

Re: SQL Server – Very long query time with OpenJPA and Hibernate

Source: <http://coding.derkeiler.com/Archive/Java/comp.lang.java.databases/2007-12/msg00001.html>

- *From:* Silvio Bierman <sbierman@xxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Mon, 03 Dec 2007 13:18:24 +0100
-

rico.fabrini@xxxxxxxxx wrote:

Hello Everyone,

I am hoping somebody could point me in the right direction..

I have a table with close to a million entries. I've set up indexes that are relevant to the expected queries. Thanks to the indexes I can use Enterprise Manager and Query Analyzer to obtain results within say 3 seconds.

Those results are for when manually running queries generated by OpenJPA and also by Hibernate. However, when running the program itself, that uses those frameworks, the same query can take up to 30seconds before finally, and correctly, returning even an empty collection. That's on a production server, in instances when I have even made sure that the program is the only thing accessing the database server.

I don't observe such a long delay when executing the program on a restored copy of the database, and the table with ~million entries, on my own development PC. The query time is a matter of milliseconds.

I've copied all the data over to a PostgreSQL database, and here, on both production server and development workstation, the execution time is also a matter of milliseconds. I've used virtually identical META-INF/persistence.xml for the respective frameworks on the respective DBMS, and respective workstation/server.

I'm not clear where else to look for a culprit(s) here. I'm hoping some people could help with

Re: SQL Server – Very long query time with OpenJPA and Hibernate

some advice. Thanks.

Rico.

Hello Rico,

Unrealistically slow response times often stem from one of:

- 1) DNS resolution delays (could be both client and server side)
- 2) Encoding mismatch between query and indexes (Unicode versus plain, note that the MS JDBC driver defaults to Unicode) leading to table scans.

If I had a choice I would go with PG anyway...

Kind regards,

Silvio Bierman

.