Cost of Inserts vs Update in SQL Server

Asked 16 years, 3 months ago Modified 10 years, 9 months ago

Viewed 47k times

26

I have a table with more than a millon rows. This table is used to index `tiff` images. Each image has fields like `date`, `number`, etc. I have users that index these images in batches of 500. I need to know if it is better to first insert 500 rows and then perform 500 updates or, when the user finishes indexing, to do the 500 inserts with all the data. A very important thing is that if I do the 500 inserts at first, this time is free for me because I can do it the night before.

So the question is: is it better to do inserts or inserts and updates, and why? I have defined a `id` value for each image, and I also have other indices on the fields.

sql-server **performance**

Share

Improve this question

Follow

edited Mar 9, 2014 at 0:27

alvonellos

1,062 ● 1 ● 9 ● 28

asked Sep 3, 2008 at 14:47

Jedi Master Spooky

5,809 ● 14 ● 59 ● 86

6 Answers

Sorted by: Highest score (default)

40

Updates in Sql server result in ghosted rows - i.e. Sql crosses one row out and puts a new one in. The crossed out row is deleted later.

Both inserts and updates can cause page-splits in this way, they both effectively 'add' data, it's just that updates flag the old stuff out first.

On top of this updates need to look up the row first, which for lots of data can take longer than the update.

Inserts will just about always be quicker, especially if they are either in order or if the underlying table doesn't have a clustered index.

When inserting larger amounts of data into a table look at the current indexes - they can take a while to change and build. Adding values in the middle of an index is always slower.

You can think of it like appending to an address book: Mr Z can just be added to the last page, while you'll have to find space in the middle for Mr M.

Share Improve this answer

Follow

answered Sep 3, 2008 at 15:13

Keith

155k ● 82 ● 306 ● 446

Does that required time increase with the size of the table being indexed? – Nathan Hinchey Apr 3, 2019 at 23:59

@NathanHinchey some of it - obviously finding which record to update takes more work if there is more data, but the write operation and the page split stay constant (as the page sizes are fixed). The more data then the more mid-cluster inserts and updates cost. – Keith Apr 4, 2019 at 7:03

This might complement the answer: learn.microsoft.com/en-us/sql/relational-databases/… – tehmas May 29, 2020 at 8:08

3

This isn't a cut and dry question. Krishna's and Galegian's points are spot on.

For updates, the impact will be lessened if the updates are affecting fixed-length fields. If updating varchar or blob fields, you may add a cost of page splits during update when the new value surpasses the length of the old value.

Share Improve this answer

Follow

answered Sep 3, 2008 at 15:08

spoulson

21.6k ● 16 ● 78 ● 102

2

Doing the inserts first and then the updates does seem to be a better idea for several reasons. You will be inserting at a time of low transaction volume. Since inserts have more data, this is a better time to do it.

Since you are using an id value (which is presumably indexed) for updates, the overhead of updates will be very low. You would also have less data during your updates.

You could also turn off transactions at the batch (500 inserts/updates) level and use it for each individual record, thus reducing some overhead.

Finally, test this out to see the actual performance on your server before making a final decision.

Share Improve this answer

Follow

answered Sep 3, 2008 at 15:00

Krishna Kumar

4,864 ● 13 ● 47 ● 46

2

I think inserts will run faster. They do not require a lookup (when you do an update you are basically doing the equivalent of a select with the where clause). And also, an insert won't lock the rows the way an update will, so it won't interfere with any selects that are happening against the table at the same time.

Share Improve this answer

Follow

answered Sep 3, 2008 at 15:12

Eric Z Beard

38.4k ● 27 ● 101 ● 147

1

The execution plan for each query will tell you which one should be more expensive. The real limiting factor will be the writes to disk, so you may need to run some tests

while running perfmon to see which query causes more writes and causes the disk queue to get the longest (longer is bad).

Share Improve this answer

Follow

answered Sep 3, 2008 at 15:03

Booji Boy

4,582 ● 4 ● 41 ● 45

0

I'm not a database guy, but I imagine doing the inserts in one shot would be faster because the updates require a lookup whereas the inserts do not.

Share Improve this answer

Follow

answered Sep 3, 2008 at 14:54

Giovanni Galbo

13.1k ● 13 ● 61 ● 79

Giovanni, it will also depend on other issues such as indexing (clustered or non-clustered) and fill factor. Your specific situation will contribute largely on how you proceed.
– Galwegian Sep 3, 2008 at 14:59