stackoverfloooooooow

Questions | Tags | Users | Badges | Unanswered | Ask Question

cypher indexing common property on all nodes

asked   4 months ago
viewed   38 times
active   4 months ago

▲
0
▼
★

As per new indexing rules, the `auto_index` will go away in future and its expected to create indexes using `cypher`. According to this new way, to index a node property, you MUST provide a Node Label.

I have a 'nodeId' property present on all types of Node Labels - User, Employee, Bank, Car, etc. I used to auto-index this property to retrieve any type of node if its nodeId is known. Please note that since auto-index did not require me to give a Node Label, it was possible for me to do what I did.

```
ReadableIndex<Node> readableIndex = this.graphDatabaseService.index().getNodeAutoInd
readableIndex.get("nodeId", "0").getSingle();
```

But with new style, I have to create index on nodeId property for each and every Node Label. So I have to do this:

```
create index on :User(nodeId)
create index on :Employee(nodeId)
...
```

Moreover, my method `getByNodeId(String nodeId)` is useless now because this cypher query IMHO will not be able to use the index anymore since I am not passing any node label.

```
match (node) where node.nodeId = {nodeId} return node;
```

Since the whole point of my `getByNodeId()` method was to be generic across all nodes, I cannot give this cypher query a node label. So what should I do here. My 2 questions are:

- How do I tell neo4j via cypher to index on all node labels
- How do I write a cypher query which uses index not based on node label, but based on node property.

Note:

- It is essential for me to use cypher because I am using neo4j-jdbc and they have no method to create auto-index or access the auto-indexer (atleast not that I know of).

- Some might suggest me to change the neo4j.properties to enable auto-indexing there, but I dont like changing configuration files. I want to do it in my program. Anyway, that would have only solved the first issue. Second issue is still there.

`java` `neo4j` `cypher`

share improve this question

asked Mar 30 at 18:40
👤 Rash
818 ●1 ●6 ●17

add a comment

Related

1 Neo4j: fulltext indices and auto indexing in Cypher

1 Create relationship between nodes having same property value in common, using one Cypher query

1 Get all nodes with given type by (select * from tablename in SQL world) with Cypher Neo4j

0 Cypher - aggregating nodes by string in a property

0 how to remove node itself in NOT logical cypher query?

0 Fetching nodes with a label and index using cypher

1 How to return all properties of a node with their name and their value using Cypher

0 Cypher: Create relationships between nodes based on a common property key id

1 Cypher query for all nodes that lack a certain property

0 deleteing a node cypher java

Hot Network Questions

1 Answer

active   oldest   **votes**

▲
1
▼

A node can have multiple labels.

Thus, if you make all your nodes share a common label, say `Base` (in addition to whatever labels they currently have), you can just have a single index that covers all your nodes:

```
CREATE INDEX ON :Base(nodeId)
```

```
CREATE INDEX ON :Base(nodeId)
```

share improve this answer

answered Mar 30 at 20:03

cybersam
9,941 ● 2 ● 7 ● 21

This is definitely a good answer. However is there really no other way? Here is why - I have three different database flavors - neo4j embedded and rest, and titan. I want all of them to have the same graph structure because it helps me keep stuff simple. I would not like to deviate from the consistency of 3 databases until I really really have to. Though your answer makes a lot of sense, having more than one label is something Titan does not support. Hence I would only use it if there is no other way. Hope this makes sense. :) – Rash Mar 31 at 2:44 ✏

You can omit the shared label in the non-neo4j DBs. Aside from that, the structure should be the same. – cybersam Mar 31 at 5:51

add a comment

Your Answer

B	*I*	🔗 66 {} 🖼	≡ ≡ ≡ ≡	↶ ↷

Post Your Answer

By posting your answer, you agree to the privacy policy and terms of service.

Not the answer you're looking for? Browse other questions tagged java neo4j cypher or ask your own question.

question feed

Information Security   more (13)

Money   Role-playing Games   more (5)

Academia   more (21)

more (9)