

Database - Bug #8496

investigate if a 'batch insert with deactivated index' in H2 is possible

03/21/2024 02:43 PM - Constantin Asofiei

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
billable:	No	case_num:	
vendor_id:	GCD	version:	
Description			

History

#2 - 03/21/2024 02:52 PM - Constantin Asofiei

There are operations where we do lots of inserts. In some case, in a FILL operation, some 50k temp-table records are inserted (and indexes updated on each insert, with ~3.5 million 'compareRows' calls).

The idea is to have a mode in H2 where indexes are not updated(?) after each insert. Possible problems:

- unique constraint violations - how do we check for this?
- lookups needed in MERGE mode, to find any existing records

A point would be to i.e. insert the record in the TreeIndex, but do not re-balance it; rebalance would be done when i.e. a SELECT is done on that index (we see the index as 'dirty' and force re-balance); and also when a 'ANALYZE' on that table is called.

But, with the [#7323](#) changes, the unique constraints are checked at INSERT, and not via SELECT.

#3 - 03/21/2024 02:55 PM - Constantin Asofiei

Above 're-balance' actually means re-building the index... so this may be expensive on its own.

#4 - 03/22/2024 07:44 AM - Alexandru Lungu

Constantin, for FILL we do 50k independent inserts. Can we maybe bulk them? Or do we actually need the intermediate states?

My thinking was to gather the inserts without actually inserting and emit one single INSERT in the end with all values. I don't know if H2 is actually performing better in regard to index look-up, but we can experiment.

#5 - 03/22/2024 09:37 AM - Ovidiu Maxiniuc

Constantin Asofiei wrote:

The idea is to have a mode in H2 where indexes are not **updated(?) after each insert**. Possible problems:

- unique constraint violations - how do we check for this?
- lookups needed in MERGE mode, to find any existing records

The bold text above triggered an idea in my head. I think 4GL does a nasty thing: they flush/write the record to database without checking/updating the indices. That is, delaying this operation until a specific time (records are validated or the transaction block ends, for example). The new records are probably marked as as final until that happens. They can do this because do the low-level management. This way they can quickly write multiple records, but validate only once. In fact, this is what we do this when importing the database, but with a larger data set (the whole database).

OTOH, FWD must obey the constraints of DBMS/SQL. We cannot flush without proper validation or the SQL will reject the new data. An idea is to fully disable the SQL unique indices for permanent tables as we do for temp-tables and rely only on in-application validation. This can be rather safely be done for the temp-tables, but doing this for a permanent tables compromises the database's integrity. Again, for OE this might not be an issue because of the proprietary format of the database. Some of our customers if not all of them will not be happy with that. I have to mention that we 'upgraded' the normal indices to semantically unique, but this should not affect this issue.