

Cypher Query for clustering nodes based on relationships and specific properties

2 posts by 2 authors 



Thomas Bruckmayer

Mar 29



Hi,

I need to cluster/sort a list of nodes based on the following criteria:

1. Start with the node based on a specific property condition (e.g. $n1$ where $n1.priority = \max(priority)$)
2. Get all related nodes for $n1$ (e.g. $n1-n1.1$, $n1-n1.2$)
3. Sort related nodes based on a specific property (e.g. order by $n.priority$)
4. Continue with the next node

Sample query result:

1. $N1.priority=100$
2. $N1.1.priority=60$
3. $N1.2.priority=40$
4. $N2.priority=80$
5. $N2.1.priority=70$ (although $N2.1$ has a higher prio than $N1.1$, $n1.1$ is listed before $n2.1$ because $n1.1$ has a relationship to $N1$)
6. $N2.2.priority=50$
7. $N3.2.priority=45$

My first intention was to use a programming control statement approach (e.g. for loops and recursion) but Neo4j should be suitable to deliver an answer.

Thanks for any hints creating the cypher statement.

Best regards



Michael Hunger

Apr 4



As this is an iterative algorithm it won't be pretty, trivial or fast with plain Cypher.

How big is your graph?

According to your description it is actually this

```
MATCH (n1:Label)-->(n2:Label)
// WHERE n1.priority = {priority}
WITH n1, n2
ORDER BY n1.priority DESC, n2.priority DESC
```

Am 29.03.2015 um 19:25 schrieb Thomas Bruckmayer <bruck...@gmail.com>:

Hi,

I need to cluster/sort a list of nodes based on the following criteria:

1. Start with the node based on a specific property condition (e.g. $n1$ where $n1.priority = \max(priority)$)

2. Get all related nodes for n1 (e.g. n1—n1.1, n1—n1.2)
3. Sort related nodes based on a specific property (e.g. order by n.priority)
4. Continue with the next node

what is "next" node for you?

Sample query result:

1. N1.priority=100
2. N1.1.priority=60
3. N1.2.priority=40
4. N2.priority=80
5. N2.1.priority=70 (although N2.1 has a higher prio than N1.1, n1.1 is listed before n2.1 because n1.1 has a relationship to N1)
6. N2.2.priority=50
7. N3.2.priority=45

My first intention was to use a programming control statement approach (e.g. for loops and recursion) but Neo4j should be suitable to deliver an answer.

Thanks for any hints creating the cypher statement.

Best regards

--

You received this message because you are subscribed to the Google Groups "Neo4j" group. To unsubscribe from this group and stop receiving emails from it, send an email to [neo4j+un...@googlegroups.com](mailto:neo4j+unsubscribe@googlegroups.com).

For more options, visit <https://groups.google.com/d/optout>.
