

IMPLEMENTASI REPLIKASI DATA SISTEM INFORMASI AKADEMIK STMIK PROFESIONAL MAKASSAR

Sitti Arni

Program Studi : Sistem Informasi
STMIK Profesional Makassar
arnist19@yahoo.com

Abstrak

Tulisan ini membahas tentang Replikasi Database Akademik pada STMIK Profesional, dimana dalam aktivitas akademik terjadi di dua tempat yang berbeda. Masalah yang terjadi saat ini mahasiswa S1 kuliah di kampus I sedangkan administrasi akademiknya diselesaikan di kampus II, sehingga untuk menyelesaikan administrasi akademik mahasiswa harus ke kampus II. Untuk mengatasi hal tersebut maka implementasi Replikasi Data untuk sistem terdistribusi merupakan salah satu solusinya. Dalam tulisan ini dibahas bagaimana melakukan konfigurasi pada komputer server dan komputer slave yang terkoneksi melalui jaringan sehingga replikasi data dapat dilakukan. Konfigurasi yang tepat merupakan syarat yang harus terpenuhi untuk mengimplementasikan Replikasi Data.

Kata Kunci : Replikasi Data, Sistem Terdistribusi

A. PENDAHULUAN

Ciri utama sistem informasi adalah distribusi dan interaksi basis data. Sistem informasi merupakan kesatuan elemen yang tersebar dan saling berinteraksi yang menciptakan informasi. Proses interaksi tersebut berupa proses data dengan cara pemasukan, pengolahan, integrasi, pengolahan, komputasi atau perhitungan, penyimpanan, serta distribusi data atau informasi. Perkembangan sistem informasi didukung oleh kemajuan teknologi jaringan komputer sehingga memungkinkan dilakukannya komunikasi

dan interaksi antar data yang secara fisik terpisah. Teknologi ini mengatasi semua hambatan baik dimensi waktu maupun dimensi geografis, yang dapat dilakukan dimana saja selama terhubung dengan jaringan komputer baik melalui kabel maupun nirkabel.

Saat ini STMIK Profesional mengimplementasikan sistem informasi dalam operasionalnya. Setiap biro menggunakan sistem informasi yang terintegrasi untuk menjalankan aktifitasnya, salah satunya adalah biro akademik. Biro akademik saat ini berada pada dua tempat yakni kampus I yang

terletak di Jl. A.P.Pettarani no.27 melayani mahasiswa program D3 sedangkan kampus II yang terletak di Jl.G.Latimojong No.18 melayani mahasiswa program studi S1. Perkuliahan dan administrasi untuk mahasiswa program D3 sepenuhnya dilaksanakan di kampus I sedangkan perkuliahan untuk mahasiswa program studi S1 dilaksanakan di kampus I dan kampus II, untuk administrasinya dilaksanakan di kampus II.

Masalah yang timbul saat ini, ketika mahasiswa program studi S1 berada di kampus I untuk mengikuti perkuliahan dan pada saat yang sama akan menyelesaikan administrasi akademik maka mahasiswa yang bersangkutan harus ke kampus II. Idealnya ketika mahasiswa berada di kampus baik di kampus I maupun di kampus II, mahasiswa dapat menyelesaikan administrasi akademiknya tanpa harus berpindah kampus. Untuk itu penulis mencoba memberi gambaran tentang implementasi replikasi data pada sistem informasi akademik.

Penelitian ini diharapkan dapat membantu pihak pengelola untuk mengimplementasikan sistem terdistribusi namun tetap sinkron antara database sehingga konsistensi data tetap terjamin.

B. TINJAUAN PUSTAKA

Replikasi

Replikasi adalah suatu teknik untuk melakukan copy dan pendistribusian data dan objek-objek database dari satu database ke database lain dan melaksanakan sinkronisasi antara database sehingga konsistensi data dapat terjamin. Dengan menggunakan teknik replikasi, data dapat didistribusikan ke lokasi yang berbeda melalui koneksi jaringan lokal maupun internet. Replikasi juga memungkinkan untuk mendukung kinerja aplikasi, penyebaran data fisik sesuai dengan penggunaannya, seperti pemrosesan transaksi online dan DSS (Decision Support System) atau pemrosesan database terdistribusi melalui beberapa server.

Replikasi merupakan Sistem basis data terdistribusi yang dapat menyimpan duplikat dari data yang sama dalam site yang berbeda agar perolehan informasi yang semakin cepat dan melakukan toleransi kesalahan. Replikasi memungkinkan organisasi untuk memindahkan basis data dari mainframe terpusat ke server-server yang spesifik pada lokasi tertentu, dekat dengan para pengguna.

Strategi Replikasi

1. **Sinkron:** sebelum seluruh proses transaksi *update* dinyatakan selesai, data yang telah dimodifikasi disinkronkan ke setiap duplikatnya; proses ini harus menunggu hingga data di tempat penyimpanan duplikat selesai ditulis sebelum dilakukan perubahan lainnya sehingga menjadi lebih kompleks. **Synchronous data replication** terjadi ketika aplikasi *update* tabel di lokasi lokal replikasi, pada waktu yang sama juga melakukan proses *update* tabel yang sama di lokasi replikasi yang lain
2. **Asinkron:** *copy* data diperbaharui secara periodik berdasarkan data utama yang diperbaharui; proses penulisan data selesai tanpa perlu menunggu penulisan data di tempat penyimpanan duplikat selesai; proses ini memang meningkatkan kinerja sistem namun risikonya, inkonsistensi data bisa terjadi. **Asynchronous data replication** : terjadi ketika aplikasi *update* table di lokasi lokal replikasi, penyimpanan informasi replikasi di dalam antrian local, dan meneruskan informasi replikasi ke site replikasi yang lain beberapa waktu kemudian.

Jenis-jenis Replikasi

1. **Snapshot replication,** mendistribusikan data yang dapat dilihat pada saat tertentu tanpa melakukan *update*.
2. **Transactional replication,** memelihara konsistensi transaksi yang terjadi
3. **Merge replication,** memungkinkan pengguna bekerja dan merubah data sesuai dengan wewenangnya.

Sistem Terdistribusi

Sistem distribusi adalah sebuah sistem yang komponennya berada pada jaringan komputer. Komponen tersebut saling berkomunikasi dan melakukan koordinasi hanya dengan pengiriman pesan (*message passing*). Sistem terdistribusi merupakan kebalikan dari Sistem Operasi Prosesor Jamak. Pada sistem tersebut, setiap prosesor memiliki memori lokal tersendiri. Kumpulan prosesornya saling berinteraksi melalui saluran komunikasi seperti LAN dan WAN menggunakan protokol standar seperti TCP/IP. Karena saling berkomunikasi, kumpulan prosesor tersebut mampu saling berbagi beban kerja, data, serta sumberdaya lainnya.

Dalam penggunaannya sistem terdistribusi sangat diperlukan karena:

1. **Performance**, sekumpulan prosesor dapat menyediakan kinerja yang lebih tinggi dari pada komputer yang terpusat
2. **Distribution**, banyak aplikasi yang terlibat, sehingga lebih baik jika dipisah dalam mesin yang berbeda.
3. **Reliability**, jika terjadi kerusakan pada salah satu mesin, tidak akan mempengaruhi kinerja sistem secara keseluruhan
4. **Incremental Growth**, mesin baru dapat ditambahkan jika kebutuhan proses meningkat
5. **Sharing Data/Resource**, segala hal yang dapat digunakan bersama dalam jaringan komputer meliputi hardware dan software.
6. **Communication**, menyediakan fasilitas komunikasi antar manusia.

Model Sistem Terdistribusi

1. **Sistem client – server**, merupakan bagian dari model sistem terdistribusi yang membagi jaringan berdasarkan pemberi dan penerima jasa layanan.
2. **Sistem point to point**, merupakan bagian dari model sistem terdistribusi dimana sistem dapat sekaligus berfungsi sebagai client maupun server.

3. **Sistemterkluster**, adalah gabungan dari beberapa sistem individual (komputer) yang dikumpulkan pada suatu lokasi, saling berbagi tempat penyimpanan data (storage), dan saling terhubung dalam jaringan lokal (Local Area Network).

Tantangan Sistem Terdistribusi

1. Keheterogenan komponen (heterogeneity)
2. Keterbukaan (openness)
3. Keamanan (security)
4. Scalability
5. Penanganan kegagalan (failure handling)
6. Concurrency of components
7. Transparansi

C. PEMBAHASAN DAN HASIL

Langkah-langkah Replikasi Database dengan MYSQL

1. Pastikan antara komputer Master dengan Komputer Slave terhubung dalam jaringan.
2. Teskoneksi antara komputer tersebut dengan melakukan ping melalui command prompt pada kedua komputer.
3. Buatlah Database dengan struktur dan isi table yang sama, baik pada

komputer Master maupun pada komputer Slave.

4. Setting Komputer Master
5. Setting Komputer Slave
6. Uji Replikasi Data yang telah dibuat.

Setting Komputer Master

1. Setup lokasi file log bin dengancara menambahkan kode berikut melalui editor notepad ke directory *C:\xampp\mysql\bin\my.cnf*, dibawah [*mysqld*] ketik perintah berikut :

server-id=1

log-bin=C:\xampp\mysql\mysql-bin.log

binlog-do-db = akademik

Server-id=1 digunakan untuk mendefenisikan id-server komputer master adalah 1 (id-server ini harus berbeda antar komputer).

Log-bin=C:\xampp\mysql\mysql-bin.log

digunakan untuk mengaktifkan pencatatan pada file binary log dengan nama file *mysql-bin.log*, sehingga semua perubahan yang terjadi pada database akademik akan tercatat pada log ini.

Binlog-do-db=akademik

menunjukkan bahwa database

akademik akan dicatat log-nya dan akan disinkronkan ke komputer slave.

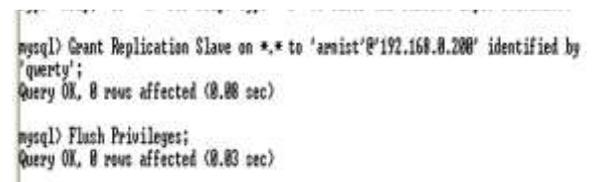
2. Simpan file konfigurasi MySQL tersebut, selanjutnya restart service MySQL agar konfigurasi yang baru dapat berjalan.

3. Daftarkan user baru untuk melakukan replikasi terhadap database Akademik dengan cara menuliskan perintah berikut pada my sql prompt :

```
mysql> grant replication slave on *.* to 'arnist'@'192.168.0.200' identified by
```

```
'<password_use_slaver>';
```

```
Mysql> flush privileges;
```



Gambar 2. Layar perintah Pemberian Izin Akses

Grant Replication Slave digunakan untuk memberi izin akses kepada user yang login menggunakan user name dan password yang sudah didaftar dengan hak replikasi slave menggunakan komputer dengan alamat IP yang ditentukan.

Flush Privileges digunakan untuk merefresh konfigurasi yang telah

dilakukan pada table user dalam database mysql.

4. Aktifkan database yang akan direplikasi dengan menuliskan perintah berikut :

```
mysql>use akademik;
```

```
mysql> flush tables with read lock;
```

Use Akademik digunakan untuk mengaktifkan atau memilih database Akademik.

5. Tampilkan informasi binary log dari database yang direplikasi pada komputer master dengan menuliskan perintah :

```
mysql>reset master;
```

```
mysql>show master status;
```

```
mysql> show master status;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000001 | 554     | akademik     |                    |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Gambar 3. Layout Binary log Database Akademik

Informasi binary log database Akademik pada komputer master digunakan untuk setting komputer slave, dimana informasi yang dibutuhkan antara lain file dan posisinya.

6. Selanjutnya, lakukan konfigurasi pada komputer Server Slave.

Setting komputer Slave

1. Setup lokasi file log bin pada komputer slave dengan cara menambahkan kode berikut melalui editor

```
notepadkedirectory C:/xampp/mysql/bin/my.cnf, dibawah [mysqld]
```

```
server-id=2
```

```
master-host=192.168.0.200
```

```
master-user=arnist
```

```
master-password=qwerty
```

```
master-connect-retry=60
```

```
replicate-do-db = akademik
```

Nilai **server-id=2**, dapat diganti dengan angkat integer positif selain 1 yang sudah digunakan pada komputer Master.

Master-host=192.168.0.200,

digunakan untuk mendefenisikan alamat IP dari komputer master tempat database yang akan direplikasi

Master-user=arnist, digunakan untuk mendefenisikan user name yang akan digunakan untuk login dari komputer slave

Master-password=qwerty,

digunakan untuk mendefenisikan password yang akan digunakan untuk login dari komputer slave

Master-connect-retry=60,

digunakan untuk mendefenisikan

bahwa master akan melakukan koneksi ke slave setiap 60 detik.

Replicate-do-db=akademik,

digunakan untuk mendefinisikan nama database yang akan direplikasi yaitu akademik.

2. Simpan file konfigurasi MySQL tersebut selanjutnya restart service MySQL agar konfigurasi yang baru dapat berjalan.
3. Lakukan aktivasi replikasi untuk sinkronisasi komputer master dengan komputer slave dengan perintah berikut ini:

```
mysql>slave stop;
```

```
mysql>change master to
```

```
    master_host='192.168.1.200
```

```
,
```

```
    master_user='arnist',
```

```
    master_password='qwerty',
```

```
    master_log_file='mysql-
```

```
bin.000001',
```

```
    master_log_pos=554
```

```
mysql>start slave;
```

```
mysql>flush privileges;
```

```
mysql> CHANGE MASTER TO MASTER_HOST='192.168.0.200',MASTER_USER='arnist',MASTER_PASSWORD='qwerty',MASTER_LOG_FILE='mysql-bin.000001',MASTER_LOG_POS=554;
Query OK, 0 rows affected (0.02 sec)
```

Gambar 5. Aktivasi Replikasi

Slave stop, digunakan untuk menghentikan proses yang berjalan pada komputer slave.

Change master to master host=192.168.1.200, digunakan untuk mendefinisikan alamat IP komputer master yang akan direplikasi databasenya.

Master_user='arnist', digunakan untuk mendefinisikan user name yang akan digunakan untuk login dari komputer slave.

Master_password='qwerty', digunakan untuk mendefinisikan password yang akan digunakan untuk login dari komputer slave.

Master_log_file='mysql-bin.000001', menjelaskan bahwa slave membaca log pada master dengan nama log mysql-bin.000001.

Master_log_pos=554, digunakan untuk menentukan posisi log pada komputer master.

Start slave, digunakan untuk menjalankan slave.

Flush privileges, digunakan untuk mereshuffle konfigurasi yang telah dilakukan.

4. Aktivasi replikasi pada komputer slave dengan menuliskan perintah :

```
mysql>show slave status\G
```

```
mysql> START SLAVE;
Query OK, 0 rows affected (0.01 sec)

mysql> show slave status\G;
+-----+
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.0.200
Master_User: root
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000001
Read_Master_Log_Pos: 554
Relay_Log_File: mysql-relay-bin.000002
Relay_Master_Log_File: mysql-bin.000001
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Replicate_Do_DB: akademik
Replicate_Ignore_DB:
Replicate_Ignore_Table:
Replicate_Ignore_Table:
Replicate_Wild_Ignore_Table:
Last_Error:
Skip_Counter: 0
Exec_Master_Log_Pos: 554
Relay_Log_Space: 4M
Until_Condition: None
Until_Log_Pos: 0
Master_SSL_Allowed: No
Master_SSL_File:
Master_SSL_Key:
Seconds_Behind_Master: 0
Master_SSL_Ssl_verify: Error: Cert
Last_IO_Error:
Last_SQL_Error:
1 row in set (0.00 sec)
```

Gambar 5. Status Komputer Slave

Jika status Parameter Slave_IO_Running and Slave_SQL_Running: bernilai YES menunjukkan bahwa aktivasi replikasi di slave berhasil dan berjalan dengan baik.

D. PENGUJIAN

Untuk mencoba replikasi, lakukan perubahan data pada salah satu table misalnya tabel kuliah dalam database akademik yang ada di komputer master, selanjutnya refresh database pada komputer slave, maka data yang ada pada table kuliah di komputer slave secara otomatis akan berubah sehingga kedua komputer memiliki data yang sama.

Selanjutnya lakukan penambahan data pada table kuliah dari komputer slave kemudian refresh pada komputer master

maka data pada table kuliah akan mengalami penambahan sesuai data yang dimasukkan pada komputer slave sehingga kedua komputer memiliki data yang sama.

E. KESIMPULAN

1. Untuk melakukan replikasi lakukan konfigurasi terlebih dahulu sehingga Slave_IO_Running dan Slave_SQL_Running bernilai YES.
2. Konfigurasi yang tepat pada komputer master dan komputer slave merupakan syarat yang harus terpenuhi dalam melakukan replikasi data.
3. Replikasi Database adalah bagaimana mengcopy data dalam suatu database ke database lain secara otomatis.

DAFTAR PUSTAKA

[1] **Kadir, Abdul.** 2009. *Mudah Mempelajari Database MySQL*. Yogyakarta : Andi Offset.

[2] **Fathansyah.** 2004. *Sistem Basis Data Lanjutan*. Bandung : Informatika.

[3] **Fathansyah.** 2012. *Basis Data Edisi Revisi*. Bandung : Informatika.

[4] **Indrajani.** 2002. *Pengantar dan Sistem Basis Data*. Elex Media Komputindo.

- [5] **NugrohoAdi.** 2011. Perancangan dan Implementasi Sistem Basis Data. Yogyakarta :Andi.
- [6] **Raghu Ramakrishnan, dkk.** 2004. SistemManajemen Database Edisi 3.Yogyakarta : Andi.