

REKAYASA DATABASE TERDISTRIBUSI PADA LAYANAN PEMESANAN TIKET PESAWAT TERBANG

Febri Nova Lenti¹

¹STMIK AKAKOM Yogyakarta

Masuk: 3 Oktober 2013, revisi masuk: 12 Januari 2014, diterima: 2 Februari 2014

ABSTRACT

Distributed database allows users at each location are dispersed can access data stored in another location. Advantages compared with a web-based database is if a database failure at one location then not will disable all other database operations. In addition, a distributed database has the readability of the data better than the centralized database, for the reason each site only handle a portion of the entire database, resulted in an difference to service the CPU and I/O such as the characteristics of the centralized database. The airlines as an organization that has branches and sales agents scattered in a few places if you want to always maximize service to its customers, it would be very appropriate if you are using a distributed database in a ticket booking transaction services. In this paper will be engineering a distributed database on tickets booking services. Implementation is done by applying a distributed database replication and applications performed using a sequential process model with object-oriented paradigm. Final result of the application of an airplane ticket booking services based distributed database that will improve the performance of the services.

Keywords : *distributed databases, database replication, site, centralized database, the master, slave, UML*

INTISARI

Database terdistribusi memungkinkan pengguna pada masing-masing lokasi yang tersebar dapat mengakses data yang disimpan pada lokasi lain. Kelebihannya dibanding dengan database yang berbasis web adalah kegagalan database pada satu lokasi tidak akan mematikan seluruh operasional database yang lain. Di samping itu database terdistribusi memiliki kecepatan dalam pengaksesan data yang lebih baik dibandingkan dengan database tersentralisasi karena masing-masing site hanya menangani sebagian saja dari keseluruhan database, sehingga beban pelayanan CPU dan I/O lebih rendah. Perusahaan penerbangan sebagai suatu organisasi yang mempunyai cabang dan agen penjualan yang tersebar di beberapa tempat jika ingin memaksimalkan layanan terhadap pelanggannya, maka akan sangat tepat jika menggunakan database terdistribusi dalam layanan transaksi pemesanan tiketnya. Pada makalah ini akan dilakukan rekayasa database terdistribusi pada layanan pemesanan tiket pesawat terbang. Implementasi database terdistribusi dilakukan dengan menerapkan replikasi database dan rekayasa aplikasi dilakukan menggunakan model proses sequential dengan paradigma berorientasi obyek. Hasil akhir berupa aplikasi layanan pemesanan tiket pesawat terbang berbasis database terdistribusi.

Kata kunci: database terdistribusi, replikasi database, site, database tersentralisasi, master, slave, UML.

PENDAHULUAN

Motivasi utama dibelakang dari pada pengembangan sistem database adalah keinginan usaha untuk menyatukan data operasional suatu organisasi dan mengaksesnya secara terkontrol. Selama ini yang paling umum dikem-

bangkan adalah implementasi integrasi dan kontrol data dalam bentuk tersentralisasi (database tersentralisasi). Adanya perkembangan pada jaringan komputer menghasilkan suatu bentuk desentralisa-

¹ febri@akakom.ac.id

si. Pendekatan desentralisasi ini dapat digambarkan sebagai suatu organisasi yang memiliki banyak kantor cabang dimana setiap cabang tersebut mengoperasikan datanya secara sendiri sendiri. Data yang digunakan secara bersama sama dan efisiensi dalam pengaksesan data harus juga diiringi dengan perkembangan dari sistem data base yang merupakan refleksi dari struktur organisasi sehingga data dapat diakses dimana saja dan melakukan penyimpanan data di lokasi yang memang data tersebut paling sering digunakan. Dan database terdistribusi adalah merupakan jawabannya karena database terdistribusi ini dapat mengatasi sekumpulan permasalahan informasi seperti masalah geografi, masalah arsitektur komputer dan masalah protokol komunikasi.

Pada kasus pemesanan tiket pesawat terbang jika layanannya diimplementasi dalam database tersentralisasi maka ada kemungkinan gangguan terjadi pada lamanya proses pelayanan pemesanan maupun pembelian sebuah tiket pesawat. Hal ini dikarenakan database tersentralisasi diakses oleh beberapa agen secara bersamaan, sehingga akan mengakibatkan lamanya proses transaksi.

Kelemahan terhadap database tersentralisasi akan sedikit teratasi jika proses transaksi diproses terhadap database cabang yang kemudian data pada database cabang tersebut akan disinkronkan dengan database pusat, dengan kata lain beban transaksi akan lebih ringan karena transaksi diproses di database lokal bukan di database pusat.

Penelitian sejenis pernah dilakukan oleh Mubarak dkk (Mubarak, 2012) yang meneliti pengembangan sebuah Data Center yang bekerja secara otomatis dengan menggunakan metode Replikasi Basis Data Terdistribusi. Pada penelitian ini server pada sisi *client* melakukan proses pendistribusian data dan replikasi dari basisdata SKPD (Sistem Kepegawaian Pemerintah Daerah) ke basisdata Data Center sehingga dapat membantu pihak pemerintah daerah dalam penyatuan data dan akses data. Darwis (Darwis, 2012) juga melakukan

penelitian mengenai implementasi Basis Data Terdistribusi Menggunakan MySQL pada PT Thamrin Brothers Palembang suatu perusahaan dealer utama Yamaha yang memiliki dealer dealer cabang. Oleh sebab itu disini akan di lakukan rekayasa database terdistribusi yang diimplementasikan pada aplikasi layanan pemesanan tiket pesawat terbang.

METODE

Rekayasa dilakukan dengan dua tahap yaitu analisis dan perancangan aplikasi dan analisis dan perancangan database terdistribusi. Analisis dan perancangan aplikasi dilakukan dengan menggunakan model proses sequential linear dengan paradigma berorientasi obyek. Pressman [Pressman, 1997] menyatakan bahwa: pengembangan perangkat lunak dengan pendekatan sekuensial dilakukan dengan tahapan tahapan analisis, desain, pengkodean, pengujian dan pemeliharaan.

Paradigma berorientasi obyek dalam rekayasa perangkat lunak memandang sistem sebagai kumpulan obyek-obyek diskrit yang saling berinteraksi satu sama lain.

Edward Berard [Berard, 1993] mencatat bahwa manfaat dari teknologi berorientasi obyek akan optimal jika sedari awal dan selama proses rekayasa perangkat lunak diterapkan. Penerapan paradigma berorientasi obyek harus dipertimbangkan terhadap rangkaian proses rekayasa perangkat lunak secara keseluruhan. Jika hanya menggunakan pemrograman berorientasi obyek (OOP) saja tidak akan menghasilkan hasil yang terbaik. Perekayasa perangkat lunak juga harus mempertimbangkan penggunaan beberapa paradigma berorientasi obyek yang lain seperti analisis kebutuhan berorientasi obyek (OORA), desain berorientasi obyek (OOD), analisis domain berorientasi obyek (OODA), sistem data base berorientasi obyek (OODBMS) dan alat bantu/utilitas rekayasa perangkat lunak yang berorientasi obyek (OOCASE).

Oleh karena itu sebelum di implementasikan dalam OOP, sistem akan dianalisis dan didesain menggunakan OORA dan OOD. Bahasa pemodelan yang digunakan adalah UML (*Unified*

Modelling Language). UML adalah bahasa pemodelan yang standar untuk lingkungan berorientasi obyek, yang berisi notasi notasi grafis yang relative sudah dibakukan (*open standard*). Munawar [Munawar, 2005] mengatakan: Paling tidak ada tiga karakter penting yang melekat pada UML, yaitu sketsa, cetak biru dan bahasa pemrograman. Sebagai sebuah sketsa, UML bisa berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dari sistem. UML bisa juga berfungsi sebagai cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bisa diketahui informasi detail tentang *coding program (forward engineering)* atau bahkan membaca program dan menginterpretasikannya kembali ke dalam diagram (*reverse engineering*). Sebagai bahasa pemrograman, UML dapat menterjemahkan diagram yang ada di UML menjadi code program yang siap untuk dijalankan.

Beberapa diagram yang umum ada dalam pemodelan UML adalah sebagai berikut :Diagram Use case, Diagram Sequence dan Diagram Kelas.

PEMBAHASAN

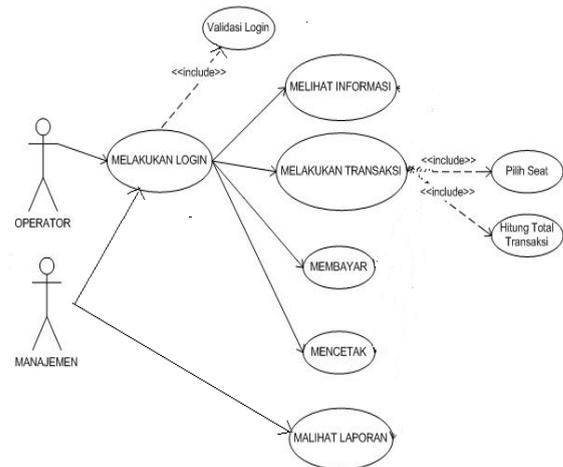
Kebutuhan aplikasi meliputi kebutuhan input, proses dan output sebagai berikut: Kebutuhan input meliputi ; identitas user, data-data tiket, data-data penerbangan, data-data pesawat, data-data pelanggan, data-data transaksi
Kebutuhan proses meliputi ; pengolahan data, transaksi pemesanan, validasi dan proses laporan. Kebutuhan output meliputi ; bukti pemesanan

Perancangan aplikasi dimodelkan menggunakan UML. UML ini mendeskripsikan obyek, proses, aturan dan hubungan antar obyek yang terjadi pada system. Adapun diagram UML yang digunakan adalah Diagram usecase, Diagram Sequence dan Diagram *Class*.

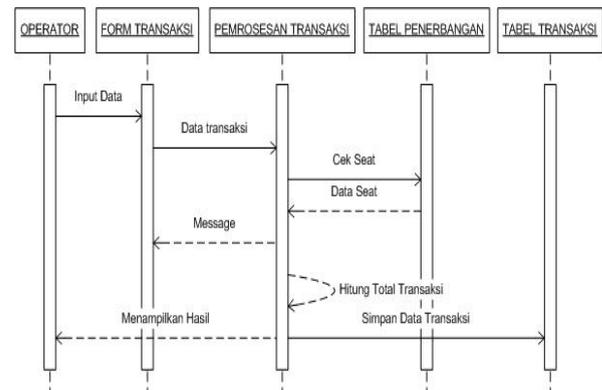
Diagram Use case aplikasi ditunjukkan oleh Gambar 3.1. Pada gambar 3.1 ditunjukkan bagaimana perilaku actor yang terdiri dari operator dan manager pada sistem.

Sequence diagram menggambarkan interaksi antar obyek di dalam dan di sekitar obyek yang digambarkan dalam

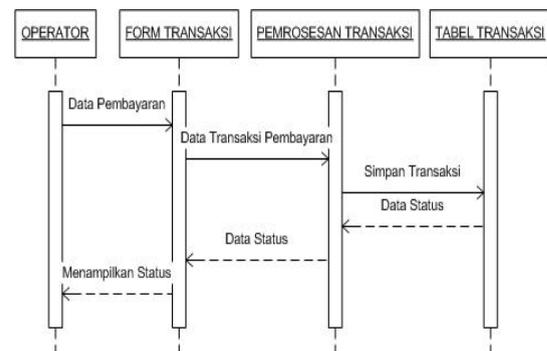
suatu urutan waktu Sequence diagram untuk usecase melakukan transaksi dan usecase membayar ditunjukkan oleh Gambar 3.2 dan. 3.3. Diagram ini juga memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu didalam *Use case*.



Gambar 3.1. Diagram Use case



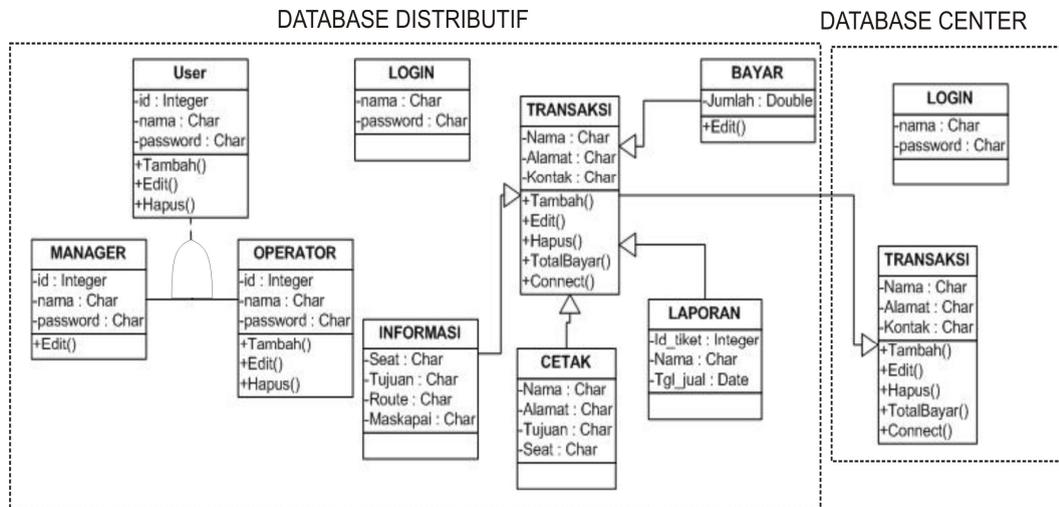
Gambar 3.2 Sequence Diagram untuk use case Melakukan Transaksi



Gambar 3.3. Sequence Diagram untuk use case Membayar

Aplikasi ini memiliki 11 class yang ditunjukkan pada Gambar 3.4. yang juga menggambarkan struktur dan deskripsi class, package dan objek beserta

hubungan satu sama lain seperti pewarisan dan *instant connection*



Gambar 3.4 Class Diagram

Database terdistribusi adalah kumpulan data yang digunakan bersama dan mempunyai hubungan secara logika tersebar secara fisik dalam jaringan komputer. Pengguna mengakses basis data terdistribusi dengan menggunakan dua aplikasi yaitu aplikasi lokal dan aplikasi global.

Ada dua macam teknologi yang di gunakan untuk membangun sebuah database terdistribusi. Yang pertama adalah synchronous distributed database technology dimana data yang terhubung pada suatu jaringan akan selalu ter-update sehingga user pada beberapa tempat dapat mengakses data. Sedangkan teknologi yang lain yaitu asynchronous distributed database technology di mana sistem akan menyediakan suatu *copy* dari replikasi data pada beberapa node sehingga *local server* dapat mengakses data tanpa harus keluar dari jaringan lokal. Metode yang biasa di gunakan yaitu *Replication* serta *circular Replication*.

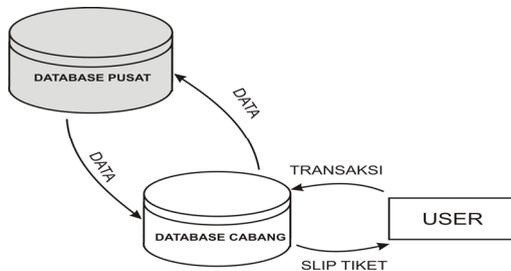
Replikasi adalah suatu teknik untuk melakukan *copy* dan pendistribusian data dan obyek-obyek database dari satu database ke database lain dan melaksanakan sinkronisasi antara database sehingga konsistensi data dapat terjamin. Dengan menggunakan teknik

replikasi ini, data dapat didistribusikan ke lokasi yang berbeda melalui koneksi jaringan lokal maupun internet.

Pada aplikasi ini yang dipakai adalah *asynchronous distributed database technology* dengan metode replikasi, karena: a). *Reliability*. jika sebuah sistem yang menyimpan suatu database mengalami kerusakan maka sistem lain dapat menemukan *copy* dari data-base yang di maksud pada node yang lain yang tidak mengalami kerusakan sistem karena database di simpan pada beberapa node. b). *Fast response*. setiap site mampu-nyai *copy* dari database sehingga proses query data dapat berjalan lebih cepat. c). *Node decoupling*. setiap transaksi data bisa di proses tanpa koordinasi antar jaringan sehingga jika sebuah node sibuk atau mengalami kesalahan maka akan ada sistem tersendiri yang menangani koordinasi antar data. d) *Reduced network traffic at prime time*. kemampuan sistem untuk memindahkan transaksi database pada jaringan yang sedang tidak sibuk atau berfungsi dengan tidak sebagaimana mestinya.

Replikasi ini diimplementasikan pada DBMS MySQL versi v5.0.24a. Replikasi database adalah suatu teknik melakukan *copy* dan pendistribusian data

dan obyek-obyek database dari satu database ke database lain dan melaksanakan sinkronisasi antara database sehingga konsistensi data dapat terjamin.



Gambar 3.5 Pemodelan Replikasi Database

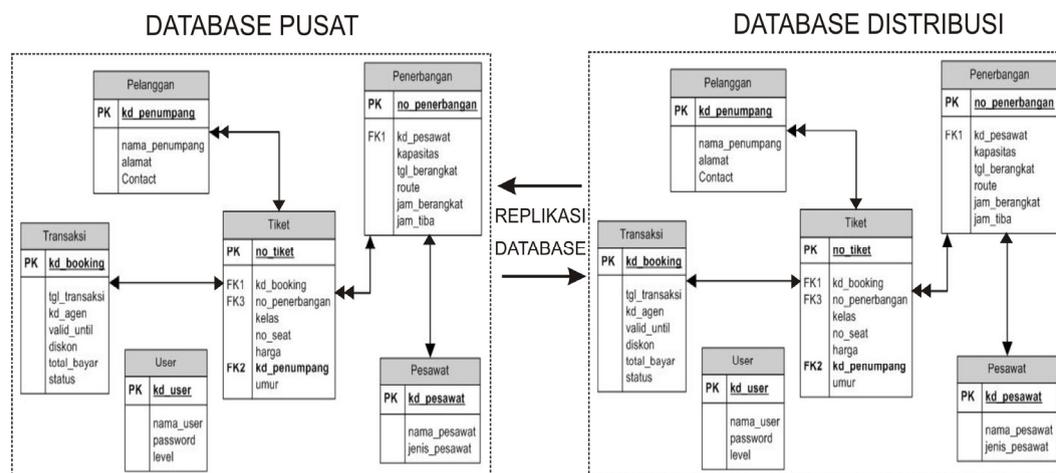
Gambar 3.5 di atas menunjukkan pemodelan replikasi database, dimana terdapat 2 buah database yaitu database pusat dan database cabang. Data dalam database cabang akan mereplikasi atau mengkopikan datanya ke database pusat sehingga kedua database server tersebut akan saling berbagi data. Dan ketika database cabang mengalami down maka database pusat akan tetap melakukan eksekusi terhadap database cabang, dan ketika database cabang sudah kembali normal, maka akan segera mengkopikan semua transaksi yang sudah dieksekusi

di database pusat. itulah maksud dari garis panah bolak-balik pada kedua database. Perancangan Tabel Pusat Dan Distributif.

Perancangan database yang akan diimplementasikan dalam sistem antara database pusat dan database terdistribusi adalah sama dalam artian kedua database tersebut identik. Hal ini dikarenakan dalam perancangan *class* diagram, *class* transaksi mencakup di kedua database, sehingga akan melibatkan semua tabel dalam database distribusi ke dalam database pusat.

Dalam penyusunan suatu program basis data diperlukan adanya kelengkapan komponen yang mendukungnya, antara lain tabel dalam kesatuan database. Rancangan tabel dan relasi tabel baik tabel pusat maupun tabel terdistribusi ditunjukkan pada Gambar 3.6

Proses Replikasi Database, proses ini adalah proses dimana setting antara 2 database server yang berbeda sehingga akan terdapat proses replikasi atau pengkopian dan pengelolaan obyek-obyek dari database yang membentuk suatu sistem database terdistribusi.



Gambar 3.6 Relasi antar Tabel Keterangan: PK = Primary Key (Kunci Utama)
FK = Foreign Key (kunci tamu)

Terdapat 2 buah komputer yang mana yang satu berperan sebagai master dan yang lain berperan sebagai slave. Pada komputer master, sisipkan beberapa

setting di *my.ini* pada Mysql pada komputer master (ip address:192.168.0.1):
Server -id: 1; Log-bin: mysql-bin; Log-

```
slave-updates;Replicare-do  
db=tiket_pesawat*
```

Pada komputer slave, sisipkan beberapa setting di my.ini pada mysql pada komputer slave (ip address: 192.168.0.2): Server -id: 2; Log-bin: mysql-bin; Log-slave-updates; Replicare-do-db=tiket_pesawat*

Tambahan setting replicate-do-db boleh disertakan ataupun tidak, jika tidak disertakan, berarti bahwa proses replikasi akan dilakukan kepada seluruh database dalam database server master. Jika menggunakan, maka hanya akan mereplikasi hanya 1 database saja. Jika menginginkan untuk mereplikasi dengan 1 arah, maka pada komputer master perintah replicate-do-db digantikan dengan bin-log-do-db dan diikuti dengan nama tabel. Sedangkan untuk setting komputer slave masih menggunakan perintah yang sama.

Setelah semua setting pada file my.ini selesai, maka selanjutnya adalah setting pada command mysql. Kemudian masuk kedalam mysql command: Memastikan bahwa service mysql berjalan. Dan juga membuat user baru yang akan memberikan akses kepada mesin master dapat berkomunikasi dengan mesin slave. Grant replication slave on *.* to 'admin'@'%' identified by 'admin';

Flush privileges; Dilakukan proses pembacaan status dari log (perlu dicatat untuk nama dari log dan posisi), maka dilakukan perintah sebagai berikut: a). Use tiket_pesawat; Flush tables with read lock; Show master status; Unlock tables;

Karena komputer 1 akan menjadi slave dari komputer 2, maka perlu dilakukan perubahan konfigurasi master yang akan menjadikan database saling terhubung. Untuk perintah yang dilakukan adalah sebagai berikut:

```
Change master to  
Master_host='192.168.0.2',  
Master_user='admin',  
Master_password='admin',  
Master_log_file='mysql-bin.000012',  
Master_log_pos=143;
```

Semua perintah diatas juga berlaku untuk komputer slave, hal itu dilakukan jika menginginkan kedua komputer tersebut terhubung 2 arah. Jika hanya menginginkan hanya 1 arah maka salah satu komputer saja yang disisipkan perintah di atas.

KESIMPULAN

Database terdistribusi dengan menggunakan replikasi database pada MySQL v5.0.24a dapat diimplementasikan dengan baik pada layanan pemesanan tiket pesawat terbang.

Replikasi database akan menyediakan *back up* data pada database lokal sehingga akan menjamin data tetap valid dan konsisten.

DAFTAR PUSTAKA

- Berard, E.V., 1993, *Essays on Object-Oriented Software Engineering*, Addison-Wesley
- Efri Darwis, Merry Agustina, A. .Mutaqin Bakti, M.M., 2012, *Implementasi Basis Data Terdistribusi Menggunakan My SQL Pada PT Thamrin Brothers Palembang*, http://blog.binadarma.ac.id/mutakin/wp-content/uploads/2012/10/08142341_journal_new.pdf
- Munawar, 2005, *Pemodelan Visual Dengan UML*, Graha Ilmu
- Pressman, R.S. 1997, *Software Engineering A Practitioner's Approach*, Fourth Edition, McGraw Hill
- Ramakrishnan, R., 2004, *Sistem Manajemen Database*, Penerbit Andi, Yogyakarta