

**TUGAS BESAR MANAJEMEN BASIS DATA**  
**“ Tuning Pada DBMS”**



Dosen Pengampuh :  
Ahmad Luky Ramdani, S.Komp., M.kom.

Nama : Pasha Abdul Khalid (14116090)

Mata Kuliah :  
Manajemen Basis Data RA

**TEKNIK INFORMATIKA**  
**INSTITUT TEKNOLOGI SUMATERA**  
**2019**

## DAFTAR ISI

DAFTAR ISI.....	2
BAB I ISI.....	3
A. Studi Literatur .....	3
1.1 Tunning : Indexing .....	3
1.2 Tunning : Setting Configuration DBMS .....	3
BAB II HASIL DAN PEMBAHASAN .....	5
B. Deskripsi Percobaan.....	5
2.1 Tunning : Indexing .....	5
2.2 Tunning : Setting Configuration DBMS .....	7
BAB III KESIMPULAN .....	10
DAFTAR PUSTAKA .....	11

## BAB I ISI

### A. Studi Literatur

#### 1.1 Tuning : Indexing

*Tuning* adalah sebuah metodologi untuk memaksimalkan *throughput* dan meminimalkan *response time* untuk proses-proses seperti transaksi *online*, aktivitas-aktivitas internet dan *batch job*. Dengan kata lain, *Performance and Tuning* adalah sebuah upaya untuk memaksimalkan infrastruktur baik *software* ataupun *hardware* yang ada agar dapat berfungsi dengan optimal. *Performance and Tuning* sebenarnya memiliki dua tugas yang dijalankan berdampingan, *performance* bertujuan untuk memonitor kinerja dari infrastruktur. Hasil monitor ini akan ditindaklanjuti dengan *tuning* sebagai upaya untuk mengoptimalkan kinerja. *Tuning* basis data adalah peningkatan kinerja pada desain basis data secara fisik yang mencakup relasi dan view sesuai dengan kebutuhan pengguna. Tujuan dari *tuning* basis data adalah Mengurangi waktu respon dari sistem ke pengguna akhir. Serta Mengurangi sumber daya yang diperlukan untuk melakukan pengolahan data dalam basis data. Index adalah sebuah object dalam sistem basis data yang bertujuan untuk mempercepat proses pencarian data. database yang tidak dilengkapi dengan index akan menurunkan performa database itu sendiri yang dimana saat pencarian data akan banyak sekali memakan resource dari CPU.

#### 1.2 Tuning : Setting Configuration DBMS

```
# Comment the following if you are using InnoDB tables
#skip-innodb
innodb_data_home_dir="C:/xampp/mysql/data"
innodb_data_file_path=ibdata1:10M:autoextend
innodb_log_group_home_dir="C:/xampp/mysql/data"
#innodb_log_arch_dir = "C:/xampp/mysql/data"
## You can set ..buffer_pool_size up to 50 - 80 %
## of RAM but beware of setting memory usage too high
innodb_buffer_pool_size=16M
## Set ..log_file_size to 25 % of buffer pool size
innodb_log_file_size=5M
innodb_log_buffer_size=8M
innodb_flush_log_at_trx_commit=1
innodb_lock_wait_timeout=50
```

**Gambar 1.1** Sebelum Konfigurasi

Gambar diatas menunjukkan keadaan awal sebelum melakukan proses konfigurasi pada Xampp. Adapun proses konfigurasi dilakukan dengan mengubah nilai innodb\_buffer\_pool\_size, innodb\_log\_file\_size, innodb\_flush\_log\_at\_trx\_commit dan innodb\_lock\_wait\_timeout.

```
# Comment the following if you are using InnoDB tables
#skip-innodb
innodb_data_home_dir="C:/xampp/mysql/data"
innodb_data_file_path=ibdata1:10M:autoextend
innodb_log_group_home_dir="C:/xampp/mysql/data"
#innodb_log_arch_dir = "C:/xampp/mysql/data"
## You can set ..buffer_pool_size up to 50 - 80 %
## of RAM but beware of setting memory usage too high
innodb_buffer_pool_size=16M
## Set ..log_file_size to 25 % of buffer pool size
innodb_log_file_size=3M
innodb_log_buffer_size=6M
innodb_flush_log_at_trx_commit=1
innodb_lock_wait_timeout=40
```

### **Gambar 1.2 Setelah Konfigurasi**

Gambar diatas menunjukkan keadaan setelah melakukan proses konfigurasi pada Xampp. Adapun proses konfigurasi dilakukan dengan mengubah nilai innodb\_buffer\_pool\_size, innodb\_log\_file\_size, innodb\_flush\_log\_at\_trx\_commit dan innodb\_lock\_wait\_timeout.

## BAB II HASIL DAN PEMBAHASAN

### B. Deskripsi Percobaan

#### 2.1 Tunning : Indexing

1. Data 1 (advisor = 100, student = 100, section = 200,takes = 200)

Dibawah ini merupakan tabel respons time pada setiap query.

Query	Waktu sebelum tunning (s)	Waktu sesudah tunning (s)
SELECT * FROM student;	0.0	0.0
SELECT * FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT `name`, dept_name FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id;	0.0	0.0
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;	0.0	0.0

2. Data 2 (advisor = 200, student = 200, section = 400,takes = 400)

Dibawah ini merupakan tabel respons time pada setiap query.

Query	Waktu sebelum tunning (s)	Waktu sesudah tunning (s)
SELECT * FROM student;	0.0	0.0
SELECT * FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT `name`, dept_name FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id;	0.0	0.0

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;	0.0	0.0
--	-----	-----

3. Data 3 (advisor = 500, student = 500, section = 1000,takes = 1000)

Dibawah ini merupakan tabel respons time pada setiap query.

Query	Waktu sebelum tunning (s)	Waktu sesudah tunning (s)
SELECT * FROM student;	0.0	0.0
SELECT * FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT `name`, dept_name FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id;	0.01	0.01
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;	0.01	0.01

Dari data diatas menunjukkan bahwa dalam pemrosesan waktu respon proses query sebelum melakukan indeks tunning membutuhkan waktu respon yang sama lama dengan setelah melakukan proses indeks tunning. Menurut teori proses pencarian query seharusnya menjadi lebih cepat jika menggunakan index tuning tetapi dari hasil data 1 sampai 3 tidak menunjukkan adanya perbedaan antara sebelum dan sesudah indexing yang terjadi karena data yang diproses masih dalam jumlah yang sedikit sehingga tidak terlihat perbedaannya . Pada data diatas dapat dilihat bahwa waktu respon paling lama yaitu pada query ke 4 dan 5 untuk proses sebelum dan

sesudah melakukan tuning index. Percobaan dilakukan hanya dengan data 1 sampai 3 saja dikarenakan pada saat melakukan percobaan pada data ke 4 sampai 7 terjadi kesalahan pada PC sehingga proses tidak dapat dilanjutkan kembali. Semakin banyak data yang dari suatu database akan membuat proses tuning akan menjadi lebih banyak serta membutuhkan waktu respon yang lama dalam proses indeks tuning.

## 2.2 Tunning : Setting Configuration DBMS

1. Data 1 (advisor = 100, student = 100, section = 200,takes = 200)

Dibawah ini merupakan tabel respons time pada setiap query.

Query	Waktu sebelum tuning (s)	Waktu sesudah tuning (s)
SELECT * FROM student;	0.0	0.0
SELECT * FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT `name`, dept_name FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id;	0.0	0.0
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;	0.0	0.0

2. Data 2 (advisor = 200, student = 200, section = 400,takes = 400)

Dibawah ini merupakan tabel respons time pada setiap query.

Query	Waktu sebelum tunning (s)	Waktu sesudah tunning (s)
SELECT * FROM student;	0.0	0.0
SELECT * FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT `name`, dept_name FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id;	0.0	0.0
SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,secti on.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;	0.0	0.0

3. Data 3 (advisor = 500, student = 500, section = 1000,takes = 1000)

Dibawah ini merupakan tabel respons time pada setiap query.

Query	Waktu sebelum tunning (s)	Waktu sesudah tunning (s)
SELECT * FROM student;	0.0	0.0
SELECT * FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT `name`, dept_name FROM student WHERE tot_cred > 30;	0.0	0.0
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id;	0.0	0.0



SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,secti on.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;	0.01	0.006
--	------	-------

Dari data diatas menunjukkan bahwa setelah melakukan proses konfigurasi DBMS dengan mengubah dan mengatur nilai pada pengaturan konfigurasi di Xampp dimana kita mengubah nilai dari innodb\_buffer\_pool\_size, innodb\_log\_file\_size, innodb\_flush\_log\_at\_trx\_commit, dan innodb\_lock\_wait\_timeout menjadi nilai yang lebih kecil dari keadaan awal, Dari data diatas menunjukkan bahwa dalam pemrosesan waktu respon proses query sebelum melakukan setting config DBMS membutuhkan waktu respon yang sama lama dengan setelah melakukan proses setting config DBMS. Menurut teori proses pencarian query seharusnya menjadi berubah jika setting config DBMS diubah tetapi dari hasil data 1 sampai 3 tidak menunjukkan adanya perbedaan antara sebelum dan sesudah indexing yang terjadi karena data yang diproses masih dalam jumlah yang sedikit sehingga tidak terlihat perbedaannya . Pada data diatas dapat dilihat bahwa waktu respon paling lama yaitu pada query ke 4 dan 5 untuk proses sebelum dan sesudah melakukan tuning index. Percobaan dilakukan hanya dengan data 1 sampai 3 saja dikarenakan pada saat melakukan percobaan pada data ke 4 sampai 7 terjadi kesalahan pada PC sehingga proses tidak dapat dilanjutkan kembali.

### **BAB III KESIMPULAN**

Dari percobaan yang telah dilakukan dapat disimpulkan bahwa *Tuning* adalah sebuah upaya untuk memaksimalkan sebuah DBMS yang ada agar dapat berfungsi dengan optimal. Tujuan dari tuning basis data adalah Mengurangi waktu respon dari sistem ke pengguna akhir. Serta Mengurangi sumber daya yang diperlukan untuk melakukan pengolahan data dalam basis data. Dari data yang diperoleh selama percobaan Proses indeks tuning dan konfigurasi belum mempengaruhi performa dari suatu database. Dimana dengan menggunakan indeks tuning dan konfigurasi sehausnya dapat membuat waktu respon pada database menjadi lebih efisien.

## DAFTAR PUSTAKA

- [1] AlfiantoFerdian . Oktober 12 2013” Meningkatkan Performa Mysql/MariaDBDenganMySqlTuner”di: <https://tuts.web.id/meningkatkan-performa-mysql-mariadb-dengan-mysqltuner/> (Diakses pada tanggal 20 Desember 2019)
- [2] Clarissa E.Amos Pah . Juni 18,2019” Seberapa Peringkat Performa Dan Tuning Pada SQL Database ”di: <https://mti.binus.ac.id/2019/06/18/seberapa-pentingkah-performance-dan-tuning-pada-sql-database/>(Diakses pada tanggal 20 Desember 2019)
- [3] SidikFajarBayu. Mei,7 2017” Apa Itu Indexing Pada Query Dan Fungsinya”di: <https://kursuswebprogramming.com/apa-itu-indexing-pada-query-dan-apa-fungsinya/> (Diakses pada tanggal 20 Desember 2019)