

**PEMROGRAMAN DATABASE CLIENT SERVER DENGAN
MS-SQL SERVER
(CLIENT SERVER DATABASE PROGRAMMING)**

Oleh: Arief Setiyanto

1. PERMASALAHAN

Permasalahan yang dihadapi manusia adalah semakin banyak urusan yang ditanganinya. Dengan semakin banyak urusannya maka semakin banyak hal yang harus diingat. Pada awalnya orang membuat catatan untuk membantu ingatannya, jalan keluar ini cukup membantu ketika volume catatan (data) yang diingat cukup kecil. Dari model catatan kemudian dicoba data-data yang sejenis mulai dikelompokkan, sehingga terbentuk tabel.

Ketika Komputer mulai dipergunakan orang untuk mengolah data, orang mulai merasakan berbagai kemudahan yang ditawarkannya. Namun semakin hari urusan yang harus ditangani dengan komputer semakin banyak, satu komputer tidak lagi cukup mampu menanganinya. Kemudian mulai digunakan banyak komputer untuk menyelesaikan pekerjaan. Banyaknya komputer yang digunakan akhirnya menimbulkan problem tersebarnya data pada beberapa komputer yang digunakan. Akhirnya orang mengusahakan agar komputer dapat saling berhubungan, mulailah orang menggunakan jaringan komputer. Kemampuan

berkomunikasi ini saja belum dianggap cukup karena pengguna jaringan ingin dapat mengakses data dari yang dia perlukan dari komputernya dengan mudah. Sementara pemilik data juga ingin agar data yang dimiliki hanya dapat diakses oleh orang-orang yang berhak. Oleh karena itu dibutuhkan program yang mengatur lalu lintas data, pengatur siapa yang berhak dan tidak berhak mengakses data dan sebagainya. Disinilah nantinya server database akan berperan, dia akan melayani bagaimana database itu disimpan, keamanan datanya, mendefinisikan user dengan hak-hak yang dimilikinya, mengunci record-record database, menjadwalkan pengaksesan agar tidak terjadi inkonsistensi data karena editing oleh lebih dari satu user pada waktu yang sama, menjaga konsistensi data dan semua urusan yang berhubungan dengan database dalam sebuah jaringan komputer.

2. DATABASE

Database dalam pengertian yang paling sederhana dapat dianalogikan dengan sebuah tabel. Pada awalnya orang mengumpulkan data-data yang saling berhubungan satu dengan lainnya agar lebih mudah dalam menarik kesimpulan dalam bentuk tabel. Sebuah tabel terdiri dari kolom dan baris, dalam setiap kolom terdiri dari data data yang memiliki sifat yang sama, sedangkan dalam setiap baris berisi data yang saling berhubungan satu dengan yang lainnya. Konsep inilah yang diadopsi oleh

database, yaitu tabel yang disimpan dalam komputer. Pengertian kolom pada tabel dimanifestasikan dalam bentuk field dalam database, sedangkan baris dalam tabel dimanifestasikan dalam record dalam database.

Pada perkembangannya untuk lebih efektif pengertian database mulai bergeser, sehingga kini database lebih dikenal sebagai beberapa tabel yang saling berhubungan satu dengan yang lainnya. Walaupun pada dasarnya tabel-tabel tersebut dapat dibuat hanya sebagai satu tabel saja, namun hal tersebut akan membuat database tidak efektif karena banyak terjadi pengulangan penyimpanan data.

Setiap tabel akan memiliki sebuah atau lebih field yang dijadikan kunci, field kunci ini merupakan identitas dari setiap baris(record). Kunci dapat berupa kunci utama (*primary key*), atau kunci asing (*Foreign key*). Kunci utama harus bersifat unik, karena kunci utama inilah yang akan menjadi identitas setiap baris (*record*).

Dalam sebuah rancangan database ada yang disebut sebagai tabel induk, tabel induk ini dapat memiliki beberapa tabel anak. Setiap tabel induk harus memiliki kunci utama yang unik, sedangkan tabel-tabel anak memiliki field yang mereferensi dari kunci yang terdapat pada tabel induk, yang pada tabel anak akan berlaku sebagai kunci asing. Dari pengertian ini akan melahirkan pengertian hubungan tabel *one to many* yang artinya satu data

berkomunikasi ini saja belum dianggap cukup karena pengguna jaringan ingin dapat mengakses data dari yang dia perlukan dari komputernya dengan mudah. Sementara pemilik data juga ingin agar data yang dimiliki hanya dapat diakses oleh orang-orang yang berhak. Oleh karena itu dibutuhkan program yang mengatur lalu lintas data, pengatur siapa yang berhak dan tidak berhak mengakses data dan sebagainya. Disinilah nantinya server database akan berperan, dia akan melayani bagaimana database itu disimpan, keamanan datanya, mendefinisikan user dengan hak-hak yang dimilikinya, mengunci record-record database, menjadwalkan pengaksesan agar tidak terjadi inkonsistensi data karena editing oleh lebih dari satu user pada waktu yang sama, menjaga konsistensi data dan semua urusan yang berhubungan dengan database dalam sebuah jaringan komputer.

2. DATABASE

Database dalam pengertian yang paling sederhana dapat dianalogikan dengan sebuah tabel. Pada awalnya orang mengumpulkan data-data yang saling berhubungan satu dengan lainnya agar lebih mudah dalam menarik kesimpulan dalam bentuk tabel. Sebuah tabel terdiri dari kolom dan baris, dalam setiap kolom terdiri dari data data yang memiliki sifat yang sama, sedangkan dalam setiap baris berisi data yang saling berhubungan satu dengan yang lainnya. Konsep inilah yang diadopsi oleh

database, yaitu tabel yang disimpan dalam komputer. Pengertian kolom pada tabel dimanifestasikan dalam bentuk field dalam database, sedangkan baris dalam tabel dimanifestasikan dalam record dalam database.

Pada perkembangannya untuk lebih efektif pengertian database mulai bergeser, sehingga kini database lebih dikenal sebagai beberapa tabel yang saling berhubungan satu dengan yang lainnya. Walaupun pada dasarnya tabel-tabel tersebut dapat dibuat hanya sebagai satu tabel saja, namun hal tersebut akan membuat database tidak efektif karena banyak terjadi pengulangan penyimpanan data.

Setiap tabel akan memiliki sebuah atau lebih field yang dijadikan kunci, field kunci ini merupakan identitas dari setiap baris(record). Kunci dapat berupa kunci utama (*primary key*), atau kunci asing (*Foreign key*). Kunci utama harus bersifat unik, karena kunci utama inilah yang akan menjadi identitas setiap baris (*record*).

Dalam sebuah rancangan database ada yang disebut sebagai tabel induk, tabel induk ini dapat memiliki beberapa tabel anak. Setiap tabel induk harus memiliki kunci utama yang unik, sedangkan tabel-tabel anak memiliki field yang mereferensi dari kunci yang terdapat pada tabel induk, yang pada tabel anak akan berlaku sebagai kunci asing. Dari pengertian ini akan melahirkan pengertian hubungan tabel *one to many* yang artinya satu data

pada tabel induk yang bersifat unik berhubungan dengan beberapa data (record) pada tabel anak. Hubungan *many to many* yang artinya beberapa data pada tabel induk berhubungan dengan beberapa data (record) pada tabel anak dan sebaliknya. Dalam hubungan seperti ini harus diperhatikan bahwa, data pada tabel anak harus memperhatikan referensi pada tabel induknya. Misalnya jika ada tabel pelanggan yang memiliki kunci utama kode pelanggan, dan memiliki tabel anak misal tabel pembelian, yang memiliki foreign key kode pembeli, maka tidak boleh ada kode pelanggan yang belum tercatat di tabel pelanggan tetapi sudah tercatat di tabel pembelian. Inilah yang disebut referential integrity. Demikian juga jika pada tabel pelanggan sebuah kode pelanggan dihapus maka semua kode pelanggan yang sama pada tabel-tabel anak harus juga dihapus. (*special edition using visual basic 4.0*)

Selain harus mempertahankan integritas data, sebuah file database juga harus mempertahankan validitas data. Validitas data pada tingkat field akan mengatur nilai-nilai yang boleh dan tidak boleh pada sebuah field. Misalnya untuk tanggal tidak boleh lebih dari 31 dan kurang dari 0.

3. SEJARAH DAN PERKEMBANGAN ARSITEKTUR DATABASE JARINGAN

Seiring perkembangan dunia komputer, maka cara pengelolaan database mengalami perkembangan yang sama cepatnya. Ketika pada awalnya penggunaan komputer lebih terfokus pada penggunaan stand alone (single user), seiring dengan kemampuan komunikasi yang dimiliki komputer, maka pengelolaan database ini-pun menuju ke penggunaan multiuser. Database jaringan memiliki keuntungan pada kemampuan berbagi data antar komputer, sehingga data yang pernah dimasukkan dalam sebuah komputer dapat diakses, maupun dimanipulasi oleh komputer yang terhubung dalam jaringan komputer tersebut. Berikut ini arsitektur-arsitektur komputasi yang pernah dikembangkan dalam pengelolaan database.

1. HOST BASED COMPUTING

Arsitektur host based merupakan awal dari perkembangan arsitektur pemrosesan dalam jaringan. Bentuk arsitektur ini menempatkan sebuah komputer dengan kemampuan komputasi yang sangat tinggi sebagai host, kemudian di hubungkan dengan komputer-komputer terminal yang akan meminta pelayanan sepenuhnya dari host. Komputer-komputer terminal ini hanya bertindak sebagai

tempat melakukan input dan output data saja, tanpa melakukan pemrosesan apapun.

Keuntungan dari arsitektur seperti ini adalah kemudahan dalam pemeliharaan data maupun program, karena semuanya hanya terletak pada satu tempat yaitu di komputer host. Kerugiannya tentunya adalah pada penyediaan host yang harus memiliki kemampuan super yang tentunya harganya akan menjadi sangat mahal. Arsitektur seperti ini digunakan untuk pengolahan database menggunakan super komputer pada sekitar 20 tahun yang lalu. (*Microsoft SQL 6,5 Unleashed*).

2. PEER TO PEER

Arsitektur Peer to peer merupakan arsitektur terdistribusi di mana semua terminal yang terhubung dalam jaringan memiliki kemampuan pemrosesan data yang setara. Pada arsitektur ini semua terminal dapat berfungsi sebagai server dan pada saat yang lain bertindak sebagai yang dilayani (requester/client). Pada arsitektur ini hardware yang dibutuhkan tidak perlu terlalu bagus seperti yang digunakan sebagai host pada arsitektur host based, namun juga harus lebih baik dari terminal yang digunakan pada arsitektur host based. (*Client/Server Architecture*)

Data dan aplikasi pada arsitektur ini tersebar pada semua bagian jaringan. Hal ini menimbulkan kesulitan pada saat pemeliharaan data dan pemeliharaan program aplikasi.

3. CLIENT SERVER

Arsitektur Client Server merupakan arsitektur yang mencoba menghilangkan kelemahan-kelemahan yang ditemukan pada kedua arsitektur yang disebutkan sebelumnya. Pada Arsitektur client server terdapat sebuah atau lebih server yang memiliki kemampuan pemrosesan lebih dari komputer-komputer lain yang terhubung dalam jaringan. Disini data dan pemrosesan dilakukan terpusat, namun terdapat pembagian kerja antara server dan client. Komputer-komputer client akan melakukan pemrosesan awal terhadap data-data yang masuk dan melakukan pemrosesan akhir pada data data yang akan dikeluarkan lewat komputer client. Server hanya akan melakukan pemrosesan pada data data yang diperlukan untuk dipakai bersama. (*Client/Server Architecture*)

Program aplikasi pada arsitektur ini diletakkan tersebar, program aplikasi yang diletakkan di client merupakan program-program front-end. Program yang berada di server merupakan program yang bertugas melayani permintaan pelayanan dari program program pada client. Program yang

melayani ini disebut sebagai back-end. (*Microsoft SQL 6,5 Unleashed*).

Dengan cara seperti ini maka kelemahan dari kedua arsitektur sebelumnya dapat diperbaiki. Server tidak perlu sebaik kualitas host dan tentunya harganya juga jauh lebih murah bahkan dapat diaplikasikan dengan personal computer. Pemeliharaan data relatif lebih mudah karena masih terpusat letaknya. Pemeliharaan program aplikasi juga dapat dilakukan dengan mudah karena terpisah dengan jelas antara front end dan back end.

4. SERVER DATABASE

1. SEKILAS SERVER DATABASE

Database server merupakan aplikasi back end yang digunakan untuk melakukan pengelolaan database yang digunakan bersama pada sebuah jaringan. Server database bertugas melayani semua permintaan data maupun manipulasi data dari client yang terhubung dalam jaringan tersebut. Database server juga bertugas untuk mengatur keamanan data, database server mengatur siapa yang berhak dan siapa yang tidak berhak mengakses data, dan memanipulasi data. Database server akan mengidentifikasi user dengan user id beserta passwordnya. Setiap user ini diberi hak masing masing tergantung posisinya didalam struktur organisasi di dalam

jaringan tersebut. Untuk penentuan hak masing masing user tersebut dilakukan oleh seorang user dengan kualifikasi database administrator. Database Administrator ini memiliki hak untuk mengatur hak (right) dari masing masing user yang terdaftar, membuat user baru dan menghapus seorang user dari daftar user.

Ada beberapa produk database server yang beredar di pasaran, beberapa yang terkenal seperti DB2, MS-SQL, Interbase, Oracle dan lain-lain. Dari semua produk itu hampir semuanya mendukung penggunaan standar Query language (SQL). Ms-SQL adalah salah satu dari produk database server buatan Microsoft, database server inilah yang selanjutnya akan dikupas oleh penulis.

2. SQL (STANDARD QUERY LANGUAGE)

Standard Query language adalah sekumpulan perintah untuk melakukan pengaksesan terhadap data. Bahasa ini dibuat standarnya dan saat ini dipakai pada semua server database. Walaupun ada beberapa penambahan feature pada masing-masing server database namun bahasa query standar ini dapat digunakan pada semua server database yang ada.

Bahasa Query dapat dibagi menjadi dua bagian yaitu DML(Data manipulation language) yang berupa perintah perintah untuk melakukan manipulasi data seperti menambah

data, mengedit data dan proses-proses manipulasi data yang lain. Bagian yang lain adalah DDL(Data definition language) yang digunakan untuk melakukan pendefinisian data seperti membuat tabel, memfilter tabel dan lain-lain.

4. MS-SQL SERVER

MS-SQL Server merupakan salah satu server database yang saat ini banyak dipakai. Server database ini merupakan produk dari Microsoft. Server database ini berjalan di atas sistem operasi Windows NT. Sebagai aplikasi back end MS-SQL dapat di padukan dengan bahasa pemrograman under windows apa saja untuk membuat aplikasi front end-nya.

1. SECURITY

Untuk menjaga keamanan data MS-SQL Server menerapkan pembatasan terhadap hak akses untuk setiap user yang terdaftar. Database administrator (DBA) dibuat pada saat pertama kali menjalankan server ini setelah diinstall. Pada saat log on pertama kali DBA dipersilahkan untuk mengisikan passwordnya. Setelah itu jika kita logon ke server database menggunakan DBA maka kita berhak melakukan apa saja baik terhadap data, user, maupun pengaturan device yang digunakan untuk menyimpan database.

Untuk mengatur pemakai/user seorang DBA berhak untuk Mendefinisikan user ID, passwordnya, groupnya serta hak dari masing masing user terhadap database yang ada. Pengelompokan user kedalam group user dilakukan untuk mempermudah pengaturan hak (rights)masing masing user. Setiap group dapat di beri rights, secara otomatis user yang berada pada group tersebut akan mewarisi hak yang dimiliki oleh groupnya. Pengaturan Rights disini berorientasi kepada obyek tabel, jadi seorang user yang berhak menghapus data pada suatu tabel tidak secara otomatis mendapat hak yang sama jika mengakses tabel yang lain. Bahkan hak seorang user bisa diatur hanya untuk field-field tertentu saja dalam sebuah tabel. Misalnya seorang user hanya diberi hak untuk mengedit field nama dan alamat saja, tetapi tidak didijinkan untuk mengedit field kode pelanggan. Seorang user juga dapat dibatasi haknya untuk mengakses database hanya pada saat saat tertentu saja.

Beberapa macam hak pengaksesan data yang dapat dimiliki oleh user adalah hak untuk melihat data (browse) pada tabel. Hak untuk mengedit data(Edit) pada tabel. Hak untuk menambah data pada tabel (Insert) pada tabel. Hak untuk melakukan penghapusan data (Delete) pada tabel. Hak untuk melakukan eksekusi (Execute) khusus terhadap store procedure.

2. MANAJEMEN DATA

Di dalam MS-SQL Server Untuk membuat sebuah database pertama kali yang dilakukan adalah mendefinisikan device yang akan digunakan untuk menyimpan data. Disini ditentukan dimana letaknya secara fisik misalnya di direktori c:\data lalu berapa besarnya atau kapasitas maksimal data yang dapat ditampung.

Selanjutnya dilakukan pembuatan database-nya. Setelah kita memiliki database kita tinggal membuat tabel-tabel yang ingin kita cantumkan dalam database tersebut. Pada saat membuat tabel kita dapat memberikan sekalian constraintnya, seperti referential integritynya serta validitas data untuk masing masing field pada tabel tersebut. Untuk mengatur validitas data pada masing-masing field dapat dilakukan dengan menerapkan rule untuk masing masing field. Rule ini bisa dianalogikan dengan picture pada Clipper atau Maskedit pada Delphi yaitu proses pembatasan nilai mana yang boleh dan mana yang tidak boleh. MS-SQL melakukan pengecekan secara otomatis sesuai rule yang ditetapkan setiap kali terjadi perubahan data. Setelah tabel dibuat langkah selanjutnya adalah menentukan siapa saja yang berhak melakukan pengaksesan data.

Data adalah sesuatu yang berharga, konsep ini juga diakomodasi oleh MS-SQL yang menawarkan cara untuk melakukan backup secara otomatis. Untuk membuat backup secara berkala, MS-SQL akan melakukannya sendiri setelah kita tentukan pada device mana dia harus membuat backup data. Kita juga bisa memberikan perintah untuk melakukan backup secara otomatis setiap hari pada jam 4 sore atau kapan saja. Sekali itu diatur maka MS-SQL akan melakukannya dengan patuh dengan catatan pada saat itu komputer server database dalam keadaan aktif.

Untuk menduplikat database yang kita miliki MS_SQL menawarkan cara yang mudah untuk membuat duplikat dari database yang kita miliki sekaligus bersama isi datanya. Ini memberikan kemudahan kepada kita jika suatu saat kita ingin memindahkan server database kita.

3. STORE PROCEDURE

Store procedure seperti layaknya procedure pada bahasa pemrograman. Dia merupakan sebuah sub rutin yang akan melakukan sesuatu jika di panggil atau dieksekusi. Store procedure juga mengenal parameter, yaitu nilai-nilai yang akan dilewatkan baik sebagai masukan atau keluaran yang didapatkan jika store procedure ini dijalankan.

Store procedure terletak atau disimpan di dalam database dan termasuk salah satu obyek yang dimiliki oleh database disamping tabel. Store procedure ini berisi pernyataan-pernyataan bahasa query. Store procedure dibuat agar kita tidak terlalu banyak menuliskan kode yang sama jika ada pekerjaan-pekerjaan yang sama dan berulang-ulang. Jika ditilik dari namanya seharusnya berupa prosedur untuk menyimpan, namun itu pengertian yang salah karena sebenarnya store procedure dapat melakukan apa saja seperti browsing data, edit data dan sebagainya, jadi tidak hanya menyimpan data saja.

Penggunaan store procedure ini sangat dianjurkan pada pemrograman database jaringan, karena jika perintah-perintah SQL tersebut kita perintahkan dari sisi client, padahal clientnya cukup banyak akan menimbulkan kesibukan jalur komunikasi menjadi sangat padat. Jika dibuat store procedure client hanya mengirimkan nama store procedure beserta parameternya, sedangkan jika tidak maka yang dikirim ke server berupa perintah-perintah yang cukup panjang. Hal ini akan berimbas pada kinerja program dalam jaringan tersebut. Dengan menggunakan store procedure waktu eksekusi akan banyak berkurang karena alasan kelambatan proses komunikasi data antara client dengan server.

4. TRIGER

Triger adalah mirip dengan store procedure yang berupa pernyataan-pernyataan dalam bahasa SQL yang disimpan dalam database. Triger akan dilakukan pada event-event yang dapat diatur. Misalnya jika ada proses penghapusan database diinginkan melakukan pengecekan apakah field hutang =0 atau tidak, jika =0 maka penghapusan dilaksanakan jika sebaliknya maka proses penghapusan data dibatalkan atau ditolak.

Demikian kupasan singkat mengenai pemrograman database client server menggunakan MS-SQL server 6.5. Sebagai penutup penulis mohon kritik serta saran dari pembaca terhadap makalah ini. Kupasan diatas masih berkisar pada gambaran konseptual, adapun secara teknis memang tidak mungkin dibahas secara mendetail dalam makalah ini, untuk memahami lebih mendalam rujukan pada tulisan ini dapat dijadikan bahan yang cukup lengkap.

DAFTAR PUSTAKA

- Alex Person & Joy Ranade. 1995. *Client Server Architecture*. McGraw Hill International Edition.
- Special Edition Using VB 4*. Simon & Schuster (Asia) Pte Ltd. 1996.