

Database - Bug #8044

Fix "Failed direct-access due to Error direct accessing for finding record by unique index" messages

11/14/2023 05:03 AM - Alexandru Lungu

Status:	Test	Start date:	
Priority:	Normal	Due date:	
Assignee:	Dănuț Filimon	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:		case_num:	
billable:	No		
vendor_id:	GCD		
Description			

History

#1 - 11/15/2023 09:12 AM - Alexandru Lungu

- Assignee changed from Alexandru Lungu to Dănuț Filimon
- Status changed from New to WIP

#2 - 11/15/2023 09:21 AM - Dănuț Filimon

I have multiple warning where this direct access exception is logged in a customer application: 23/11/15 16:08:27.398+0200 | WARNING | com.goldencode.p2j.persist.RandomAccessQuery | ThreadName:Conversation [0000000B:<Process>], Session:00000011, Thread:00000003, User:<Process> | Failed direct-access due to Can't normalize FWD value to use it with direct access.. After investigating, I found out that DirectAccessHelper.normalizeValue receives a rowid object as an argument and it is not handled properly.

#3 - 11/15/2023 09:41 AM - Dănuț Filimon

I am still getting the same error but in a different process now, there is another object aside from rowid that is not handled properly. Investigating...

#4 - 11/16/2023 03:53 AM - Dănuț Filimon

Committed 8044a/rev.14829 Normalized rowid and longchar arguments.

Alexandru, can you provide a reproduction case for the error you obtained? I have an example where I do a FIND FIRST on a temp-table and use a field that is part of a unique index, but I don't get any error. This error should not be caused by the problem mentioned above, it should be something else that I have to look into.

#5 - 11/16/2023 04:21 AM - Alexandru Lungu

- % Done changed from 0 to 50

Not quite; I was just seeing this pop up thousand of times.

Please also mind handling:

- the unknown value as well (beside rowid and longchar). It should simply normalize to null.
- date and date-time. This requires extensive testing.

The error is caused when you do direct-access (CAN-FIND (tt WHERE tt.f1 = 1) and you have an unique index on f1). this won't throw any error as this normalizes and int (1). Attempt to do the same test with rowid, longchar, unknown, date and date-time.

Please do smoke tests on a large application and ensure this warning is not shown anymore.

#6 - 11/16/2023 04:32 AM - Dănuț Filimon

- % Done changed from 50 to 0

Alexandru Lungu wrote:

Not quite; I was just seeing this pop up thousand of times.

Please also mind handling:

- the unknown value as well (beside rowid and longchar). It should simply normalize to null.
- date and date-time. This requires extensive testing.

The error is caused when you do direct-access (CAN-FIND (tt WHERE tt.f1 = 1) and you have an unique index on f1). this won't throw any error as this normalizes and int (1). Attempt to do the same test with rowid, longchar, unknown, date and date-time.

Please do smoke tests on a large application and ensure this warning is not shown anymore.

I've experimented with date, but I was not sure if I should return dateValue(). I will go ahead and test it properly.

#7 - 11/16/2023 04:32 AM - Dănuț Filimon

- % Done changed from 0 to 50

#8 - 11/16/2023 04:34 AM - Alexandru Lungu

Please check the DateType.setParameter function. It should do the same.

#9 - 11/16/2023 06:04 AM - Dănuț Filimon

While I was testing a large application, there were not a lot of instances where the unknown value is normalized to null and none where longchar/rowid/date/datetime were normalized. When I tested yesterday, most of the direct access exceptions were thrown by a process and it was because of rowid, but now the same process is not starting. In fact, no processes are starting. Until this problems are solved, I will resume to using specific test cases.

Alexandru Lungu wrote:

Please check the DateType.setParameter function. It should do the same.

This is not the solution for date, there is no PreparedStatement available and the cast is done to java.sql.Date.

#10 - 11/17/2023 04:05 AM - Dănuț Filimon

- Status changed from WIP to Review

- % Done changed from 50 to 100

Committed 8044a/rev.14830 Normalized unknown value, date and datetime arguments. In the end, I used `dateValue()` to return a Date instance, this instance is then converted to a ValueDate in H2 using `DataType.convertToValue` which I think you actually wanted to mention.

Alexandru, please review.

#11 - 12/05/2023 03:33 AM - Alexandru Lungu

- Status changed from Review to Internal Test

Review of 8044a:

- I am OK with the changes.

As long as you've done smoke tests with a large application and independently test each of the new data types supported (date, datetime, rowid, longchar and unknown), I am ready to merge this.

Greg, this branch should clear the Failed direct-access due to Can't normalize FWD value to use it with direct access. messages we are still seeing in some applications. Let me know if I can merge it (I can't move it to Merge Pending - I guess it is private).

#12 - 12/05/2023 09:42 AM - Greg Shah

- Status changed from Internal Test to Merge Pending

You can merge now.

#13 - 12/05/2023 09:55 AM - Alexandru Lungu

- Status changed from Merge Pending to Test

Branch 8044a was merged into trunk as rev. 14860 and archived.

#14 - 03/14/2024 03:48 PM - Tomasz Domin

Just to notify - I am still getting a lot of "Failed direct-access due to Error direct accessing for finding record by unique index" in a code based on trunk/15051 while running regression tests. Not sure what is the core problem behind. Please see #6667-898