

6

Penerapan Replikasi Asynchronous Pada Basis Data Terdistribusi Untuk Keamanan Data

Oleh : Dyas Widyasto

Abstrak

Keamanan data merupakan salah satu aspek penting dalam sebuah basis data akan tetapi masalah keamanan ini sering kali kurang mendapatkan perhatian dari pemilik system. Pentingnya sebuah data mengharuskan para pemilik kebijakan untuk menjaga dan merawat data agar tidak rentan dengan kerusakan. Dalam sebuah mesin database terdapat banyak fitur terutama untuk menjaga keamanan data, salah satu fitur yang ada adalah replikasi basis data, dalam replikasi basis data sendiri ada beberapa jenis salah satunya replikasi asynchronous. Fitur ini berfungsi untuk meningkatkan availability data, memungkinkan beberapa lokasi menyimpan data yang sama. Hal ini sangat berguna pada saat lokasi-lokasi tersebut membutuhkan data yang sama atau memerlukan server yang terpisah dalam pembuatan aplikasi laporan. Pengguna dapat bekerja dengan meng-copy data pada saat tidak terkoneksi kemudian melakukan perubahan untuk dibuat database baru pada saat terkoneksi dan juga dapat meningkatkan kinerja pembacaan. Dalam bisnis perhotelan data tamu adalah data terpenting dalam perusahaan karena dengan data tersebut perusahaan dapat menjaga hubungan baik dengan para tamu. Begitu pula yang terjadi pada hotel Ole Suites Sentul. Dengan mengaplikasikan replikasi asynchronous di harapkan dapat menjaga keamanan data dengan efektif.

Kata Kunci : *Replikasi Database, MySQL, Asynchronous, Keamanan Data*

Pendahuluan

Latar Belakang

Perkembangan teknologi informasi menuntut semua bidang usaha dapat mengakses informasi secara realtime dan mempunyai keamanan data yang terjamin. Dalam hal ini untuk menyimpan semua informasi dibutuhkan sebuah basis data yang mampu menyimpan data-data informasi dan dapat dengan mudah mengakses, mengedit, dan menyimpan informasi baru. Salah satu system basis data pada sebuah hotel mempunyai peranan penting adalah system basis data tamu. System basis data tamu berfungsi untuk menyimpan data-data tamu yang menginap dihotel tersebut, data-data ini penting untuk mengetahui identitas para tamu sehingga dapat dimanfaatkan untuk media promosi.

System basis data yang digunakan menggunakan satu computer server yang melayani beberapa user, sehingga aktifitas server menjadi rame yang menyebabkan computer server menjadi lambat. Apabila terjadi kerusakan server semua user tidak dapat mengakses system basis data. Betapa pentingnya sebuah basis data sehingga memerlukan suatu device atau teknologi untuk mem back-up data-data tersebut. Teknologi basis data yang digunakan adalah database dengan metode replikasi database. Replikasi database adalah suatu fitur teknologi pada dunia pemrograman database yang memungkinkan suatu basis data untuk mereplikasi atau mengopy datanya ke basis data lain secara otomatis. Secara sederhana replikasi basis data adalah proses otomatis perintah dari basis data satu ke basis data lain yang tentunya terhubung dalam jaringan.

Replikasi merupakan proses penyalinan dan pemeliharaan objek basis data, seperti table-tabel, dalam banyak basis data yang membentuk suatu system basis data terdistribusi.

Perubahan-perubahan yang dilakukan pada suatu tempat dicatat dan disimpan secara lokal sebelum diteruskan dan diterapkan pada setiap lokasi lain (Triyono. 2012).

Salah satu jenis replikasi adalah replikasi asynchronous, replikasi ini menduplikat basis data dari server master terhadap server slave, replikasi dapat meningkatkan efektifitas biaya proses transaksi, karena dapat mendistribusikan data ke berbagai tempat tanpa harus menginput ulang ke masing-masing server.

Adapun Replikasi basis data dapat juga diartikan sebagai seperangkat teknologi yang digunakan untuk menyalin dan mendistribusikan data dari satu basis data ke basis data yang lain. Dan selanjutnya, mensinkronisasikan antar basis data untuk menjaga konsistensi. Dengan replikasi, data dapat didistribusikan ke lokasi yang berbeda dan pengguna yang jauh melalui LAN, WAN, Dial-up Connection, wireless connections, dan internet.

Dalam mendistribusikan database menggunakan sistem basis data terdistribusi atau distributed database (DDB). Basis data terdistribusi bukan terdiri dari sekumpulan file yang berada pada berbagai tempat tetapi pada sebuah basis data diberbagai tempat. Setiap tempat secara mandiri memproses permintaan user yang membutuhkan akses ke data ditempat tersebut dan juga mampu untuk memproses data yang tersimpan ditempat lain.

Permasalahan

Dalam system basis data tunggal semua pengguna menggunakan server basis data yang sama, pada system ini kemungkinan kegagalan system sangat rentan sehingga apabila terjadi kerusakan pada basis data, user tidak dapat menggunakan system tersebut. Dengan demikian dapat diidentifikasi permasalahannya, yaitu :

1. Apabila terjadi kerusakan pada basis data, user tidak dapat menggunakan system tersebut.
2. Dibutuhkan data cadangan untuk back up basis data

Berdasarkan masalah – masalah diatas permasalahannya dapat dirumuskan, yaitu :

1. Replikasi data Asynchronous merupakan salah satu teknologi yang diduga dapat dimanfaatkan untuk meningkatkan kinerja pembacaan data.
2. Kemampuan Penerapan Replikasi Asynchronous dapat menjadi alternatif untuk keamanan data dalam basis data

Maksud dan Tujuan Penelitian

Maksud penelitian :

Menerapkan replikasi asynchronous pada basis data terdistribusi untuk keamanan data di Ole Suites Sentul.

Tujuan Penelitian :

1. Melakukan pencegahan adanya kerusakan basis data sehingga basis data dapat ter back up secara real time.
2. Membuat replikasi data sebagai data cadangan.

Landasan Teori

Keamanan Data

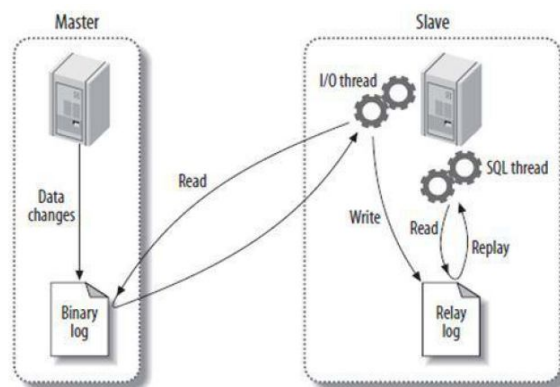
Masalah keamanan merupakan salah satu aspek penting dari sebuah sistem. Sayangnya sekali masalah keamanan ini sering kali kurang mendapat perhatian dari para pemilik dan pengelola sistem informasi. Seringkali masalah keamanan berada di urutan kedua, atau bahkan di urutan terakhir dalam daftar hal-hal yang dianggap penting. Apabila mengganggu performansi dari sistem, seringkali keamanan dikurangi atau ditiadakan (Rahardjo (2005: 1)). Keamanan itu tidak dapat muncul demikian saja. Dia harus direncanakan. Ambil contoh berikut. Jika kita membangun sebuah rumah, maka pintu rumah kita harus dilengkapi dengan kunci pintu. Jika kita terlupa memasukkan kunci pintu pada budget perencanaan rumah, maka kita akan dikagetkan bahwa ternyata harus keluar dana untuk menjaga keamanan.

Replikasi

Replikasi adalah system memelihara sejumlah salinan/duplikat table-tabel data. Setiap salinan tersimpan dalam simpul yang berbeda, yang menghasilkan replikasi data. (Fathansyah, 2012, h.339).

Jika table r direplikasi, salinan dari table tersebut disimpan dalam dua atau lebih simpul. Pada kasus yang paling ekstrim, kita dapat memiliki replikasi penuh (full replication), jika salinan dari table r tersebut disimpan disemua simpul yang ada didalam system. Keuntungan dari replikasi data antara lain : Ketersediaan data yang tinggi (Availability), Peningkatan proses parallel (Increased parallelism) Secara umum, Replikasi akan memperbaiki performansi dari operasi query (pembacaan data) dan meningkatkan ketersediaan data khususnya untuk transaksi-transaksi pembacaan saja (read only). Sebaliknya, transaksi perubahan data akan berlangsung lebih lama dan sukar.

Replikasi data asynchronous : Proses replikasi terjadi setelah transaksi di master selesai. Pertukaran data secara buffering, data akan diletakkan dalam sebuah buffer terlebih dahulu, kemudian pada jangka waktu tertentu akan direplikasi ke disk slave. Tidak menjamin kesinkronan data apabila salah satu site/node mengalami crash saat replikasi belum selesai dilaksanakan. Gambar 1 menggambarkan replikasi lebih terinci.



Gambar 1. Arsitektur Replikasi Asynchronous (Sumber : Fathansyah 2012)

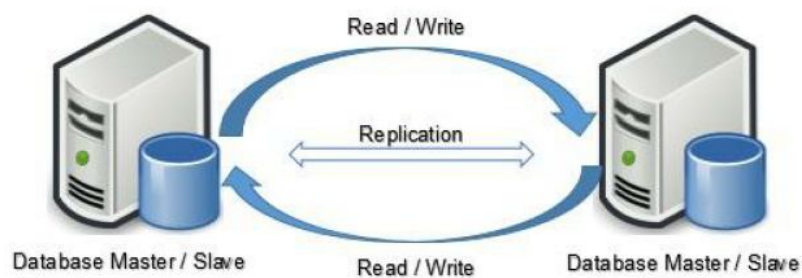
Bagian pertama dari proses ini adalah penulisan binary log di master. Tepat sebelum setiap transaksi terbaru melengkapi data pada master, master mencatat perubahan dalam binary log nya. MySQL menulis transaksi secara berurutan dalam binary log ini, bahkan jika laporan dalam transaksi tersebut terselip selama proses terjadi. Setelah menulis events pada binary log, master kembali memberitahu mesin penyimpanan untuk melakukan transaksi. Langkah berikutnya adalah bagi slave untuk menyalin binary log master ke hard-drivanya, yang biasa disebut relay log. Di mulai dengan susunan pekerja, yang disebut thread slave I/O. Thread I/O membuka koneksi klien agar bisa terhubung ke master, kemudian mulai proses khusus binlog dump. Proses binlog dump membaca transaksi dari binary log master. Jika sudah mengambil semua data di master, thread ini diam dan menunggu master memberi sinyal lagi ketika ada events baru. Lalu thread I/O ini mencatat events ke relay log slave. Slave SQL thread menangani bagian terakhir dari proses. Thread ini membaca dan merespon events dari relay log, sehingga memperbarui data slave agar sesuai dengan master. Selama thread ini terus sesuai dengan thread I/O, relay log biasanya tetap dalam bagian cache sistem operasi, sehingga relay log memiliki overhead yang sangat rendah.

Replikasi Asynchronous

Replikasi asynchronous adalah replikasi yang dimana server master tidak harus terkoneksi dengan server slave. Pada saat server master tidak terkoneksi data yang tersimpan pada server master akan terjadi buffer terhadap server slave sehingga saat server master terkoneksi dengan server slave data akan otomatis tersimpan di server slave.

Replikasi Multi Master

Replikasi multi master merupakan sebuah metode replikasi basis data yang memungkinkan data untuk disimpan di sejumlah sitedan di update dari semua site. Jadi semua site dapat melakukan operasi pembacaan (read) maupun penulisan (write). Perbedaan yang mendasar bentuk replikasi two-way dan one-way adalah kemampuan operasi penulisan, dimana pada one-way hanya dilakukan di server master. Pada bentuk two-way, setiap site akan berperan sebagai master dan slave pada masing-masing site, replikasi two-way dapat dilihat pada gambar 2.



Gambar 2 Replikasi Two Way

Dari gambar 2 masing-masing server berfungsi sebagai server master dan slave, setiap server melakukan replikasi basis data, jadi untuk melakukan proses update dan insert dapat dilakukan di masing-masing server.

Metode Penelitian

Penelitian ini dilakukan secara analitis serta berdasarkan masalah yang terjadi di lapangan, dengan tujuan dasar membuktikan teori yaitu menerapkan replikasi asynchronous pada basis data terdistribusi untuk keamanan data di hotel.

Penelitian ini menggunakan metode penelitian kuantitatif adapun jenis penelitian kuantitatif yang digunakan adalah penelitian eksperimen. Penelitian Eksperimen yang bertujuan untuk menyelidiki kemungkinan hubungan sebab akibat dengan melakukan kontrol/kendali.

Sumber Data

Data menjadi bagian penting dalam penelitian proses pengolahan data. Sumber data dalam penelitian ini merupakan data primer yaitu data yang diambil pada Ole Suites Sentul dengan data yang digunakan adalah data tamu yang menginap tahun 2014.

Instrumen Penelitian

Instrumen penelitian yang digunakan dalam penelitian ini berupa data jumlah tamu tahun 2014, serta menggunakan aplikasi database MySQL sebagai tools untuk mengolah data tersebut.

Pengumpulan Data

Teknik pengumpulan data pada penelitian ini adalah :

1. Mengambil data yang didapatkan pada Ole Suites Sentul.
2. Dokumentasi : upaya untuk mengumpulkan data yang akurat tentang keperluan proses penelitian yang sesuai dengan data tamu hotel di Indonesia

Pengujian

Proses analisis data dengan menggunakan Replikasi Asynchronous terhadap data primer yang dihasilkan dari instrumen penelitian ini adalah menggunakan trigger. Trigger adalah objek didalam basis data yang berasosiasi dengan suatu table. Trigger akan diaktifkan ketika table tersebut dikenai event tertentu. Event yang dimaksud disini adalah suatu kejadian yang menimpa table, bisa berupa penambahan, perubahan, maupun penghapusan data. Dalam pengelolaan basis data, trigger memiliki beberapa manfaat, yaitu :

1. Trigger dapat menguji validitas data yang akan dimasukkan maupun yang akan digunakan untuk melakukan perubahan
2. Trigger dapat memperoleh nilai lama dari baris data yang dihapus atau diubah (misalnya untuk keperluan pencatatan histori data dari suatu table)

3. Trigger dapat mengubah nilai kolom pada table lain
4. Trigger dapat meringankan proses logik dari aplikasi atau stored procedure yang di kembangkan. Alasannya adalah karena proses validasi data sudah dikerjakan oleh trigger sehingga dalam aplikasi kita tidak perlu menuliskan kode program untuk melakukan hal tersebut.

Untuk menguji proses trigger menggunakan Black box dan white box.

Black box adalah pengujian yang dilakukan untuk mengevaluasi pemenuhan system atau komponen dengan kebutuhan fungsional tertentu. Kebenaran pengujian dilihat dari keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diketahui kesalahannya.

White box adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur control dari desain program secara procedural untuk membagi pengujian kedalam beberapa kasus pengujian.

Pembahasan

Pengolahan Data

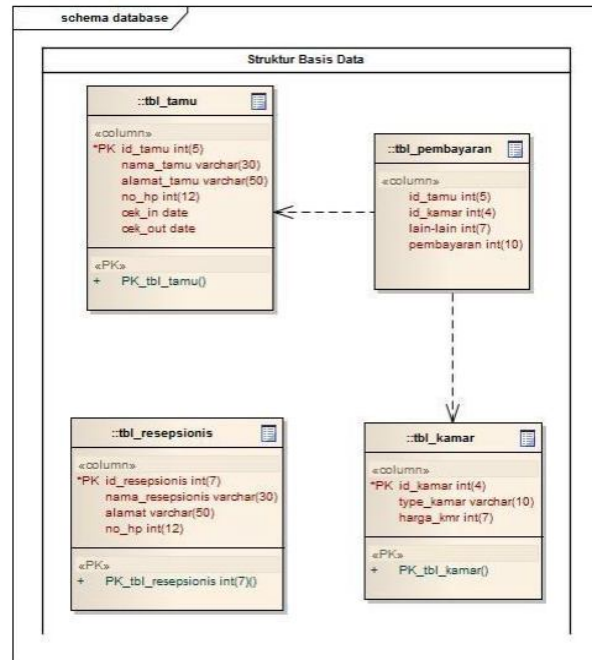
Data yang diperoleh adalah data tamu yang menginap di hotel Ole Suites tahun 2014. Berikut contoh form tamu yang menginap seperti pada gambar 3. Data di peroleh dari hotel Ole Suites Sentul, data dikelola oleh departemen Front Office. Dari form gambar 3 kita dapat membuat intitas untuk menyusun basis data yang dibutuhkan, antara lain Resepsionis, Tamu, Kamar, Pembayaran.

CHECK IN / OUT LIST									
No : 001/SLDC/VIII/14 Ref: Reservasi FIT No 001/SLDC/VIII/14									
Receptionist : Ita Kurniasari Check in - Date : 01 Agustus 2014 - Time : 16.00	Room No : B-210 & B-211 Check Out - Date : 04 Agustus 2014 - Time : 12.00								
Guest Name : Bapak. Salamsyah Address : Jl. Kalibata Timur I Amount of payment : Rp. 425,000,- x 2 Kamar x 3 Malam	Rp. 2.550,000								
Dp Reservasi Pelunasan	Rp. 2.550,000								
Form Distribution : (1) Guest (2) Receptionist	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 50%; text-align: center;">Prepared by</td> <td style="width: 50%; text-align: center;">Approved by</td> </tr> <tr> <td style="text-align: center;">Date : 01/08/14</td> <td style="text-align: center;">Date : 01/08/14</td> </tr> <tr> <td style="text-align: center; height: 30px;">Ita</td> <td style="text-align: center; height: 30px;">Guest</td> </tr> <tr> <td style="text-align: center; font-size: x-small;">Reception</td> <td style="text-align: center; font-size: x-small;">Guest</td> </tr> </table>	Prepared by	Approved by	Date : 01/08/14	Date : 01/08/14	Ita	Guest	Reception	Guest
Prepared by	Approved by								
Date : 01/08/14	Date : 01/08/14								
Ita	Guest								
Reception	Guest								
Raya Babakan Madang No. 99 Sentul Selatan - Bogor 16810 P hone : (021) 8795 0061, (021) 87951258 Fax : (021) 8795126									

Gambar 3. Form Pembayaran

Basis Data yang Sedang Berjalan

Basis data yang sedang berjalan masih menggunakan basis data server tunggal, strukturnya seperti pada gambar 3

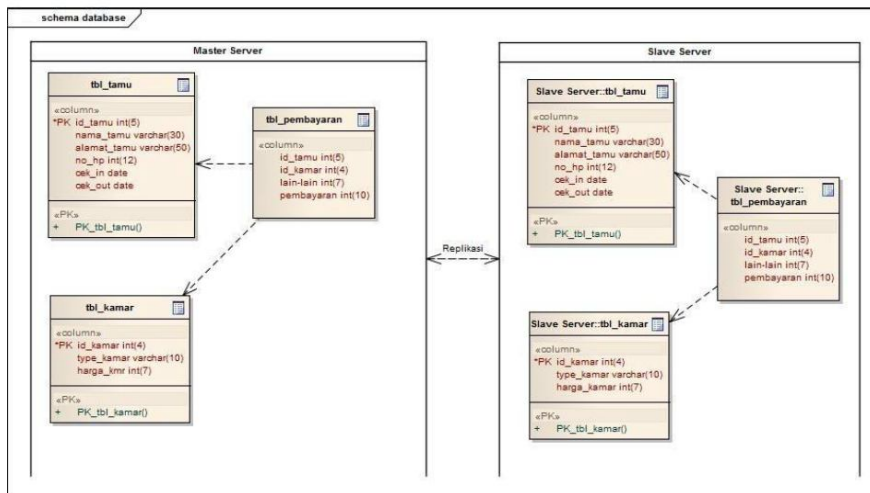


Gambar 4. Struktur Basis Data Lama

Basis Data Baru

Basis data yang baru dikembangkan dengan replikasi asynchronous, untuk melakukan replikasi dibutuhkan dua computer, maka struktur basis data seperti ditunjukkan pada gambar 5..

Dari gambar 5 terdapat dua server yaitu server master sebagai server utama dan server slave sebagai replikasi dari server master. Pada server master terdapat 3 tabel yaitu tbl_kamar, tbl_tamu, dan tbl_pembayaran. Sedangkan pada server slave terdapat 3 tabel yaitu tbl_kamar, tbl_tamu, dan tbl_pembayaran. Untuk operator basis data dilakukan oleh resepsionis dan proses input data dilakukan pada server slave sehingga server slave di batasi aksesnya yaitu hanya untuk input tbl_tamu, tbl_kamar dan tbl_pembayaran.



Gambar 5 Struktur Basis Data Baru

Replikasi Asynchronous

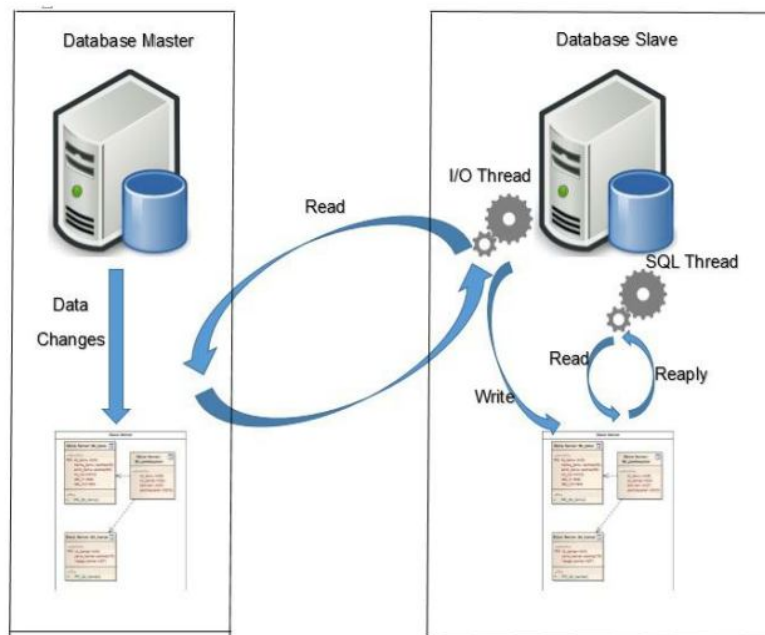
Replikasi asynchronous adalah replikasi yang dimana server master tidak harus terkoneksi dengan server slave. Pada saat server master tidak terkoneksi data yang tersimpan pada server master akan terjadi buffer terhadap server slave sehingga saat server master terkoneksi dengan server slave data akan otomatis tersimpan di server slave.

Kegunaan dari proses replikasi ini adalah bila suatu database satu mengalami kerusakan maka secara otomatis semua data yang telah disimpan pada

database satu dapat dilihat kembali di database lain. Proses backup dan restore kurang praktis, maka konsep inilah solusi keefektifan dalam menjawab kekurangan praktisan kedua proses di atas. Konsep ini merupakan konsep teknologi yang unik, karena secara otomatis segala perintah yang ada pada database satu akan tersimpan pada database lain sehingga bila kehilangan data pada database satu, data masih tersimpan rapi di database lain.

Ada tiga bagian dasar dari cara kerja replikasi di MySQL secara umum:

1. Master mencatat perubahan data dalam basis data
2. Slave menyalin aktivitas basis data master ke basis data slave.
3. Slave merespon proses di basis data dan kemudian mengubah data pada basis datanya sendiri. Bagian dasar ini hanya gambaran dari beberapa langkah proses replikasi dari bagian yang cukup kompleks. Gambar 6 menggambarkan replikasi lebih terinci.



Gambar 6 Replikasi Asynchronous

Pada database master terjadi proses perubahan data (insert,update) basis data di master. Tepat sebelum setiap transaksi terbaru melengkapi data pada master, master mencatat perubahan dalam basis datanya. MySQL menulis transaksi secara berurutan dalam basis data ini, bahkan jika laporan dalam transaksi tersebut terselip selama proses terjadi. Setelah

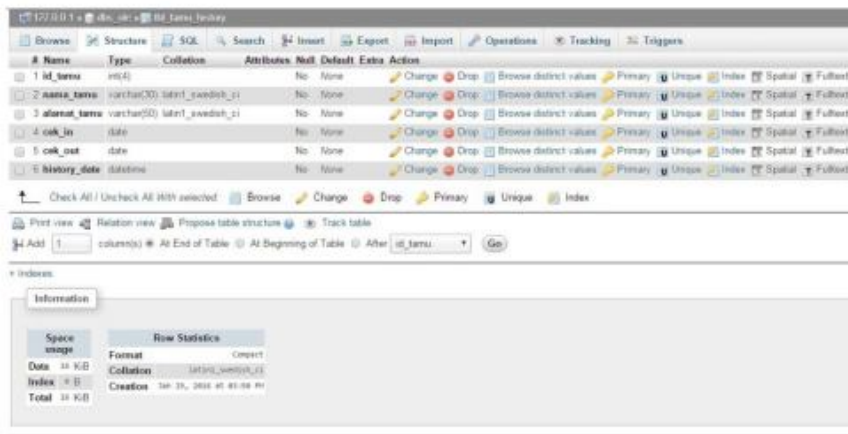
menulis events pada basis data, master kembali memberitahu mesin penyimpanan untuk melakukan transaksi.

Kemudian pada bagian database slave untuk menyalin basis data master ke hard-drivanya, yang biasa disebut relay log. Di mulai dengan susunan pekerja, yang disebut thread slave I/O. Thread I/O membuka koneksi klien agar bisa terhubung ke master, kemudian mulai proses khusus binlog dump. Proses binlog dump membaca transaksi dari basis data master. Jika sudah mengambil semua data di master, thread ini diam dan menunggu master memberi sinyal lagi ketika ada events baru. Lalu thread I/O ini mencatat events ke relay log slave.

Slave SQL thread menangani bagian terakhir dari proses. Thread ini membaca dan merespon events dari relay log, sehingga memperbarui data slave agar sesuai dengan master. Selama thread ini terus sesuai dengan thread I/O, relay log biasanya tetap dalam bagian cache sistem operasi, sehingga relay log memiliki overhead yang sangat rendah.

Pengujian

Pengujian dilakukan dengan membuat fungsi trigger pada database master dan slave, trigger digunakan untuk menampilkan waktu atau history pada saat data di entry, sebelum membuat trigger ,buat table baru pada database , dengan nama tbl_tamu_history, buat fieldnya sama dengan tbl_tamu kemudian tambah field history_date. Struktur table dapat di lihat digambar



Gambar 7. Struktur Tbl_Tamu_Histori

Untuk mendapat mengisi field history_date dibuat perintah trigger berikut sintak trigger nya dapat dilihat pada gambar

```
CREATE
/*!50017 DEFINER = 'root'@'localhost' */
TRIGGER `insert_tamu` AFTER INSERT ON `tbl_tamu`
FOR EACH ROW BEGIN
insert into tbl_tamu_history values(new.id_tamu,new.nama_tamu,new.alamat_tamu,new.cek_in,new.cek_out,now());
END;
..
```

Gambar 8 Trigger Insert Tbl_tamu

Setelah trigger sudah dibuat kemudian entry data pada tbl_tamu, data yang dientry sebanyak 10 data dan entry pada waktu yang sama. Untuk melihat history_date buka tbl_tamu_history, hasilnya dapat dilihat pada table Perbandingan waktu antara database master dan database slave dengan melihat waktu pada saat entry data, hasilnya dapat dilihat pada tabel 4

Tabel 1 Tbl_Tamu_Histori

id_tamu	nama_tamu	alamat_tamu	cek_in	cek_out	history_date
27	Bpk Bolton Siregar	Jl. Nakula VII c/234.Pasar Rebo.Jakarta Timur	2014-03-29	2014-03-30	2016-01-29 09:07:24
28	Bpk Buzan (MOD)	bogor	2014-03-01	2014-03-02	2016-01-29 09:07:24
29	Bpk. JANSEN Musa M T	Jl. BALAI PUSTAKA V NO.7 RT 011/010 RAWAMANGUN JKT	2014-03-01	2014-03-02	2016-01-29 09:07:24
30	Bpk Yayan M (SOSRO)	Jl Pelindung HewanNo 9 Astana Anyar Bambang	2014-03-04	2014-03-08	2016-01-29 09:07:24
31	Bpk Hanif (PT. METRASAT)	Gg Kerkop Kutosari Kebumen Jawa Tengah	2014-03-05	2014-03-06	2016-01-29 09:07:24
32	Bpk Eddy Shambada	Komp Sbs Block c 25 Harapan Jaya Bekasi	2014-03-20	2014-03-21	2016-01-29 09:07:24
33	Bpk Machfid	Teluk Betung 1/12. Rt 06/05. Surabaya	2014-03-27	2014-03-29	2016-01-29 09:07:24
34	Kel Mas Yuda (Tina Wasitoning	Sugihan Rt 016/04.Tengaran. Semarang	2014-03-29	2014-03-30	2016-01-29 09:07:24
35	Ibu Yani	Jl Karet Ps Baru Timur V Krt Tengsin Tanah Abang	2014-03-29	2014-03-30	2016-01-29 09:07:24
36	Ibu Dessy Hermilasari	Jl. Nakula VII c/234	2014-03-29	2014-03-30	2016-01-29 09:07:24

Tabel 2 Perbandingan Waktu Antar Database

No	Entry data (id_tamu)	Perbandingan waktu antar database	
		Master	Slave
1	27	1/29/2016 9:07:24	1/29/2016 9:07:24
2	28	1/29/2016 9:07:24	1/29/2016 9:07:24
3	29	1/29/2016 9:07:24	1/29/2016 9:07:24
4	30	1/29/2016 9:07:24	1/29/2016 9:07:24
5	31	1/29/2016 9:07:24	1/29/2016 9:07:24
6	32	1/29/2016 9:07:24	1/29/2016 9:07:24
7	33	1/29/2016 9:07:24	1/29/2016 9:07:24
8	34	1/29/2016 9:07:24	1/29/2016 9:07:24
9	35	1/29/2016 9:07:24	1/29/2016 9:07:24
10	36	1/29/2016 9:07:24	1/29/2016 9:07:24

Dari table 4 waktu yang ditampilkan antara database master dan slave tidak ada perbedaan waktu, jadi waktu yang dibutuhkan oleh database slave saat entry data tidak mengalami waktu tunda (delay time) sehingga proses insert dan update database di lakukan secara real time

1. Pengujian Black Box

Pengujian black box menggunakan metode Equivalence Partitioning, metode ini merupakan test case yang ideal mengungkapkan kelas kesalahan, karena pada teknik ini berusaha mengungkapkan kelas-kelas kesalahan sehingga mengurangi jumlah total test case yang harus dikembangkan. Metode ini membagi domain input dari suatu program kedalam kelas - kelas data sehingga test case dapat diperoleh. Kelas data yang terbentuk disajikan sebagai kondisi input dalam kasus uji. Kelas merupakan himpunan nilai-nilai yang valid dan tidak valid. Desain test case partisi ekivalensi didasarkan pada evaluasi terhadap kelas ekivalensi untuk suatu kondisi input.

- a. Sebagai input data dari table tbl_tamu adalah :
 - Id_tamu : auto increment \leq 4 digit integer
 - Nama_tamu : \leq 30 digit varchar
 - Alamat_tamu : \leq 50 digit varchar
 - Cek_in : date
 - Cek_out : date
- b. Kondisi input yang sesuai dengan masing-masing elemen data untuk table tbl_tamu adalah :
 - Id_tamu : kondisi input range (\leq 4 digit integer)
 - Nama_tamu : kondisi input himpunan (\leq 30 digit varchar)
 - Alamat_tamu : kondisi input himpunan (\leq 50 digit varchar)
 - Cek_in : kondisi input range (date)
 - Cek_out : kondisi input range (date)
- c. Data test case yang didesain adalah :
 - Id_tamu (kondisi input : range)
 - Valid (0000,1111,1234,9847, 7654)
 - Invalid (00000,11111,12345,97358, 74653)
 - Invalid (000000, 111111, 123456, 987467)
 - Nama_tamu (kondisi input : himpunan)
 - Valid ('agus', 'juedi', 'rendi', 'hasan', 'anwar')
 - Invalid (> 30 digit)
 - Invalid (0 digit)
 - Alamat_tamu (kondisi input : himpunan)
 - Valid ('bogor', 'jakarta', 'depok', 'bekasi', 'parung')
 - Invalid (> 30 digit)
 - Invalid (tidak diisi)
 - Cek_in (kondisi input : range)
 - Valid (2014-01-02, 2014-02-03, 2015-03-10, 2016-04-20, 2016-10-02)
 - Invalid (02-10-2014, 02-02-2015, 10-11-2016, 07-03-2013, 14-12-2010)
 - Invalid (02/02/2015, 10/11/2016, 03/07/2013, 04/12/2015, 03/08/2015)
 - Cek_out (kondisi input : range)
 - Valid (2015-01-02, 2010-02-03, 2016-03-10, 2011-04-20, 2012-10-02)
 - Invalid (02-11-2014, 02-12-2015, 11-11-2016, 17-03-2013, 14-02-2010)
 - Invalid (01/02/2015, 12/11/2016, 13/07/2013, 24/12/2015, 23/08/2015)

2. Pengujian White Box

Pengujian white box (glass box) adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian. Penentuan

kasus uji disesuaikan dengan struktur system, pengetahuan mengenai program digunakan untuk mengidentifikasi kasus uji tambahan.

Tujuan penggunaan white box untuk menguji semua statement program. Penggunaan metode pengujian white box dilakukan untuk :

- a. memberikan jaminan bahwa semua jalur independen suatu modul digunakan minimal satu kali
- b. menggunakan semua keputusan logis untuk semua kondisi true atau false
- c. mengeksekusi semua perulangan pada batasan nilai dan operasional pada setiap kondisi.
- d. menggunakan struktur data internal untuk menjamin validitas jalur keputusan.

Kesimpulan

Berdasarkan hasil penelitian dan uraian pada bab IV sebelumnya. Hasil analisa penerapan replikasi asynchronous pada basis data terdistribusi di Ole Suites Sentul maka kesimpulannya sebagai berikut :

1. Replikasi asynchronous dapat mencegah adanya kerusakan basis data sehingga basis data dapat ter back up secara real time dan waktu entry data dari slave ke master tidak mengalami perubahan ataupun delay time.
2. Dengan adanya replikasi basis data jadi basis data ada dua yang bekerja, yang berfungsi sebagai basis data cadangan.

Saran

Adapun saran penyusun setelah melakukan penelitian antara lain :

1. Penelitian lebih lanjut diharapkan dapat menerapkan replikasi asynchronous dengan metode-metode yang lain dengan data yang berbeda dan juga mesin database yang lain. Replikasi asynchronous diharapkan dapat dikembangkan dengan mesin database yang berbeda antara server master dan server slave, sehingga dapat berfungsi multi platform.
2. Penelitian ini sebaiknya dapat dikembangkan dengan basis data yang lebih besar dan dapat ditambahkan fungsi-fungsi dari fitur mesin database.

Daftar Pustaka

- Al Karima . (2013). Analisis dan Perancangan Basis Data Terdistribusi Pengolahan Data Polis Asuransi Menggunakan Metode Replikasi Asynchronous Pada PT Videi Kantor Cabang Palembang. Universitas Bina Darma, hall 1-29
- Belluano, Poetri Lestari Lokapitasari. (2014). Sistem Replikasi Pangkalan Data Pendidikan Tinggi (PDPT) Terintegrasi dengan Teknik IP-Streaming. Universitas Hasanuddin Makassar
- Fathansyah. 2012. Basis Data Edisi Revisi. Informatika, Bandung

- Indrajani. 2014. Database System Case Study All In One. PT Elex Media Komputindo, Jakarta
- Mubarak, Abdul. (2012). Sistem Replikasi Basis Data Terdistribusi Untuk Data Center, Universitas Indonesia Timur Makassar
- Nurwenda, Teni Ardiani. (2010). Aplikasi Pengolahan Data Kepegawaian Berbasis Web dengan Database Terdistribusi Menggunakan Metode Replikasi Asynchronous Pada PT Taspen (Persero). UNIKOM, hall 1-9
- Rahardjo, Budi. 1998-2002. Keamanan Sistem Informasi Berbasis Internet, PT. Insan Infonesia & PT. Indocisc, Bandung & Jakarta.
- Rahardjo, Budi. 2011. Belajar Otodidak Membuat Database Menggunakan MySQL, Informatika Bandung .
- Silitonga, Parasian D.P. (2014). Replikasi Basis Data Pada Sistem Pengolahan Data Akademik Univeristas Katolik Santo Thomas, Jurnal Time, Vol. III No 1, Univeristas Katolik Santo Thomas, hal 32-36
- Sutanta, Edhy. 2011. Basis Data dalam Tinjauan Konseptual. CV Andi, Yogyakarta
- Triyono, Joko. (2012) Replikasi untuk meningkatkan kinerja dan ketersediaan data (study kasus system informasi akademik). Jurnal Teknologi Technoscintia, Vol. 5 No 1, Institut Sains & Teknologi AKPRIND, hall 31-40