

Interrogazioni basate su graph pattern

1 Classificazione dei graph pattern

Sia gli edge-labelled graph (come già visto con SPARQL) che i property graph possono essere interrogati usando i **graph pattern**. Si possono distinguere due principali tipi di graph pattern:

basic graph pattern: un grafo in cui, al posto dei valori delle label (e anche delle proprietà, nel caso di un property graph), possono apparire delle variabili;

complex graph pattern: costruito a partire da uno o più basic graph pattern, mediante operazioni di proiezione, unione, join, ecc.

2 Matching

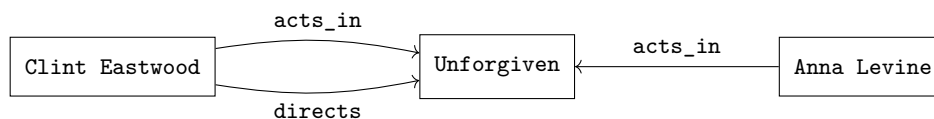
Un **matching** tra un bgp (basic graph pattern, cioè l'interrogazione) e un property graph (cioè il database) è un'operazione di **mapping** / sostituzione dalle variabili a delle costanti / valori che, quando effettuata sul bgp, produce come risultato un sottografo del property graph.

Il risultato di un'interrogazione fatta con un bgp è costituito da tutti i mapping (dalle variabili del bgp alle costanti del property graph) con i quali si ha un match (cioè, appunto, si ottiene un sottografo del property graph).

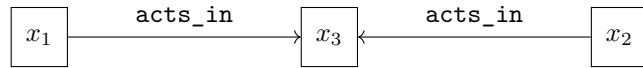
Lo stesso meccanismo si applica anche nel caso (più semplice) degli edge-labelled graph.

2.1 Esempi

- Sia G il seguente edge-labelled graph:



Si vogliono trovare tutte le coppie di attori che recitano in un film in comune. Il bgp Q corrispondente a tale interrogazione è:

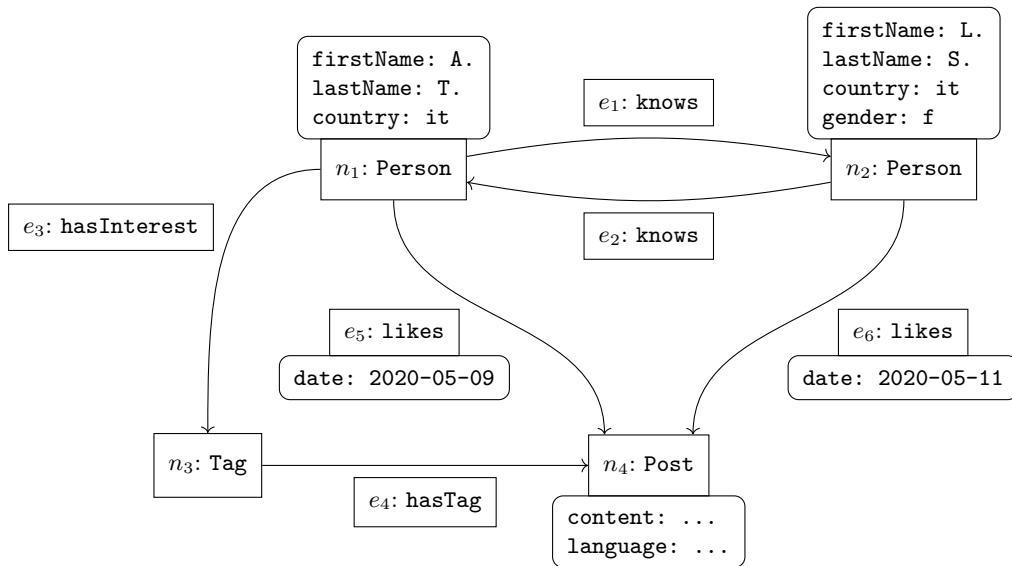


Il risultato $Q(G)$ del matching sarà allora:

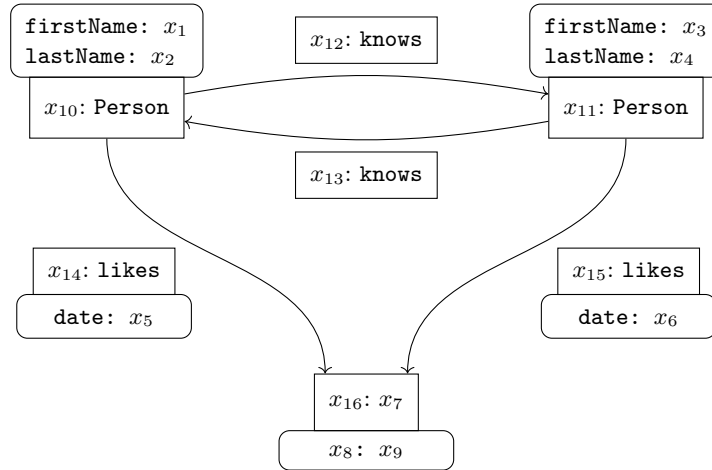
x_1	x_2	x_3
Clint Eastwood	Anna Levine	Unforgiven
Anna Levine	Clint Eastwood	Unforgiven
Clint Eastwood	Clint Eastwood	Unforgiven
Anna Levine	Anna Levine	Unforgiven

Le ultime due righe di questa tabella corrispondono a dei mapping nei quali le variabili x_1 e x_2 sono state sostituite con la stessa costante. Alcuni linguaggi di interrogazione ammettono tali mapping, mentre altri impongono il vincolo che variabili diverse vengano sostituite con costanti diverse. In altre parole, ci sono vari modi di eseguire il matching, ed essi possono dare risultati diversi, anche se il grafo interrogato e il bgp sono gli stessi.

- Adesso, si considera invece un property graph G che modella una semplice rete sociale:



Si vogliono trovare tutte le cose (oggetti della rete sociale) che piacciono a coppie di amici, estraendo dal grafo varie proprietà (nome e cognome degli amici, date dei like, ecc.). A tale scopo, si costruisce il seguente bgp Q :



Il risultato $Q(G)$ ottenuto dal matching è:

x_1	x_2	x_3	x_4	x_5	x_6	x_7	...
A.	T.	L.	S.	2020-05-09	2020-05-11	Post	...
L.	S.	A.	T.	2020-05-11	2020-05-09	Post	...
A.	T.	A.	T.	2020-05-09	2020-05-09	Post	...
L.	S.	L.	S.	2020-05-11	2020-05-11	Post	...

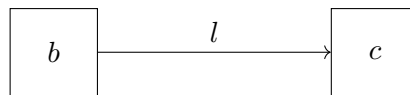
2.2 Definizione formale

Definizione: Un *match* tra un edge-labelled graph $G = (V, E)$ – dove V è l'insieme di nodi (o vertici), ed E è l'insieme di archi (edges) – e un basic graph pattern $Q = (V', E')$, nel quale V' ed E' possono contenere variabili al posto delle costanti, è un mapping

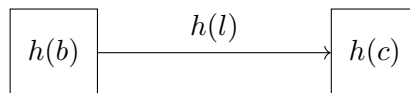
$$h : Costanti \cup Variabili \rightarrow Costanti$$

tale che:

1. Per ogni costante $a \in Costanti$, $h(a) = a$ (cioè ogni costante viene mappata a sé stessa).
2. Per ogni arco del graph pattern $(b, l, c) \in E'$,



nel grafo interrogato è presente un corrispondente arco $(h(b), h(l), h(c)) \in E$.



In altre parole, ogni arco di Q è mappato a un arco di G . Di conseguenza, quando il mapping è applicato a tutte le variabili e costanti di Q , il risultato è un sottografo di G .

Nel caso degli edge-labelled graph, le costanti (e quindi i “posti” dove, nel bgp, possono comparire le variabili) sono le label dei nodi e degli archi. Per i property graph, la definizione è analoga, ma le costanti sono:

- gli identificativi dei nodi e degli archi;
- le label dei nodi e degli archi;
- le proprietà e i loro valori.

2.3 Matching basato su omomorfismi e isomorfismi

Tecnicamente, un mapping, cioè una mappa h che soddisfa le proprietà 1 e 2 appena enunciate, è un **omomorfismo**. In particolare, è importante notare che tali proprietà *non* richiedono che nodi/archi diversi di Q vengano mappati in nodi/archi diversi di G . Se si aggiunge tale vincolo, allora h è un **isomorfismo**.

Il fatto che il matching sia basato su omomorfismi o su isomorfismi dipende dal linguaggio di interrogazione. Inoltre, tra i linguaggi che si basano sugli isomorfismi, si distinguono tre semantiche diverse:

semantica no-repeated-anything: variabili diverse di Q devono essere legate a nodi/archi diversi di G ;

semantica no-repeated-node: ciascuna variabile di nodo in Q deve essere legata a un nodo diverso in G , mentre più variabili possono essere legate allo stesso arco;

semantica no-repeated-edge: variabili diverse possono essere legate a uno stesso nodo, ma non a uno stesso arco.

Un linguaggio con matching basato su omomorfismi è SPARQL, mentre Cypher implementa il matching basato su isomorfismi con la semantica no-repeated-edge.