



This repository Search

Explore Features Enterprise Blog

Sign up

Sign in

neo4j / neo4j

Watch 238

Star 1,885

Fork 730

# Fixes underterministic collision detection ordering in IdMapper

Merged tinwelint merged 1 commit into neo4j:2.2 from tinwelint:2.2-idmapper-dupdet-fix Apr 17, 2015

Conversation 4 Commits 1 Files changed 2

+85 -18



tinwelint commented Apr 17, 2015

Collaborator

since the sorting of the collision data didn't take into account each items position in dataCache. Now it does.

tinwelint added kernel 2.2 labels Apr 17, 2015

Fixes underterministic collision detection ordering in IdMapper f127e9b

davidegrohmann commented on the diff Apr 17, 2015

...chimport/cache/idmapping/string/EncodingIdMapper.java

View full changes

```
((23 lines not shown))
472 +     }
473 +     if ( leftEId == pivotEId )
474 +     {
475 +         return left < pivot;
476 +     }
477 +     return false;
478 + }
479 +
480 + @Override
481 + public boolean ge( long right, long pivot )
482 + {
483 +     long rightEId = dataCache.get( right );
484 +     long pivotEId = dataCache.get( pivot );
485 +     if ( DEFAULT.ge( rightEId, pivotEId ) )
486 +     {
487 +         return rightEId == pivotEId ? right > pivot : true;

```

davidegrohmann added a note Apr 17, 2015



Collaborator

right >= pivot ?



tinwelint added a note Apr 17, 2015

Collaborator

correct, although there's no chance that they're ever the same actually. They just can't be. It should be an assertion if anything.

Add a line note



davidegrohmann commented Apr 17, 2015

Collaborator

@tinwelint It would be nice to have a test for the original problem that this PR is aiming to fix.



tinwelint commented Apr 17, 2015

Collaborator

There is a test which failed at least 50% of the times before. And the issue (ordering) is random since

Labels

2.2

kernel

Milestone

No milestone

Assignee

No one assigned

2 participants



ParallelSort pivot selection is random in the qsort algorithm, so it'd be quite hard to test unless there were refactorings in these areas to expose this functionality in isolation so that it could be tested. I don't think it's a good time to do that since we're dealing with a flaky test. But going forward we could refactor in small steps. It's slightly scary since this code is quite complicated.



**tinwelint** merged commit **8eaf4e0** into **neo4j:2.2** Apr 17, 2015  
1 check passed

[View details](#)

[Sign up for free](#)

to join this conversation on **GitHub**. Already have an account? [Sign in to comment](#)

