

ANALISA PERBANDINGAN TINGKAT EFISIENSI ALGORITMA DATA DEFINITION LANGUAGE (DDL) COPY, INPLACE, INSTANT DATABASE MYSQL

Comparison Analysis Of Efficiency Level Of Mysql Data Definition Language Algorithm Copy, Inplace, Instant Database

Frans Sinata, 11834@lecturer.ubm.ac.id

¹⁾Program Studi/Departemen Informatika, Universitas Bunda Mulia

Diterima 1 Jan 2023 / Disetujui 12 April 2023

MySQL database schema changes are becoming more frequent than ever, Four out of five application updates (releases) require appropriate database changes, For Database schema changes are more often a repetitive task, it may be a request from the application team to add or modify a column in a table and many cases other. This study aims to measure the level of time efficiency to modify the sales transaction table schema which consists of 50000 number of records in the MySQL database with the Data Definition Language (DDL) Copy, Inplace and Instant algorithm on the MySQL database.

Keywords: Database,MySQL,Data Definition Language (DDL),Database,Scheme,Record

ABSTRAK

Perubahan skema database *MySQL* menjadi lebih sering daripada sebelumnya,Empat dari lima pembaruan aplikasi (rilis) memerlukan perubahan database yang sesuai, Untuk perubahan skema *Database* lebih sering merupakan tugas yang berulang, mungkin permintaan dari tim aplikasi untuk menambahkan atau memodifikasi kolom dalam tabel dan banyak kasus lainnya. Pada penelitian ini bertujuan untuk mengukur tingkat efisiensi waktu untuk memodifikasi skema *database* (*Database Scheme*) tabel transaksi penjualan yang terdiri dari 50000 jumlah *record* pada database *MySQL* dengan algoritma *Data Definition Language (DDL) Copy,Inplace* dan *Instant* pada database *MySQL*

Kata Kunci: Database,MySQL,Data Definition Language (DDL),Database Scheme,Record

PENDAHULUAN

Seiring kemajuan teknologi informasi yang pesat, penggunaan *Database MySQL* merupakan komponen penting bagi suatu organisasi atau perusahaan dalam penyimpanan data yang semakin banyak baik untuk keperluan operasional dan pemanfaatan informasi untuk pihak internal maupun eksternal.

MySQL[1] adalah sebuah perangkat lunak sistem manajemen basis data *SQL* yang multi alur, multi pengguna, dengan sekitar 6 juta instalasi di seluruh dunia.

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (*RDBMS*) yang didistribusikan secara gratis di bawah *lisensi GPL (General Public License)*. Setiap pengguna dapat secara bebas menggunakan, mendistribusikan, dan membuat karya turunan dari *MySQL*.

Skema Basis Data atau *Database Scheme* adalah *framework* atau *blueprint* yang digunakan oleh para programmer untuk membangun dan menulis kode database. Skema bisa membantu user database memahami informasi yang disimpan[2].

Tabel atau *Table* digunakan untuk mengatur data dalam bentuk baris dan kolom dan digunakan untuk menyimpan dan menampilkan record dalam format struktur[3].

Record adalah kumpulan *field* yang lengkap dan kompleks, biasanya satu *record* dihitung dalam satu baris yang ada pada table basis data. Singkatnya basis data (*database*) merupakan element data yang berkaitan satu sama lain[4].

Field[4] adalah kumpulan karakter yang membentuk suatu arti, misal dalam *field* selalu ada KodeBarang, NamaBarang, KodeBarang, JenisBarang, dan sebagainya, maka semua acuannya harus yang berkaitan dengan KodeBarang. Biasa juga arti field merupakan tempat berupa kolom yang ada pada table untuk mengisi data, yang biasanya ada pada kolom database seperti di Microsoft Acces atau Database lainnya.

Untuk melakukan modifikasi skema basis data atau *Database Schema* dibutuhkan perintah *DDL (Data Definition Language)*[5] yang merupakan sekumpulan set perintah yang bertujuan untuk mendefinisikan atribut – atribut database, tabel, atribut kolom (field), maupun batasan terhadap suatu atribut dan relasi/hubungan antar tabel.

a. Latar Belakang

Semakin berkembangnya suatu organisasi atau perusahaan, secara otomatis pertumbuhan data akan semakin besar dan juga akan membutuhkan perubahan skema tabel pada database *MySQL*. Hal ini menjadi bagian penting dalam menunjang kebutuhan penambahan informasi bagi kepentingan internal maupun eksternal.

b. Identifikasi Masalah

Dengan menggunakan perintah *Query DDL (Data Definition Language)* seperti secara konvensional untuk melakukan modifikasi skema pada tabel dengan jumlah record yang semakin banyak (di atas 10000 record), akan terganggunya proses manipulasi data pada tabel pada saat tabel sedang digunakan oleh para pengguna melalui sistem aplikasi

c. Tujuan dan Manfaat Penelitian

Berfokus pada permasalahan di atas, penulis bermaksud melakukan uji coba penelitian dengan melakukan perubahan skema tabel penjualan sebanyak 50000 *record* pada database *MySQL* dengan menggunakan algoritma *Data Definition Language (DDL) Inplace, Copy, Instant*

Tujuan dari penelitian ini untuk mengetahui tingkat efisiensi penggunaan algoritma *Inplace, Copy* dan *Instant* dalam perubahan skema tabel pada *Database MySQL*.

METODOLOGI PENELITIAN

Untuk melakukan pengujian perubahan skema pada tabel penjualan sebanyak 50000 record, maka akan digunakan perintah *DDL ALTER TABLE* dengan menggunakan algoritma *COPY, INPLACE, dan INSTANT* melalui *interface database MySQL SQLYog*.

SQLyog

SQLYog[6] adalah salah satu aplikasi *Client MySQL (My Structured Query Language)* yang dikembangkan oleh Webyog, Inc. Saat ini ada dua versi *SQLyog* yang bisa digunakan yaitu *enterprise* dan *community*.

Algoritma Inplace

Algoritma *Inplace*[7] melakukan operasi di tempat ke tabel asli dan menghindari melakukan penyalinan tabel dan pembuatan ulang kembali. Jika algoritma *Inplace* ditentukan dengan klausa *ALGORITHM* jika operasi *ALTER TABLE* tidak mendukung algoritma *Inplace*, maka alter akan keluar dengan kesalahan dengan menyarankan kemungkinan algoritma yang dapat digunakan. Algoritma ini menggunakan 2 variabel penting pada saat melakukan operasi perubahan skema pada tabel, yaitu:

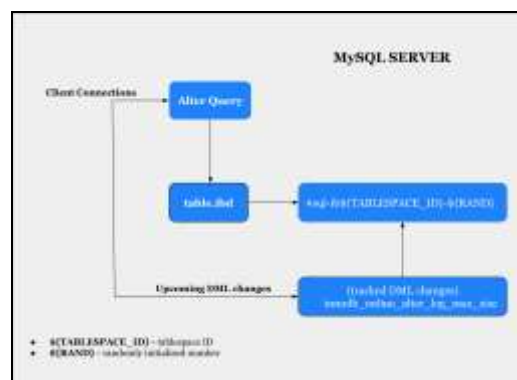
1. *tmp_dir* yang digunakan untuk melakukan proses *sort* pada *file* yang disimpan pada direktori default user/tmp, jika direktori tmp tidak cukup, bisa mendefinisikan secara eksplisit sistem variable yaitu *innodb_tmpdir*
2. *innodb_online_alter_log_max_size* digunakan untuk melacak perubahan *data* atau *record* pada tabel oleh query *INSERT, UPDATE, DELETE* di dalam tabel selama operasi *DDL* (*Data Definition Language*)



Gambar 1. SQLyog

Contoh :

```
mysql> alter table sbtest1 add column h int(11) default null,algorithm=inplace;
Query OK, 0 rows affected (14.12 sec)
Records: 0 Duplicates: 0 Warnings: 0
```



Gambar 2. Internal Flow Algoritma *Inplace* pada saat tabel direbuild

Operation	In Place	Rebuilds Table	Permits Concurrent DML
Adding a primary key	Yes	Yes	Yes
Dropping a primary key and adding another	Yes	Yes	Yes
Adding a column	Yes	Yes	Yes
Dropping a column	Yes	Yes	Yes
Reordering columns	Yes	Yes	Yes
Changing the ROW_FORMAT	Yes	Yes	Yes
Making a column NULL	Yes	Yes	Yes
Making a column NOT NULL	Yes	Yes	Yes
Changing the KEY_BLOCK_SIZE	Yes	Yes	Yes
Specifying a character set	Yes	Yes	No
Optimizing a table	Yes	Yes	Yes
Performing a full rebuild	Yes	Yes	Yes

Gambar 3. Tabel Operasi Algoritma Inplace

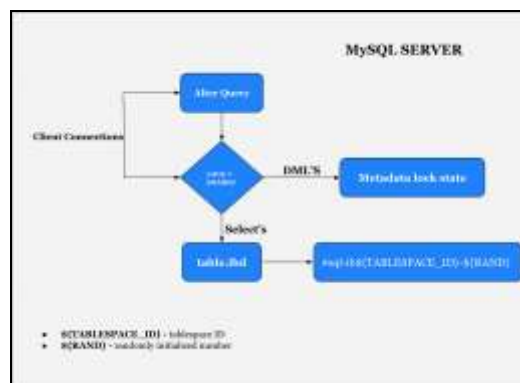
Algoritma Copy

Algoritma *Copy*[7] mengubah skema tabel yang ada dengan membuat tabel sementara baru dengan skema yang diubah, setelah memigrasikan data sepenuhnya ke tabel sementara baru, ia menukar dan menghapus tabel lama.

Contoh:

```
mysql> alter table sbtest1 modify column h varchar(20) not null,algorithm=inplace;
```

ERROR 1846 (0A000): ALGORITHM=INPLACE is not supported. Reason: Cannot change column type INPLACE. Try ALGORITHM=COPY.



Gambar 4. Internal Flow Algoritma Copy pada saat melakukan operasi penyalinan (Copy) tabel

Algoritma Instant

Dalam perbaikan lebih lanjut dalam *DDL online* (penambahan kolom) *MySQL 8.0* telah muncul algoritma Instant (patch dari *TENCENT*). Fitur ini membuat perubahan tabel instan dan di tempat untuk penambahan kolom dan memungkinkan *Data Manipulation Language (DDL)* (*Insert, Update, Delete*) bersamaan dengan peningkatan respons dan ketersediaan di lingkungan produksi yang sibuk.

Jika *ALGORITHM* tidak ditentukan, server pertama-tama akan mencoba algoritma *DEFAULT=INSTANT* untuk semua penambahan kolom. Jika tidak bisa, maka server akan

mencoba algoritma *INPLACE*; dan jika itu tidak dapat didukung, pada akhirnya server akan mencoba algoritma *COPY*.

Algoritma *INSTANT*[7] hanya melakukan perubahan metadata dalam kamus data. Itu tidak memperoleh kunci metadata apa pun selama perubahan skema dan karena tidak menyentuh file data tabel.

Contoh:

```
mysql> alter table city add pincode int(11) default null, algorithm=INSTANT;
```

```
Query OK, 0 rows affected (0.04 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

Operation	Instant	Supports Table Constraints	Supports Concurrent DML
Adding a column	Yes	No	Yes
Setting a column with default value	Yes	No	Yes
Dropping a column default value	Yes	No	No
Modifying the column name	Yes	No	Yes
Adding or dropping a virtual column	Yes	No	Yes
Renaming a table	Yes	No	Yes

Gambar 5. Operasi yang didukung Algoritma *Instant*

HASIL DAN PEMBAHASAN

Penulis melakukan perubahan skema tabel *sales_so_header* (50000 record data) dengan menambahkan 3 field *phone1*, *phone2*, *phone3* dengan tipe data *varchar(30)* melalui penggunaan algoritma *copy*, *inplace*, dan *instant*.

Pengujian Algoritma *Inplace*

```
ALTER TABLE sales_so_header ADD phone1 VARCHAR(30) NOT NULL DEFAULT "",  
ALGORITHM=inplace;
```

```
(0 row(s) affected)
```

```
Execution Time : 00:00:00:016
```

```
Transfer Time : 00:00:01:031
```

```
Total
```

Pengujian Algoritma *Copy*

```
ALTER TABLE sales_so_header ADD phone2 VARCHAR(30) NOT NULL DEFAULT "",  
ALGORITHM=copy;
```

```
(40212 row(s) affected)
```

```
Execution Time : 00:00:01:188
```

```
Transfer Time : 00:00:01:031
```

```
Total Time : 00:00:02:219
```

Pengujian Algoritma *Instant*

```
ALTER TABLE sales_so_header ADD phone3 VARCHAR(30) NOT NULL DEFAULT "",  
ALGORITHM=instant;
```

```
(0 row(s) affected)
```

```
Execution Time : 00:00:00:016
```

Transfer Time : 00:00:01:000
Total Time : 00:00:01:016

Tabel 1. Pengujian

ALGORITHM	TIME		
	EXECUTION	TRANSFER	TOTAL
INPLACE	00:00:00:016	00:00:01:031	00:00:01:047
COPY	00:00:01:188	00:00:01:031	00:00:02:219
INSTANT	00:00:00:016	00:00:01:000	00:00:01:016

SIMPULAN

Hasil yang didapat melalui pengujian di atas, dapat disimpulkan penggunaan algoritma instant untuk melakukan perubahan skema penambahan *field* hanya membutuhkan total waktu 0.16 detik dibandingkan dengan algoritma *copy* dan *inplace*.

1. *INSTAN* akan menjadi algoritma yang tepat untuk digunakan ketika Versi *MySQL* lebih besar dari 8 dan juga berdasarkan tipe *Alter* yang akan digunakan.
2. Menggunakan algoritma *COPY* untuk tabel yang lebih besar, dengan lebih banyak *slave asinkron*, akan menjadi operasi yang lebih menguras waktu

Dengan hasil pengujian ketiga algoritma, diharapkan memilih algoritma yang tepat dalam penerapan *DDL* yang terkait dengan perubahan skema tabel atau database.

DAFTAR PUSTAKA

Berikut ini contoh penulisan daftar pustaka:

- [1] id.wikipedia.org, "MySQL," *id.wikipedia.org*, 2022. <https://id.wikipedia.org/wiki/MySQL> (accessed Oct. 14, 2022).
- [2] Y. W, "Apa Itu Skema Database? Pengertian dan Contohnya Lengkap," *hostinger.co.id*, 2022. <https://www.hostinger.co.id/tutorial/skema-database> (accessed Oct. 10, 2022).
- [3] javatpoint.com, "MySQL CREATE TABLE," *javatpoint.com*, 2021. <https://www.javatpoint.com/mysql-create-table> (accessed Oct. 09, 2022).
- [4] Pengertianku.net, "Pengertian Field, Record, Table, File, Data, dan Basis Data Lengkap," *pengertianku.net*, 2022. <https://www.pengertianku.net/2014/12/pengertian-field-record-table-file-data-dan-basis-data-lengkap.html> (accessed Oct. 14, 2022).
- [5] Admin, "DATA DEFINITION LANGUAGE (DDL)," *smkn1pangandaran.sch.id*, 2019. <https://smkn1pangandaran.sch.id/sakola/public/artikel/3/data-definition-language-ddl> (accessed Oct. 14, 2022).
- [6] L. P. Jan, "Panduan Remote Database MySQL dengan SQLyog di Rumahweb," *rumahweb.com*, 2020. <https://www.rumahweb.com/journal/panduan-remote-database-mysql-dengan-sqlyog-di-rumahweb/> (accessed Oct. 12, 2022).
- [7] W. Akram, "An Overview of DDL Algorithm's in MySQL (covers MySQL 8)," *mydbops.wordpress.com*, 2020. <https://mydbops.wordpress.com/2020/03/04/an-overview-of-ddl-algorithms-in-mysql-covers-mysql-8/> (accessed Oct. 11, 2022).