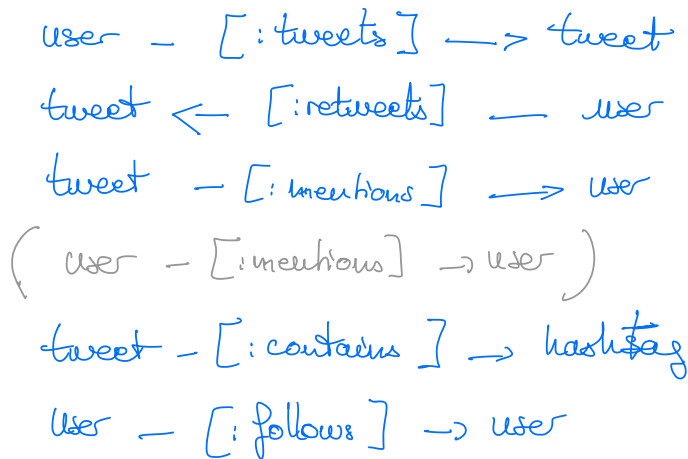


# 1. MODELISATION

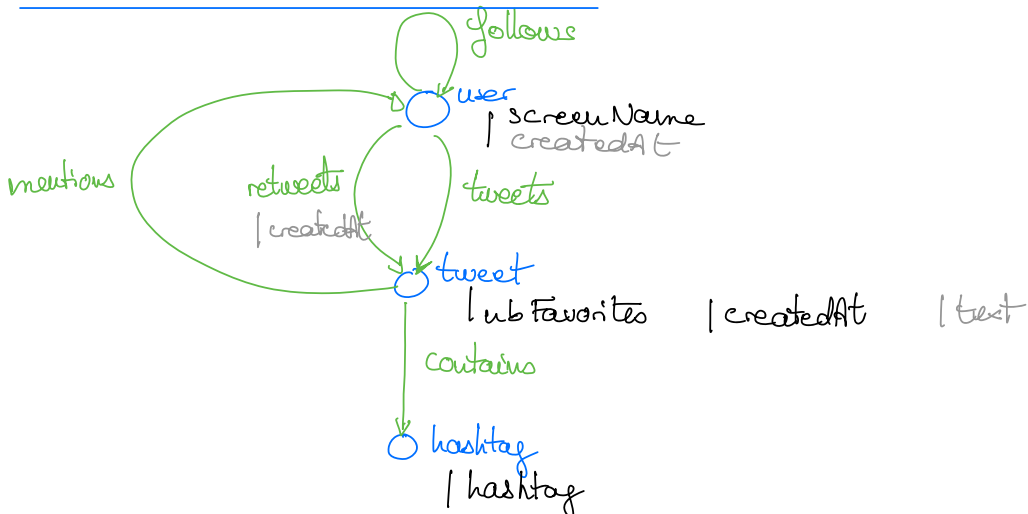
- quels sont les tweets les plus retweetés, ?
- quels sont les utilisateurs les plus suivis, ?
- quels sont les utilisateurs les plus mentionnés, ?
- quels sont les hashtags les plus populaires, ?

=> tweet, utilisateur, hashtag. (NOEUDS)

- quels sont les tweets les plus retweetés, ?
- quels sont les utilisateurs les plus suivis, ?
- quels sont les utilisateurs les plus mentionnés, ?
- quels sont les hashtags les plus populaires, ? (= contenus dans le plus de tweets ?)



## CONSOLIDATION DES CHEMINS



## 2. EXEMPLES DE REQUETES

- Q1. le nombre d'utilisateurs
- Q2. le nombre de tweets contenant le hashtag *WhiteRabbit*
- Q3. le nombre d'utilisateurs différents qui ont tweeté un tweet contenant le hashtag *AliceInWonderland*
- Q4. le nom des followers de *@Alice*
- Q5. le nom des utilisateurs suivis par *@Alice*
- Q6. le nom des utilisateurs ayant plus de 10 followers.
- Q7. les utilisateurs étant à la fois followers et followees de *@Alice* (c'est à dire qui suivent et sont suivis par *@Alice*)
- Q8. les 10 tweets les plus populaires (ici la popularité se mesure au nombre de fois où le tweet est favori)
- Q9. les 10 hashtags les plus populaires (ici, la popularité se mesure au nombre de tweets dans lesquels ils apparaissent)

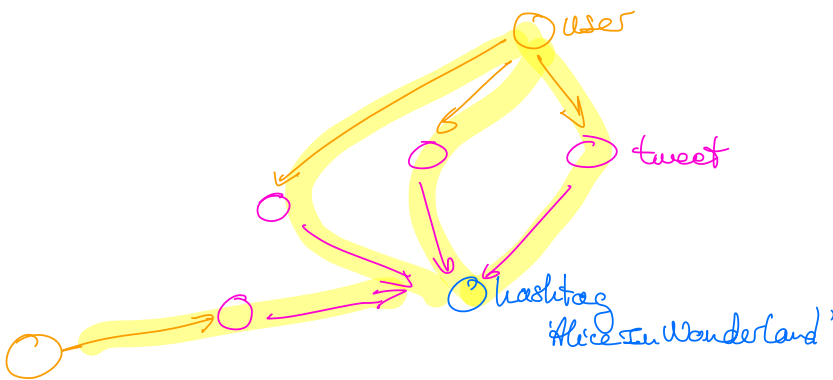
Q1. match (u: user) return count(u)

Q2. match (t: tweet) -[:contains] -> (h: hashtag  
{ hashtag: 'WhiteRabbit' })  
 return count(t)

⇔ match (t: tweet) -[:contains] -> (h: hashtag)  
 where h.hashtag = 'WhiteRabbit'.  
 return count(t)

Q3. match (u: user) -[:tweets] -> (t: tweet) -[:contains] ->  
(h: hashtag)  
 where h.hashtag = 'AliceInWonderland'  
 return count(distinct(u))

↳ chemins



[Q4]. match (u: user) - [: follows] → (u2: user {screenName: 'Alice'})  
return u.screenName

[Q5]. match (u: user {screenName: 'Alice'}) - [: follows] →  
(u2: user)  
return u2.screenName

[Q6]. match (u: user) - [: follows] → (u2: user)  
with u2, count(u)  
where count(u) > 10  
return u2.name

ou (plus joli)

match (u: user) - [: follows] → (u2: user)  
with u2, count(u) as nbFollowers  
where nbFollowers > 10  
return u2.name

[Q7]. match (u: User) - [: follows] → (u2: user {screenName: 'Alice'})  
- [: follows] → (u: user)  
return u

[Q8]. match (t: tweet)  
return t  
order by t.nbFavorites desc  
limit 10

Q9.

match (t:tweet) -[:contains] => (h:hashtag)

with h, count(t)

return h

order by count(t) desc

limit 10

= les 10  
hashtags qu'on  
trouve dans  
le plus de  
tweets

ou

match (t:tweet) -[:contains] => (h:hashtag)

with h, count(t) as nbTweets

return h

order by nbTweets desc

limit 10

Q10. le nom des followers des followers des followers des followers de @Alice (followers au 4ème degré).

Q11. le nom des followers au dixième degré de @Alice

Q10.

match (u:user) -[:follows] -> (:user) -[:follows]

-> (:user . . . . . => trop long !

match (u:user) -[:follows \*4] -> (u1:user {  
screenName:'Alice'})  
return u

Q11.

match (u:user) -[:follows \*10] -> (u1:user  
{screenName:'Alice'})  
return u