sparql</i>	7
Property Path	9

Abstract

Scopo del progetto è quello di sperimentare tecnologie del web semantico in relazione ai modelli di dati relazionali, analizzando un caso d'uso pratico inerente al cibo. In particolare, il cibo è descritto in un'ottica di composizione delle ricette di cucina, ingredienti e proprietà chimico/biologico.

Ontologia

Per quanto riguarda la creazione dell'ontologia sul cibo è stata utilizzata quella già presente nel web: FoodKG. Tale ontologia risulta un sotto insieme dell'ontologia Food-ON. Quest'ultima risulta la più importante ontologia a livello mondiale presente sotto forma di open-data nel web. Una volta reperita e studiata l'ontologia base questa è stata ampliata, come verrà descritto in seguito.

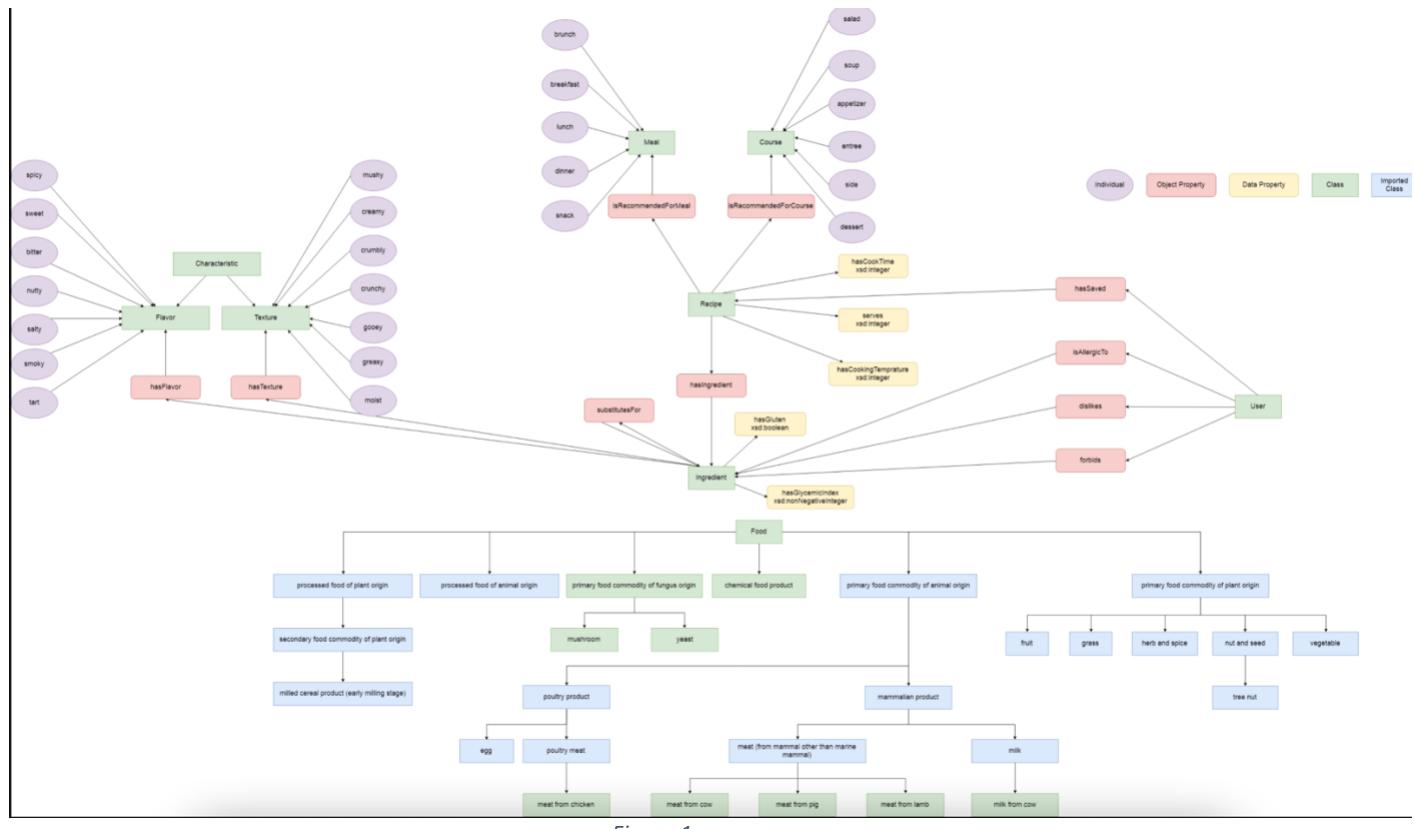


Figura 1

Nell'immagine numero 1 è riportata la struttura grafica che riassume le informazioni dell'ontologia. Le due entità principali sono "Recipe" e "Ingredient". Queste, infatti, hanno il maggior numero di archi entranti e uscenti risultando in relazione con tutte le entità secondarie.

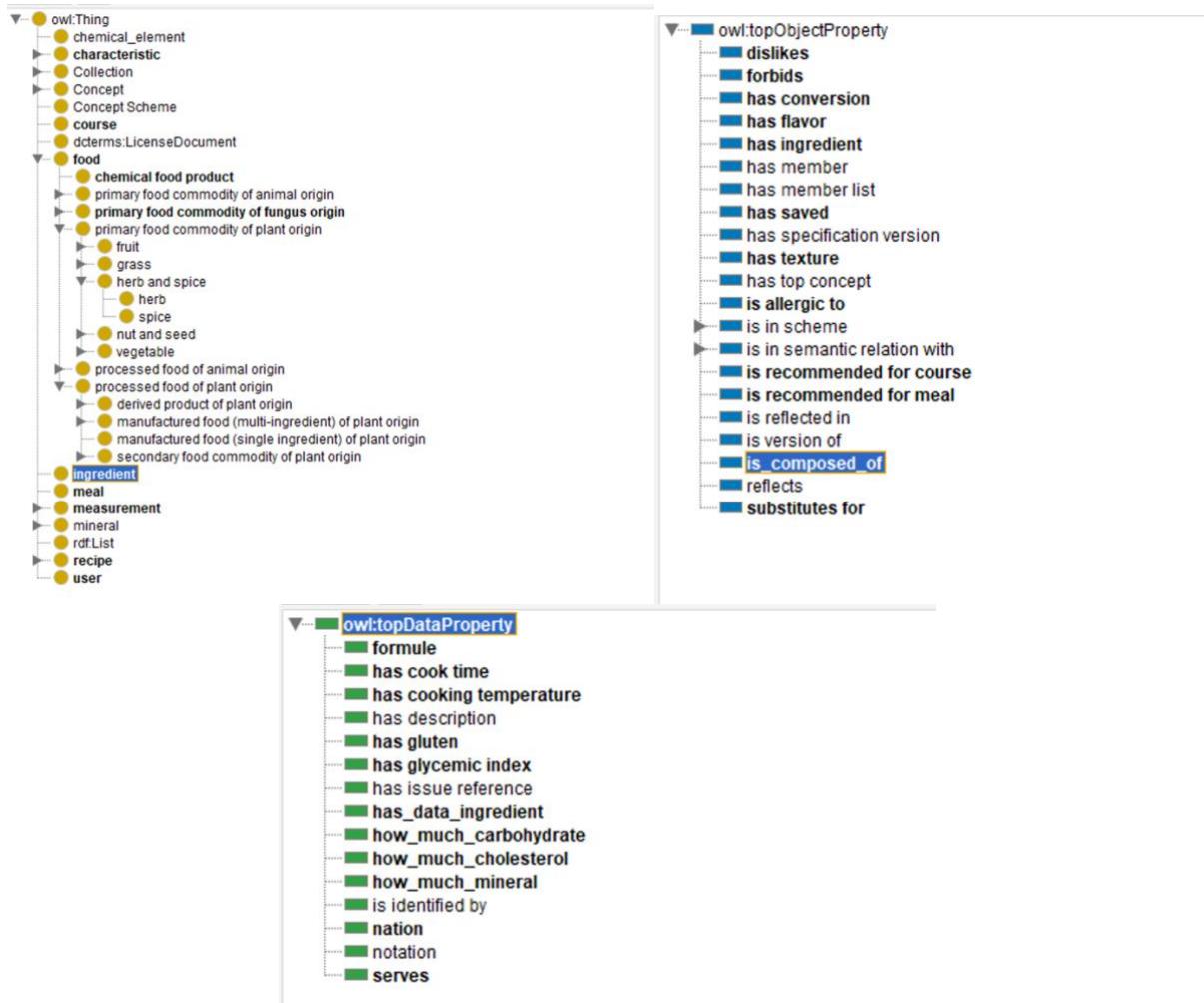


Figura 2

Nell' immagine sopra, si osservano le caratteristiche principali dell'ontologia rimaneggiata:

1. -Le classi e le relative gerarchie.
2. -Le Object Properties (che mettono in relazione i vari individuals delle classi)
3. -Le Data Properties (che mettono in relazione gli individuals con i literal)

All'ontologia dell'immagine 1 è stato aggiunta una classe “chemical_element”, un objectProperty “`is_composed_of`” e una dataProperty “`nation`”.

Triplificazione di una tabella relazionale

Per mappare una tabella con semantica relazionale in un insieme di triple è stato usato il tool Karma. Tale strumento, oltre a permettere una mappatura della semantica, permette anche di fare delle modifiche al sistema relazionale. In particolare, la prima modifica fatta alla tabella è stata quella di risolvere un conflitto strutturale (conflitto di dipendenze). La tabella in questione, infatti, possiede tre campi:

1. -id (id della ricetta)
2. -cuisine (nazione a cui appartiene la ricetta)
3. -ingredients (elenco ingredienti della ricetta)

La modifica sul conflitto riguarda gli ingredienti che nella tabella originale sono espressi nella modalità: ingrediente1, ingrediente2, ingrediente3. Vi è stata quindi una normalizzazione del campo ingrediente per renderla coerente con il modello dell'ontologia. Per fare ciò, Karma mette a disposizione una funzionalità di split come si osserva nell'immagine:

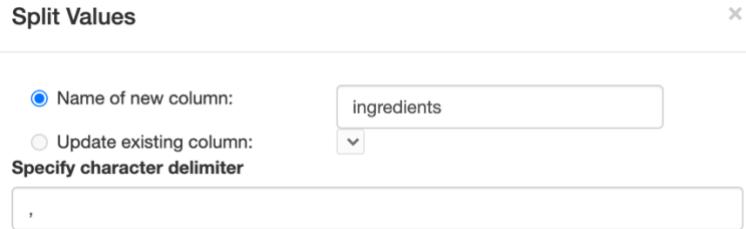


Figura 3

Specificando il delimitatore (in questo caso il carattere ",") karma splitta automaticamente i valori in più campi, cambiando così la molteplicità da 1:1 a 1:N.

Il secondo passaggio di trasformazione della tabella originale riguarda la creazione di due nuove colonne: "id uri" e "ingredient uri". Questo passaggio risulta fondamentale per associare a ogni entità un uri coerente con quello dell'ontologia da mappare. Infatti, le entità ingredienti hanno un uri nella forma:

- <http://purl.org/heals/ingredient/> + valore (per le istanze di ingredienti)
- <http://purl.org/heals/food/> + valore (per le istanze delle ricette)

Per fare ciò, karma mette a disposizione la funzionalità PyTransform. Tale funzionalità permette di creare una nuova colonna a partire da una già esistente. In questa maniera (come si vede nell'immagine numero 6 nella colonna id uri o id ingredient ad esempio) viene creata una nuova colonna (colorata di giallo) che ha il pattern generato dall'algoritmo dell'immagine numero 4.

```
1 return"/ingredient/ getValue("id")
```

Figura 4

Infine, andrà specificato il Base URI (Immagine numero 5) da cui verranno generate gli uri assoluti come concatenazione del Base URI con ogni record della colonna "id URI".



Figura 5

Se non viene specificato nessun arco "URI" di default karma assocerà l'entità a un blank-node (https://it.wikipedia.org/wiki/Risorsa_anonima).

A questo punto, selezionando la colonna id uri e specificando (attraverso la specifica interfaccia) che questa corrisponde all'uri di una determinata classe, è conclusa la prima parte del mapping che associa a ogni istanza dell'entità un identificativo univoco e coerente con quello dell'ontologia.

Il passo successivo è quello di associare alle rimanenti colonne le informazioni relative l'ontologia FoodKG.

Per fare ciò si procede similmente a quanto fatto precedentemente per associare l'uri a una classe. In particolare, si possono selezionare i nodi o le colonne e specificare di immettere un arco in entrata o in uscita. Ogni arco corrisponde a una data property o un object property.

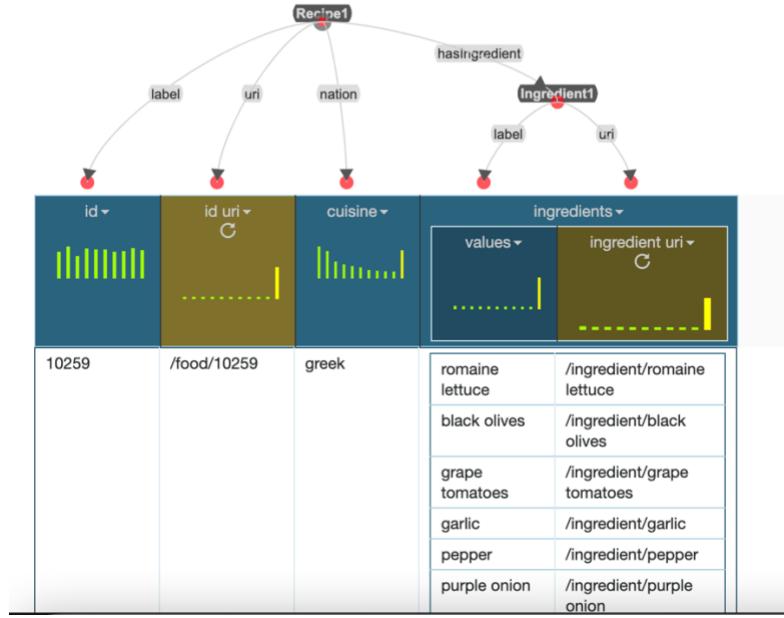


Figura 6

Una volta costruito il grafo di mapping, vi sarà la possibilità di ottenere le triple in formato RDF o un modello r2rml. Nell'immagine numero 7 si osserva un frammento delle triple risultanti dalla mappatura spiegata precedentemente.

A questo punto tale file sarà importabile in un qualsiasi triple store e i dati saranno integrati con l'ontologia già presente.

```

<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/jalapeno%20chilies> .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/white%20vinegar> .
<http://purl.org/heals/ingredient/olive%20oil> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/ingredient/chopped%20cilantro> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/ingredient/pepper> <http://www.w3.org/2000/01/rdf-schema#label> "spanish" .
<http://purl.org/heals/ingredient/chorizo%20sausage> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/ingredient/chorizo%20sausage> <http://www.w3.org/2000/01/rdf-schema#label> "chorizo sausage" .
<http://purl.org/heals/ingredient/pepper> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/ingredient/flat%20leaf%20parsley> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/ingredient/garlic> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/ingredient/chopped%20cilantro> <http://www.w3.org/2000/01/rdf-schema#label> "chopped cilantro" .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/garlic> .
<http://purl.org/heals/ingredient/flat%20leaf%20parsley> <http://www.w3.org/2000/01/rdf-schema#label> "flat leaf parsley" .
<http://purl.org/heals/ingredient/sea%20salt> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/food/42779> <http://www.w3.org/2000/01/rdf-schema#label> "sea salt" .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/salt> .
<http://purl.org/heals/ingredient/salt> <http://www.w3.org/2000/01/rdf-schema#label> "salt" .
<http://purl.org/heals/ingredient/jalapeno%20chilies> <http://www.w3.org/2000/01/rdf-schema#label> "jalapeno chilies" .
<http://purl.org/heals/ingredient/white%20vinegar> <http://www.w3.org/2000/01/rdf-schema#label> "white vinegar" .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/bay%20leaf> .
<http://purl.org/heals/ingredient/white%20vinegar> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/ingredient/medium%20shrimp> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/chopped%20cilantro> .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/medium%20shrimp> .
<http://purl.org/heals/ingredient/garlic> <http://www.w3.org/2000/01/rdf-schema#label> "garlic" .
<http://purl.org/heals/ingredient/salt> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/food/42779> <http://www.w3.org/2000/01/rdf-schema#label> "42779" .
<http://purl.org/heals/ingredient/skirt%20steak> <http://www.w3.org/2000/01/rdf-schema#label> "skirt steak" .
<http://purl.org/heals/ingredient/bay%20leaf> <http://www.w3.org/2000/01/rdf-schema#label> "bay leaf" .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/pepper> .
<http://purl.org/heals/ingredient/olive%20oil> <http://www.w3.org/2000/01/rdf-schema#label> "olive oil" .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/skirt%20steak> .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/sea%20salt> .
<http://purl.org/heals/ingredient/skirt%20steak> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/olive%20oil> .
<http://purl.org/heals/food/42779> <http://purl.org/heals/food/hasIngredient> <http://purl.org/heals/ingredient/chorizo%20sausage> .
<http://purl.org/heals/ingredient/medium%20shrimp> <http://www.w3.org/2000/01/rdf-schema#label> "medium shrimp" .
<http://purl.org/heals/ingredient/jalapeno%20chilies> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/heals/food/Ingredient> .

```

Figura 7

Una piccola nota tecnica riguarda il formato del file rdf generato, in particolare è importante stare attenti ad alcuni caratteri quali “%%” generati automaticamente dal tool quando incontra un doppio spazio nel campo di un record. Tali caratteri non possono essere parsati dal triple-store.

GraphDB

GraphDB è un database di grafi semanticci. I database a grafo semantico (chiamati anche triplestore RDF) forniscono l'infrastruttura di base per soluzioni in cui l'agilità di modellazione secondo lo standard del web semantico è reso semplice e veloce.

All'interno di GraphDB vi è la possibilità di importare ontologie, di visualizzare graficamente le triple come insieme di nodi e eventualmente di importare un insieme di regole per generare inferenza (Immagine 8).

Inference and Validation



Figura 8

Le ontologie e eventuali altri dati possono essere importate nella sezione “Import” dove dovrà essere specificato l'iri del grafo base. Vi è la possibilità di importare i dati anche da un end-point specificato come URL. In tal caso github risulta un ottimo repository dal quale attingere per importare i dati da un URL (importante che l'URL sia quello del formato “raw”, i.e. <https://raw.githubusercontent.com>...).

Figura 9

Infine, GraphDB mette a disposizione un’interfaccia per fare query SPARQL e una UI (VisualGraph) per osservare graficamente l’insieme delle triple come nodi e relativi archi (come si osserva nell’immagine numero 13 o 17).

Sparql

Nell’immagine sotto è riportato un esempio di query che ottiene tutte le ricette che hanno indice glicemico compreso tra 50 e 100 e che non hanno glutine.

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ingredient: <http://purl.org/heals/ingredient/>
PREFIX food: <http://purl.org/heals/food/>

select * where {
  #FoodProduct
  ?s food:hasGlycemicIndex ?GI.
    FILTER (?GI >= 50 && ?GI <= 100).
  ?s food:hasGluten false.

} limit 100

```

Figura 10

Si può osservare un altro esempio di query nell'immagine numero 11.

The screenshot shows a SPARQL query interface with the following details:

- Query Text:**

```

FOODKG-find food based on meal
1 PREFIX food: <http://purl.org/heals/food/>
2 PREFIX ingredient: <http://purl.org/heals/ingredient/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 SELECT DISTINCT ?recipe ?meal
5 WHERE {
6 ?recipe food:hasIngredient ingredient:Beef .
7 ?recipe food:hasIngredient ingredient:Carrot .
8 ?recipe food:isRecommendedForMeal ?meal .
9 }
10
11
12
13
14
    
```
- Run Button:** A red "Run" button is located in the top right corner.
- Results Table:**

recipe	meal
ind:BeefNilaga	wtm:Dinner
ind:BeefNilaga	wtm:Lunch
ind:BeefStew	wtm:Dinner
ind:BeefStew	wtm:Lunch
- Buttons:** On the right side, there are several orange icons for file operations (Save, Open, Copy, Print, etc.) and a "Run" button.

Figura 11

Tale interrogazione al sistema ritorna le ricette che hanno come ingredienti sia le carote che la carne e a che pasto conviene consumare tali ricette.

Nella query sotto si osserva un conteggio per ottenere una statistica: Nazioni per frequenza di uso di sale nelle ricette.

The screenshot shows a SPARQL query interface with the following details:

- Query Text:**

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ingredient: <http://purl.org/heals/ingredient/>
PREFIX food: <http://purl.org/heals/food/>
PREFIX ind: <http://purl.org/heals/ingredient/>

select ?nation (COUNT(?o) as ?frequencySaltInNationRecipe) where {
?nation food:hasIngredient ingredient:salt.
?nation food:nation ?o.
#Ritorna gli stati per frequenza di uso del sale nelle ricette
}
group by ?nation
    
```
- Run Button:** A red "Run" button is located in the top right corner.
- Results Table:**

nation	frequencySaltInNationRecipe
"italian"	"60"^^xsd:integer
"mexican"	"55"^^xsd:integer
"southern_us"	"47"^^xsd:integer
"indian"	"37"^^xsd:integer
"french"	"23"^^xsd:integer
- Buttons:** On the right side, there are several orange icons for file operations (Save, Open, Copy, Print, etc.) and a "Run" button.

Figura 12

Property Path

Nell'immagine numero 13 è riportato il grafo di una ricetta ("Backed chicken tender"), dei suoi ingredienti e le relazioni chimiche che questi hanno. In particolare, si può osservare un percorso unidirezionale che parte da Backed chicken tender e giunge fino a Sodium. Tale relazione risulta utile per spiegare la query dell'immagine 14 che fa uso dell'operatore "/". Tale operatore permette di costruire un percorso tra nodi (Property Path). In particolare, la query ritornerà tutti i nodi che hanno come primo arco quello con predicato "hasIngredient", come secondo arco "is_composed_of", come terzo arco "is_composed_of" e che, infine, questo punterà all'individual "Sodium". Traducendo la query in linguaggio naturale, questa risulta come: estrai dal grafo tutte le ricette che hanno ingredienti e che sono composti da qualcosa che a sua volta è composto dal Sodio.

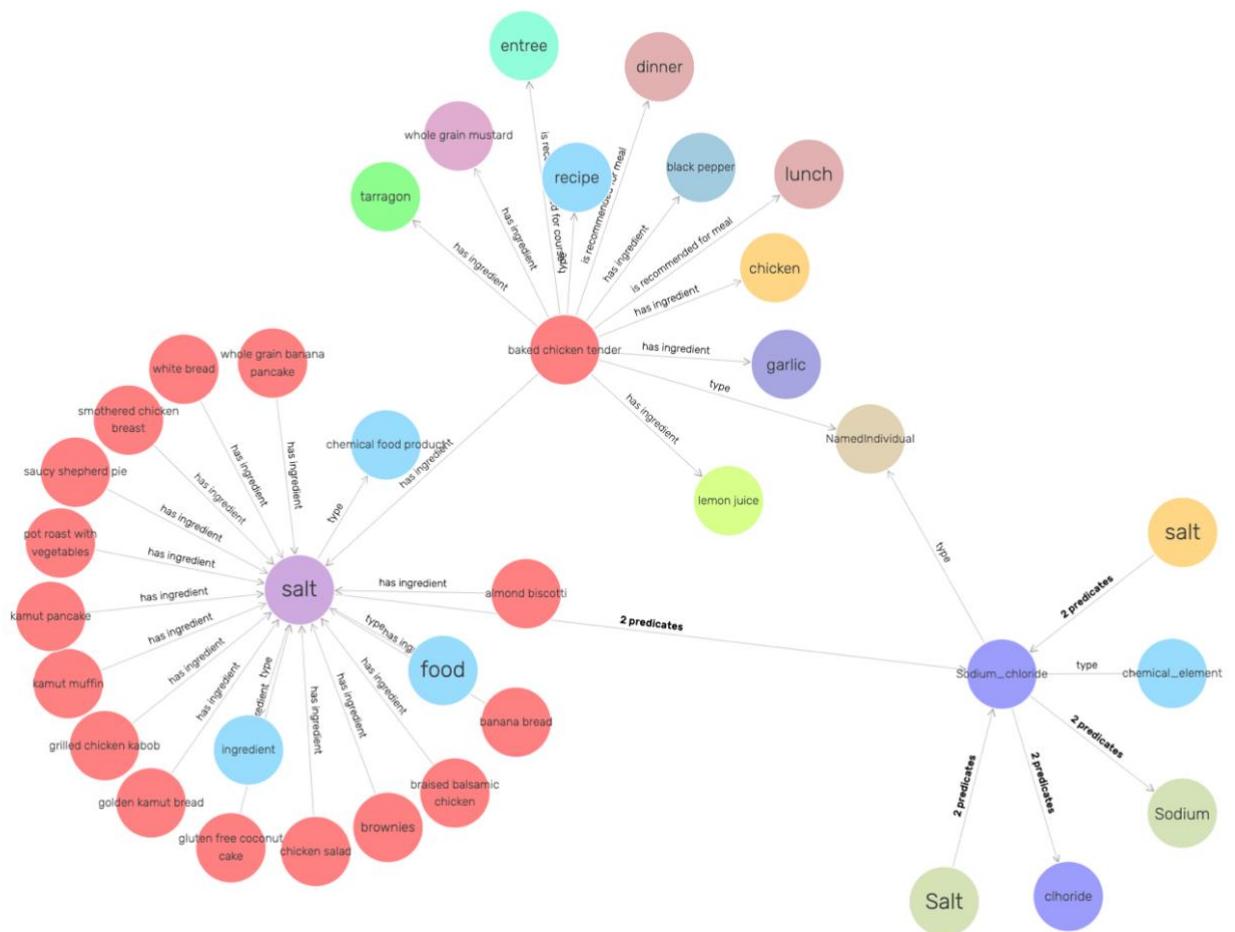


Figura 13

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ingredient: <http://purl.org/heals/ingredient/>
PREFIX food: <http://purl.org/heals/food/>

select * where {
    ?s food:hasIngredient/food:is_composed_of/food:is_composed_of food:Sodium.
}

```

Figura 14

Un altro esempio di query che esplicita la ricerca di un percorso è quella dell'immagine sotto. In questo caso oltre che all'operatore “/” vi è anche quello di “^”. Ciò permette di creare un percorso invertendo l'ordine di puntamento dell'arco. La query si può tradurre come: ritorna tutte le ricette che hanno come ingredienti in comune con le ricette a base di pollo.

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ingredient: <http://purl.org/heals/ingredient/>
PREFIX food: <http://purl.org/heals/food/>

select * where {

    ?chickenRecipes food:hasIngredient ingredient:Chicken.
    ?chickenRecipes food:hasIngredient/^food:hasIngredient ?recipeWithCommonIngredients.
    #Ritorna gli ingredienti che le ricette dei piatti a base di pollo hanno con le altre ricette
}

```

Figura 15

Nell'immagine sotto, invece, si osserva la query equivalente a quella dell'immagine 15 senza gli operatori del Property Path.

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ingredient: <http://purl.org/heals/ingredient/>
PREFIX food: <http://purl.org/heals/food/>
PREFIX ind: <http://purl.org/heals/ingredient/>

select * where {
    ?chickenRecipe food:hasIngredient ingredient:Chicken.
    ?chickenRecipe food:hasIngredient ?otherIngredientOfChickenRecipe.
    ?recipeWithCommonChickenRecipeIngredients food:hasIngredient ?otherIngredientOfChickenRecipe.
    #Ritorna gli ingredienti che le ricette dei piatti a base di pollo hanno con le altre ricette
}

```

Figura 16

L'immagine numero 17 mostra il grafo che mette in relazione il pollo con le sue ricette e i relativi ingredienti.

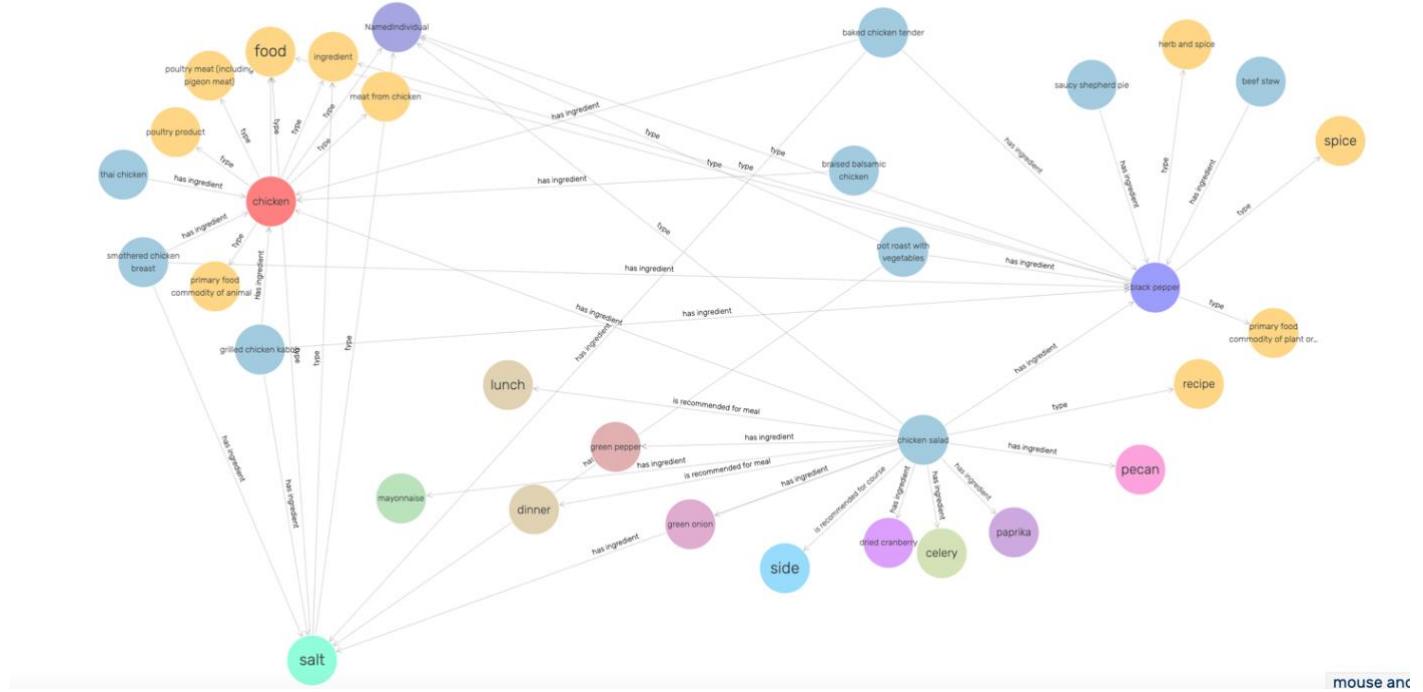


Figura 17