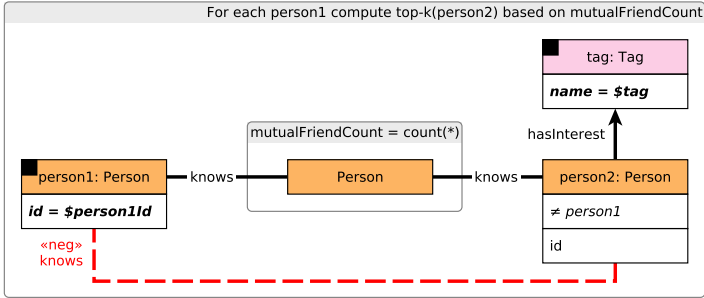


BI / read / 18

BI 1	query	BI / read / 18			
BI 2	title	Friend recommendation			
BI 3	pattern	<div>For each person1 compute top-k(person2) based on mutualFriendCount</div>  <pre>graph LR     person1[person1: Person] -- knows --&gt; Person[Person]     Person -- knows --&gt; person2[person2: Person]     person2 -- hasInterest --&gt; tag[tag: Tag]     tag -- "name = \$tag" --&gt; person2     person1 -. "«neg» knows" .-&gt; person2</pre>			
BI 4					
BI 5					
BI 6					
BI 7					
BI 8					
BI 9					
BI 10	desc.	For a given Person (person1) and a Tag (tag), recommend new friends (person2) who			
BI 11		<ul style="list-style-type: none"><li>do not yet know person1</li><li>have many mutual friends with person1</li><li>are interested in tag.</li></ul>			
BI 12		Rank Persons person2 based on the number of mutual friends.			
BI 13					
BI 14					
BI 15					
BI 16					
BI 17					
BI 18	params	1	person1Id	ID	Persons with a similar amount of friends are selected
BI 19		2	tag	Long String	Tags with a similar amount of Messages are selected
BI 20	result	1	person2.id	ID	R
		2	mutualFriendCount	32-bit Integer	A
	sort	1	mutualFriendCount	↓	
		2	person2.id	↑	
	limit	20			
	CPs	2.5, 8.1			