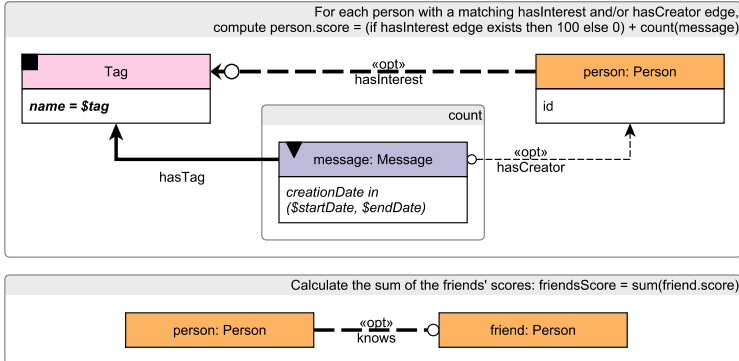


BI / read / 8

query	BI / read / 8			
title	Central person for a tag			
pattern	<div><div>For each person with a matching hasInterest and/or hasCreator edge, compute person.score = (if hasInterest edge exists then 100 else 0) + count(message)</div></div>			
description	<p>Given a \$tag, find all Persons that are interested in the \$tag and/or have written a Message (Post or Comment) with a creationDate after a given \$startDate and that has a given \$tag. For each Person, compute the score as the sum of the following two aspects:</p> <ul style="list-style-type: none">• 100, if the Person has this \$tag as their interest, or 0 otherwise• number of Messages by this Person with the given \$tag <p>Also, for each Person, compute the sum of the score of the Person’s friends (friendsScore).</p>			
params	<div><div>1</div><div>\$tag</div><div>Long String</div></div>	<div>Tags with a similar amount of Messages are selected</div>		
	<div><div>2</div><div>\$startDate</div><div>Date</div></div>	<div>(a): A range during which a flashmob event happened (it should yield at least a 5× difference)</div>		
	<div><div>3</div><div>\$endDate</div><div>Date</div></div>	<div>(b): A regular range (does not include a flashmob event)</div>		
result	<div><div>1</div><div>person.id</div><div>ID</div></div>	<div>R</div>		
	<div><div>2</div><div>score</div><div>32-bit Integer</div></div>	<div>A</div>		
	<div><div>3</div><div>friendsScore</div><div>32-bit Integer</div></div>	<div>A</div>	<div>The sum of the score of the person’s friends</div>	
sort	<div><div>1</div><div>score + friendsScore</div><div>↓</div></div>			
	<div><div>2</div><div>person.id</div><div>↑</div></div>			
limit	100			
CPs	1.2, 2.1, 2.3, 3.2, 5.3, 8.2, 8.4, 8.5			
relevance	Similarly to BI 16, there are two major ways to compute this query: (1) creating an induced subgraph of the interested Persons and their friends and performing the scoring on this graph or (2) performing the scoring without creating an induced subgraph and scoring the friends of a Person on-the-fly. The first approach is more efficient as it avoids redundant computations, however, specifying it needs support for composable graph queries.			