

Laporan Tugas Besar
Manajemen Basis Data RC



Disusun Oleh :

Yosua Tan Siswanto / 14117060

Program Studi Teknik Informatika
Jurusan Teknologi Produksi, Industri dan Informasi
Institut Teknologi Sumatera
2019

Daftar Isi

Bab 1 Studi Literatur	3
1.1 Tuning : Indexing.....	3
1.2 Tuning : Setting Configuration DBMS	3
Bab 2 Deskripsi Percobaan.....	4
2.1 Tuning : Indexing	4
2.1 Tuning: Setting Configuration DBMS.....	4
Bab 3 Hasil Percobaan.....	5
Bab 4 Daftar Pustaka dan Lampiran	6
4.1 Daftar Pustaka.....	6
4.2 Lampiran.....	6

Bab 1

Studi Literatur

1.1 Tuning : Indexing

Index Tuning adalah bagian penyetelan basis data untuk memilih dan membuat indeks. Tujuan penyetelan indeks adalah untuk mengurangi waktu pemrosesan kueri. Potensi penggunaan indeks di lingkungan dinamis dengan beberapa permintaan ad-hoc sebelumnya adalah tugas yang sulit. Penyetelan indeks melibatkan kueri berdasarkan indeks dan indeks dibuat secara otomatis saat itu juga. Tidak diperlukan tindakan eksplisit oleh pengguna basis data untuk penyetelan indeks.

1.2 Tuning : Setting Configuration DBMS

Tuning DBMS mengacu pada konfigurasi DBMS dan konfigurasi memori serta sumber daya pemrosesan komputer yang menjalankan DBMS. Ini biasanya dilakukan dengan mengkonfigurasi DBMS, tetapi sumber daya yang terlibat dibagikan dengan sistem host.

Tuning DBMS dapat melibatkan pengaturan interval pemulihan (waktu yang diperlukan untuk mengembalikan keadaan data ke titik waktu tertentu), menetapkan paralelisme (pemecahan pekerjaan dari satu permintaan tunggal menjadi tugas yang ditugaskan ke sumber daya pemrosesan yang berbeda), dan protokol jaringan digunakan untuk berkomunikasi dengan konsumen basis data.

Memori dialokasikan untuk data, rencana eksekusi, cache prosedur, dan ruang kerja [klarifikasi]. Jauh lebih cepat untuk mengakses data dalam memori daripada data pada penyimpanan, jadi menjaga cache data yang cukup besar membuat aktivitas berjalan lebih cepat. Pertimbangan yang sama diberikan pada ruang kerja. Caching rencana dan prosedur eksekusi berarti mereka digunakan kembali alih-alih dikompilasi ulang saat dibutuhkan. Penting untuk mengambil memori sebanyak mungkin, sambil menyisakan cukup untuk proses lain dan OS untuk digunakan tanpa paging memori yang berlebihan untuk penyimpanan.

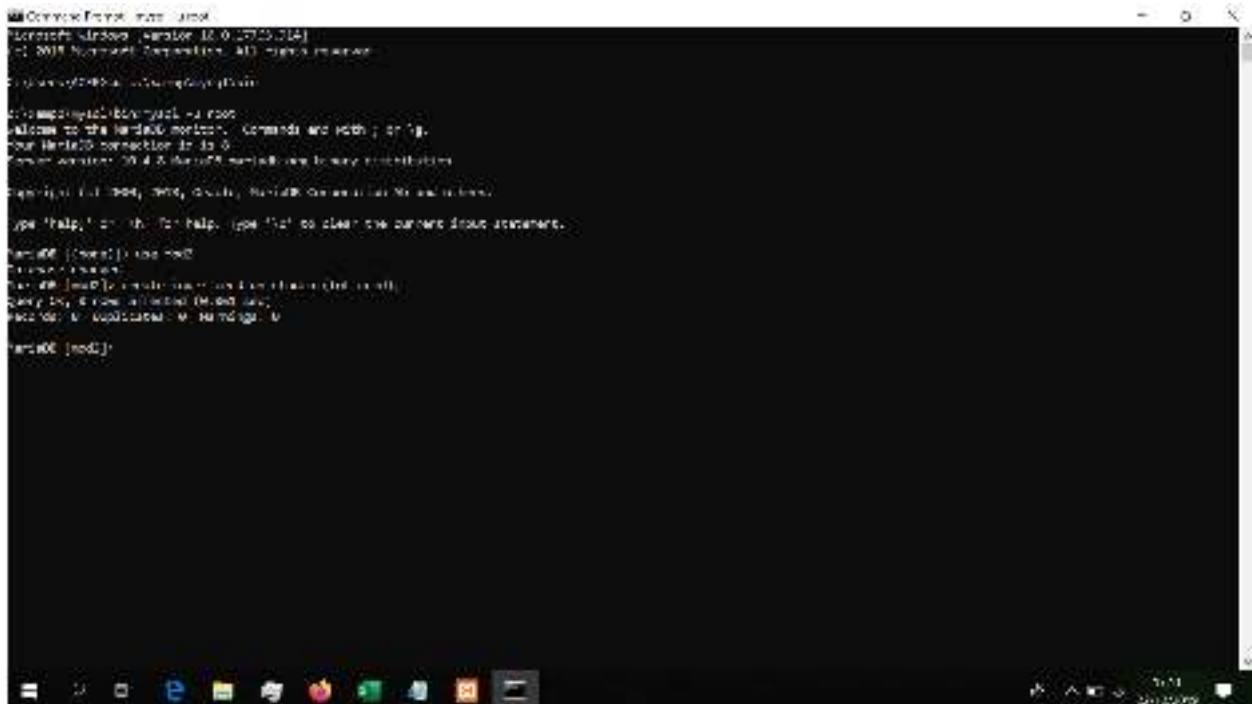
Sumber daya pemrosesan kadang-kadang ditugaskan untuk kegiatan tertentu untuk meningkatkan konkurensi. Pada server dengan delapan prosesor, enam dapat disediakan untuk DBMS untuk memaksimalkan sumber daya pemrosesan yang tersedia untuk database.

Bab 2

Deskripsi Percobaan

2.1 Tuning : Indexing

Tuning ini dilakukan dengan menambahkan index pada beberapa kolom di tabel



Gambar 1 Tuning Indexing

Tuning di atas hanya mengindexing total credit dari table student di karenakan data yang degenerate sudah ter-index.

2.1 Tuning: Setting Configuration DBMS

Tuning dilakukan dengan mengubah konfigurasi dengan my.ini pada folderxampp.
dengan mengubah beberapa nilai :

- Innodb_buffer_pool_size menjadi 2010M
- Innodb_additional_mem_pool_size menjadi 20M
- Innodb_log_file_size menjadi 128M

Bab 3

Hasil Percobaan

Percobaan ini dilakukan dengan beberapa variasi data dan 5 query.

Query yang di ujikan :

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

Hasil dari percobaan di tampilkan dalam bentuk table sebagai berikut

DATA	Waktu Sebelum Tunning (ms)					Waktu Setelah Tunning (ms)				
	query 1	query 2	query 3	query 4	query 5	query 1	query 2	query 3	query 4	query 5
advisor = 100, student = 100, section = 200, takes = 200	83,3431	2,8499	1,3901	39,7428	41,2214	3,0248	2,5781	2,5072	9,7456	8,8923
advisor = 200, student = 200, section = 400, takes = 400	1,0717	1,0271	1,2833	8,4081	8,1348	1,1027	1,145	0,6269	4,0076	11,1888
advisor = 500, student = 500, section = 1000, takes = 1000	1,5358	1,6622	1,5934	30,6967	20,1528	0,3114	2,3397	2,4888	1,4242	98,7399
advisor = 700, student = 700, section = 20000, takes = 20000	6,5282	2,2899	1,6205	3.671,92	1.771,79	2,6453	2,6467	1,8778	3752,79	1.788

Dari data yang didapat dapat dilihat untuk beberapa query waktu eksekusi nya lebih cepat setelah di konfigurasi dibandingkan dengan sebelum di konfigurasi. Perbedaan terbesar dapat di lihat pada data 1 sedangkan pada data lainnya terdapat penurunan dan kenaikan. Percobaan ini hanya dilakukan sampai data keempat dikarenakan CPU yang dipakai untuk melakukan percobaan tidak mendukung untuk data selanjutnya.

Bab 4

Daftar Pustaka dan Lampiran

4.1 Daftar Pustaka

- https://en.wikipedia.org/wiki/Database_tuning
- <https://www.careerride.com/DB-index-tuning.aspx>

4.2 Lampiran

```

C:\Users\user> sqlcmd -s sqlcmd -u user -P 'Password' -d master -i script.sql

sqlcmd [node]: show profile;

+-----+-----+-----+
| Query ID | Text | Query |
+-----+-----+-----+
| 1 | 1.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 2 | 2.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 3 | 3.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 4 | 4.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 5 | 5.00000000 | SELECT * FROM sys.dm_exec_requests; |
+-----+-----+-----+

sqlcmd [node]:

```

```

C:\Users\user> sqlcmd -s sqlcmd -u user -P 'Password' -d master -i script.sql

sqlcmd [node]: show profile;

+-----+-----+-----+
| Query ID | Text | Query |
+-----+-----+-----+
| 1 | 1.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 2 | 2.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 3 | 3.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 4 | 4.00000000 | SELECT * FROM sys.dm_exec_requests; |
| 5 | 5.00000000 | SELECT * FROM sys.dm_exec_requests; |
+-----+-----+-----+

sqlcmd [node]:

```



```
MySQL [root@localhost ~]# mysql -u root -p
mysql> show profiles;
+-----+
| Query_ID | Duration | Query |
+-----+
| 1 | 2.0000000 | SELECT * FROM student |
| 2 | 0.0000000 | SELECT * FROM student WHERE id=1000 |
| 3 | 0.0000000 | SELECT name, dept_name FROM student WHERE dept_name = 'M' |
| 4 | 0.0000000 | SELECT * FROM takes JOIN student ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
| 5 | 0.0000000 | SELECT * FROM takes JOIN section ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
+-----+
2 rows in set (0.000 sec)

mysql> show profiles;
+-----+
| Query_ID | Duration | Query |
+-----+
| 1 | 2.0000000 | SELECT * FROM student |
| 2 | 0.0000000 | SELECT * FROM student WHERE id=1000 |
| 3 | 0.0000000 | SELECT name, dept_name FROM student WHERE dept_name = 'M' |
| 4 | 0.0000000 | SELECT * FROM takes JOIN student ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
| 5 | 0.0000000 | SELECT * FROM takes JOIN section ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
+-----+
2 rows in set (0.000 sec)
```

```
MySQL [root@localhost ~]# mysql -u root -p
mysql> show profiles;
+-----+
| Query_ID | Duration | Query |
+-----+
| 1 | 2.0000000 | SELECT * FROM student |
| 2 | 0.0000000 | SELECT * FROM student WHERE id=1000 |
| 3 | 0.0000000 | SELECT name, dept_name FROM student WHERE dept_name = 'M' |
| 4 | 0.0000000 | SELECT * FROM takes JOIN student ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
| 5 | 0.0000000 | SELECT * FROM takes JOIN section ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
+-----+
2 rows in set (0.000 sec)

mysql> show profiles;
+-----+
| Query_ID | Duration | Query |
+-----+
| 1 | 2.0000000 | SELECT * FROM student |
| 2 | 0.0000000 | SELECT * FROM student WHERE id=1000 |
| 3 | 0.0000000 | SELECT name, dept_name FROM student WHERE dept_name = 'M' |
| 4 | 0.0000000 | SELECT * FROM takes JOIN student ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
| 5 | 0.0000000 | SELECT * FROM takes JOIN section ON takes.student_id = student.id JOIN section ON takes.section_id = section.section_id |
+-----+
2 rows in set (0.000 sec)
```