

TUGAS BESAR
TUNING DATABASE SYSTEM
MATAKULIAH MANAJEMEN BASIS DATA

Disusun oleh :

Yose Alloisius Saragih /14117092



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI PRODUKSI, INDUSTRI DAN INFORMASI
INSTITUT TEKNOLOGI SUMATERA

2019

DAFTAR ISI

BAB I	3
STUDI LITERATUR	3
1.1 Tunning: Indexing.....	3
1.2 Tunning: Setting Configuration DBMS	3
BAB II	4
DESKRIPSI PERCOBAAN	4
2.1 Tunning : Index.....	4
BAB III	6
HASIL DAN PEMBAHASAN.....	6
3.1 Tabel Hasil	6
3.2 Pembahasan.....	6
DAFTAR PUSTAKA	7

BAB I

STUDI LITERATUR

1.1 Tunning: Indexing

Indexing merupakan sebuah objek pada MySQL yang berisi data yang terurut dari nilai-nilai pada satu atau lebih baris dalam suatu tabel yang ada pada database. Jika database dibuat tanpa menggunakan index, maka performa DBMS dapat menurun, terlebih lagi apabila data dalam database tergolong besar. Hal ini disebabkan karena pada saat dijalankan, DBMS akan melakukan *full table scan* yang mana dapat memakan waktu yang cukup lama.

Index merupakan sebuah struktur data yang menyimpan nilai spesifik sebuah kolom pada sebuah table yang berfungsi sebagai penunjuk ke baris yang berisi nilai tersebut. Penunjuk inilah yang secara langsung ke baris yang diinginkan pada tabel, sehingga akan menghindari terjadinya *full table scan*. Tetapi banyaknya index tidak menjamin kecepatan pemrosesan suatu query karena semakin banyak index dapat juga memperlambat pemrosesan perintah-perintah DML (Data Manipulation Language).

1.2 Tunning: Setting Configuration DBMS

Database server merupakan sebuah program komputer yang menyediakan layanan pengelolaan basis data dan melayani komputer atau program aplikasi basis data yang menggunakan model klien/server. Database server didefinisikan sebagai client server model. Database management system menyediakan fungsi database server dan beberapa DBMS sangat eksklusif untuk konfigurasi database service yang kita gunakan.

BAB II

DESKRIPSI PERCOBAAN

2.1 Tuning : Index

Lakukan indexing pada field dari table student

```
MariaDB [dbms1]> create index student_pk on student(name);  
Query OK, 0 rows affected (1.25 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

Gambar 1 - Index pada dbms1

```
MariaDB [dbms2]> create index student_pk on student(name);  
Query OK, 0 rows affected (0.58 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

Gambar 2 - Index pada dbms2

```
MariaDB [dbms3]> create index student_pk on student(name);  
Query OK, 0 rows affected (0.40 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

Gambar 3 - Index pada dbms3

Query :

1. SELECT * FROM student
2. SELECT * FROM student WHERE tot_cred > 30;
3. SELECT 'name', dept_name FROM student WHERE tot_cred > 30;
4. SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section
ON takes.course_id = section.course_id
5. SELECT student.`name`,student.dept_name,takes.sec_id AS
pengambilan,takes.semester,section.room_number,section.building,course.course
_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN
section ON takes.course_id = section.course_id JOIN course ON
section.course_id = course.course_id

Eksekusi query sebelum dan sesudah melakukan pengindexan

```
| 29 | 0.10855860 | SELECT*FROM student
|
| 30 | 0.02046910 | SELECT * FROM student WHERE tot_cred > 30
|
| 31 | 0.00084300 | SELECT 'name', dept_name FROM student WHERE tot_cred > 30
|
| 32 | 0.11842030 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
|
| 33 | 0.06435270 | SELECT student.'name',student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,cours
e.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou
```

Gambar 4 - dbms1 sebelum indexing

```
| 35 | 0.00104540 | SELECT*FROM student
|
| 36 | 0.00082550 | SELECT * FROM student WHERE tot_cred > 30
|
| 37 | 0.00051800 | SELECT 'name', dept_name FROM student WHERE tot_cred > 30
|
| 38 | 0.00531950 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
|
| 39 | 0.00234980 | SELECT student.'name',student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,cours
e.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou
```

Gambar 5 - dbms1 setelah indexing

```
| 43 | 0.01034880 | SELECT*FROM student
|
| 44 | 0.00061560 | SELECT * FROM student WHERE tot_cred > 30
|
| 45 | 0.17877710 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
|
| 46 | 0.00124990 | SELECT 'name', dept_name FROM student WHERE tot_cred > 30
|
| 47 | 0.15321610 | SELECT student.'name',student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,cours
e.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou
```

Gambar 6 - dbms2 sebelum indexing

```
| 50 | 0.00089110 | SELECT*FROM student
|
| 51 | 0.00084030 | SELECT * FROM student WHERE tot_cred > 30
|
| 52 | 0.00062990 | SELECT 'name', dept_name FROM student WHERE tot_cred > 30
|
| 53 | 0.00580480 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
|
| 54 | 0.00841090 | SELECT student.'name',student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,cours
e.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou
```

Gambar 7 - dbms2 setelah indexing

```
| 58 | 0.04050390 | SELECT*FROM student
|
| 59 | 0.00095940 | SELECT * FROM student WHERE tot_cred > 30
|
| 60 | 0.00102690 | SELECT 'name', dept_name FROM student WHERE tot_cred > 30
|
| 61 | 0.17717560 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
|
| 62 | 0.23507510 | SELECT student.'name',student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,cours
e.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou
```

Gambar 8 - dbms3 sebelum indexing

```
| 64 | 0.00109550 | SELECT*FROM student
|
| 65 | 0.00093610 | SELECT * FROM student WHERE tot_cred > 30
|
| 66 | 0.00141310 | SELECT 'name', dept_name FROM student WHERE tot_cred > 30
|
| 67 | 0.02826910 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
|
| 68 | 0.01854680 | SELECT student.'name',student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,cours
e.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou
```

Gambar 9 - dbms3 setelah indexing

BAB III

HASIL DAN PEMBAHASAN

3.1 Tabel Hasil

Data	Waktu Sebelum Tunning (ms)					Waktu Setelah Tunning (ms)				
	Q1	Q2	Q3	Q4	Q5	Q1	Q2	Q3	Q4	Q5
Advisor =100, Student =100, Section =200, Takes= 200	0.10	0.02	0.00	0.11	0.66	0.00	0.00	0.00	0.00	0.00
Advisor =200, Student =200, Section =400, Takes= 400	0.01	0.00	0.17	0.00	0.15	0.00	0.00	0.00	0.00	0.00
Advisor =500, Student =500, Section =1000, Takes= 1000	0.04	0.00	0.00	0.17	0.23	0.00	0.00	0.00	0.02	0.01

3.2 Pembahasan

Tunning database merupakan aktifitas yang bertujuan untuk memperbaiki atau meningkatkan performa dari sebuah database. Tunning database memiliki beberapa aspek pendekatan seperti hardware, sistem operasi, database management system, schema basis data, indexing, dan query.

Dari hasil percobaan yang telah dilakukan, didapatkan bahwa tuning dengan sistem indexing pada database menghasilkan waktu untuk eksekusi query yang lebih singkat. Hal ini sesuai dengan tujuan dari tuning database dimana indexing berperan dalam meningkatkan performa dari DBMS yang digunakan.

DAFTAR PUSTAKA

<https://pojokprogrammer.net/content/performance-tuning-sederhana-di-mysql-menggunakan-index>

<https://www.i-3.co.id/2016/10/07/index-pada-database/>