



15V

Lektion 4

Mandag den 5. september 2016

Litteratur: [Ramakrishnan] kapitel 10 til og med 10.7 (lad være med at gå dybt ned i algoritmerne i 10.4, 10.5 og 10.6)

Emner:

Gennemgang af opgave 2.1 (forventes lavet hjemme)

Diske

- Hvordan arbejder de
- Meget lidt om RAID

Træbaserede index

- ISAM
- B+

Sammenligning af hash- og træbaserede index.

Index i SQL-server

- Hvordan er de implementeret
- Lidt flere kommandoer
- Kort eksempel fra script

Stored procedures – en ultrakort repetition

Kort oplæg til relationel algebra (hvis vi kan nå det)

Opgaver:

Opgave 4.1, 4.2, 2.2 og 4.3

Læsning til næste gang:

[Ramakrishnan] kapitel 4, siderne 100-107 (fokuser på operationerne SELECT, PROJECT og JOIN)
[Ramakrishnan] kapitel 12

Bemærkninger:



Exercise 4.1

This exercise is a continuation of exercise 3.1. We use the same table big and we assume you have the 40000 records in it. If you have created any indexes last time drop them again (if you can't remember the indexes look in the management tool).

Run *set statistics io on*

Exercise A

Make a simple SELECT on a name (for example `Andersen20113`)

Make a simple SELECT on an id (for example `20113`)

For both queries note the number of logical io's.

Exercise B

Make a nonclustered index on the attribute id.

Make a simple SELECT on a name (for example `Andersen20113`)

Make a simple SELECT on an id (for example `20113`)

For both queries note the number of logical io's.

Drop the nonclustered index again

Exercise C

Make a clustered index on the attribute name with a fillfactor of 50 and PAD_index.

Make a nonclustered index on the attribute id.

Make a simple SELECT on a name (for example `Andersen20113`)

Make a simple SELECT on an id (for example `20113`)

Make a simple SELECT on a filler (for example `bøf`)

For all queries note the number of logical io's.

Exercise D

Can you explain the results for the logical io's



Exercise 4.2

In the end of exercise 3.1 from last we calculated the number of records you will expect in one page.

You can make a corresponding calculation for the index-blocks. The complete formular is a bit complicated, because it depends on a lot of things. For the clustered index on name the formel is that a record-index contains a 10 bytes header/disc address + the search-attribute.

If we make a clustered index on name in the table big – how many records can we (as a maximum) have if the depth of tree (including data) is 2, if it is 3, if it is 4.

Exercise 4.3

This exercise is a programming exercise in T-SQL. In the script timereg you can find worktime registrations for a number of employees. The script contains the table definition and the test data.

Exercise A

Find the total amount of worktime for each employee each week.

The normal worktime is 37 hours each week. Make a stored procedure to find all employees, who have worked more than that in a given week (the weeknumber is a parameter to the procedure)

Exercise B

Make a stored procedure that given a datetime in a parameter returns how many employees, which were at the company at that time. The return should be in an output parameter.

Exercise C

There is a rule that no employee should meet earlier than 11 hours after he/she left the company last time.

Make a stored procedure to check if a given employee (a parameter to the stored procedure) has broken the rule.