

**DATABASE CLIENT SERVER**  
**DAN**  
**OPEN DATABASE CONNECTIVITY ( ODBC )**



**Disusun Oleh :**

<b>Nama ( NIS )</b>	<b>: Amirudin Nursi ( 177347 )</b>
	<b>Diaz Angger N P ( 177352 )</b>
	<b>Nabil Syahfiar ( 177363 )</b>
	<b>Riska Rahmasari ( 177366 )</b>
	<b>Sandika Agna A ( 177368 )</b>
<b>Kelas</b>	<b>: XII RPL 1</b>
<b>Kompetensi Keahlian</b>	<b>: Rekayasa Perangkat Lunak</b>
<b>Tahun Pembelajaran</b>	<b>: 2019 / 2020</b>

---

**YAYASAN PUSAT PENGEMBANGAN ILMU DAN TEKNOLOGI TEXMACO**  
**SEKOLAH MENENGAH KEJURUAN (SMK) TEXMACO SEMARANG**  
**Jalan Raya Mangkang Km. 16 Semarang 50155**  
**Telp (024) 8661966 Fax (024) 8661967 Email : [smktex\\_smg@yahoo.com](mailto:smktex_smg@yahoo.com)**

## KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa, karena dengan pertolonganNya kami dapat menyelesaikan makalah yang berjudul “Database Client - Server dan ODBC”. Meskipun banyak rintangan dan hambatan yang kami alami dalam proses pengerjaannya, tapi kami berhasil menyelesaikannya dengan baik.

Tentunya ada hal-hal yang ingin kami berikan kepada para pembaca dari hasil tulisan ini. Karena itu saya berharap semoga makalah ini dapat menjadi sesuatu yang berguna bagi kita bersama. Semoga makalah yang saya buat ini dapat membuat kita mengerti akan konsep dan pengertian dari client-server dan ODBC tersebut.

DAFTAR ISI

KATA PENGANTAR.....i

DAFTAR ISI.....ii

BAB I PENDAHULUAN

1.1 LATAR BELAKANG .....1

1.2 MAKSUD DAN TUJUAN .....1

BAB II PEMBAHASAN

2.1 CLIENT – SERVER .....2

2.2 OPEN DATABASE CONNECTIVITY ( ODBC ) .....8

BAB III PENUTUP

3.1 KESIMPULAN.....11

DAFTAR PUSTAKA .....13

# **BAB I**

## **PENDAHULUAN**

### **1.1 LATAR BELAKANG**

Di era globalisasi ini, dimana segala sesuatunya itu berjalan dengan cepat, kemajuan teknologi semakin memudahkan manusia untuk berkomunikasi dan saling bertukar informasi. Semua orang di zaman sekarang ini hampir setiap individu sudah memiliki komputer.

Di mana didalam dunia komputer ada yang namanya client server dan ODBC, Client merupakan sembarang sistem atau proses yang melakukan suatu permintaan data atau layanan ke server sedangkan server ialah, sistem atau proses yang menyediakan data atau layanan yang diminta oleh client.

Client-Server adalah pembagian kerja antara server dan client yg mengakses server dalam suatu jaringan. Jadi arsitektur client-server adalah desain sebuah aplikasi terdiri dari client dan server yang saling berkomunikasi ketika mengakses server dalam suatu jaringan.

Open Database Connectivity (disingkat menjadi ODBC) adalah sebuah standar terbuka untuk konektivitas antar mesin basis data.

### **1.2 MAKSUD DAN TUJUAN**

Maksud dan tujuan dari Database Client Server dan ODBC yaitu untuk membantu perusahaan-perusahaan dalam meningkatkan pengintegrasian data, distribusi informasi dan sebagainya tentunya yang berhubungan dengan client server, ini juga menempatkan sebuah komputer sebagai server yang bertugas sebagai pusat pengolahan dan layanan bagi terminal-terminal lain (client) yang terhubung dalam sistem jaringan tersebut.

## **BAB II**

### **PEMBAHASAN**

#### **2.1 CLIENT – SERVER**

Client-Server merupakan suatu paradigma hubungan antara computer yang menjadi penyedia dan computer yang menjadi penerima. Hubungan suatu CLIENT-SERVER adalah computer meminta sesuatu ke computer lain dan computer lain memenuhi permintaan. Hubungan ini menggunakan jaringan agar tetap terhubung satu sama lain, menggunakan metode LAN, MAN, atau WAN.

Metode LAN menghubungkan suatu Client ke Server yang lain dengan jarak yang sangat dekat, Metode MAN menghubungkan jarak yang sedang antara Client dan Server dan Metode WAN menghubungkan Client dengan Server yang lain dengan jarak yang jauh, WAN bisa juga disebut dengan INTERNET.

Server digunakan untuk mengirimkan dan menyediakan keperluan dari client. Dalam penyediaannya server dapat membatasi file dan data yang diminta oleh client hal tersebut dilakukan untuk mencegah suatu kejadian yang tidak terduga seperti server diserang dengan cara mengirim program tidak bertanggung jawab atau virus. Sehingga server memiliki hak yang lebih tinggi dibandingkan dengan client.

SERVER atau KOMPUTER SERVER adalah suatu sistem komputer yang menyediakan layanan disebuah jaringan komputer yang dilengkapi dengan prosesor cepat dan RAM yang besar, dilengkapi dengan sistem operasi khusus untuk server

Server mengontrol akses terhadap jaringan dan sumber daya yang terdapat di dalamnya seperti file atau printer, dan memberikan akses kepada Clients.

CLIENTS adalah suatu system yang mengakses suatu system computer yang menyediakan layanan atau server melalui jaringan seperti LAN, MAN, WAN. Clients meminta data ke server dan server memenuhi permintaan tersebut yang dikirimkan melalui jaringan. Client Server terdiri dari 3 komponen pembentuk yaitu Client, Middleware, dan Server.

#### a. FUNGSI CLIENT SERVER

Fungsi dari CLIENT – SERVER adalah untuk memudahkan pekerjaan dalam suatu system jaringan, yang diminta oleh client dapat dipenuhi oleh server dalam waktu singkat atau sebaliknya. Sehingga pekerjaan menjadi efisien, berikut adalah tugas dari server dan client.

Server :

- Menerima dan memproses basis data yang diminta dari client
- Memeriksa otorisasi
- Menjamin tidak terjadi pelanggaran terhadap integrity constraint
- Melakukan query/pemrosesan update dan memindahkan response ke client
- Memelihara data dictionary

Client :

- Mengatur user interface
- Menerima dan memeriksa sintaks input dari pemakai
- Memproses aplikasi
- Generate permintaan basis data dan memindahkannya ke server
- Memberikan response balik kepada pemakai
- Menyediakan akses basis data secara bersamaan
- Menyediakan kontrol recovery

**b. JENIS CLIENT SERVER**

1. File Server
2. Database Server
3. Server Transactions
4. Groupware Server
5. Object Application Server
6. Web Application Server

**c. TIPE CLIENT SERVER**

1. Standalone (One Tier)

Pemrosesan data dilakukan pada mainframe.

Kelebihan :

- Sangat mudah
- Cepat dalam merancang dan mengaplikasikan

Kekurangan :

- Skala kecil
- Susah diamankan
- Menyebabkan perubahan terhadap salah satu komponen diatas tidak mungkin dilakukan, karena akan mengubah semua bagian
- Tidak memungkinkan adanya re-usable component dan code
- Cepat dalam merancang dan mengaplikasikan

## 2. Client / Server (Two Tier)

Pemrosesan data pada sebuah aplikasi terjadi pada client dan server.

Kelebihan :

- Mudah
- Menangani Database Server secara khusus
- Relatif lebih sederhana untuk di develop dan diimplementasikan
- Lebih cocok diterapkan untuk bisnis kecil

Kekurangan :

- Kurangnya skalabilitas
- Koneksi database dijaga
- Tidak ada keterbaharuan kode
- Tidak ada tingkat menengah untuk menangani keamanan dan transaksi skala kecil
- Susah di amankan
- Lebih mahal



### 3. Three Tier

Arsitektur ini pengembangan arsitektur Client Server. Arsitektur Three Tier ini terdapat Application Server yang berdiri di antara Client dan Database Server. Contoh dari Application server adalah IIS, WebSphere, dan sebagainya.

Kelebihan :

- Segala sesuatu mengenai database terinstalasikan pada sisi server, begitu pula dengan pengkonfigurasian. Hal ini membuat harga yang harus dibayar lebih kecil
- Apabila terjadi kesalahan pada salah satu lapisan tidak akan menyebabkan lapisan lain ikut salah
- Perubahan pada salah satu lapisan tidak perlu menginstalasi ulang pada lapisan yang lainnya dalam hal ini sisi server ataupun sisi client
- Skala besar
- Keamanan dibelakang firewall
- Transfer informasi antara web server dan server database optimal
- Komunikasi antara system-sistem tidak harus didasarkan pada standart internet, tetapi dapat menggunakan protocol komunikasi yang lebih cepat dan berada pada tingkat yang lebih rendah
- Penggunaan middleware mendukung efisiensi query database dalam SQL di pakai untuk menangani pengambilan informasi dari database

Kekurangan :

- Lebih susah merancang dan mengatur
- Lebih mahal

#### 4. Multi Tier

Arsitektur Multi Tier mirip dengan Three Tier. Kelebihan arsitektur ini dilengkapi UI (User Interface) dan Data Processing. Khususnya dengan adanya Business Logic Server.

Kelebihan :

- Dengan menggunakan aplikasi multi-tier database, maka logika aplikasi dapat dipusatkan pada middle-tier, sehingga memudahkan untuk melakukan control terhadap client-client yang mengakses middle server dengan mengatur seting pada dcomcnfg
- Dengan menggunakan aplikasi multi-tier, maka database driver seperti BDE/ODBC untuk mengakses database hanya perlu diinstal sekali pada middle server, tidak perlu pada masing-masing client
- Pada aplikasi multi-tier, logika bisnis pada middle-tier dapat digunakan lagi untuk mengembangkan aplikasi client lain, sehingga mengurangi besarnya program untuk mengembangkan aplikasi lain. Selain itu meringankan beban pada tiap-tiap mesin karena program terdistribusi pada beberapa mesin
- Memerlukan adaptasi yang sangat luas ruang lingkupnya apabila terjadi perubahan sistem yang besar

Kekurangan :

- Program aplikasi tidak bisa mengquery langsung ke database server, tetapi harus memanggil prosedur-prosedur yang telah dibuat dan disimpan pada middle-tier
- Lebih mahal

d. **TUJUAN CLIENT SERVER**

Dibentuknya CLIENT – SERVER adalah untuk memudahkan komunikasi antara satu komputer dengan komputer yang lain dalam suatu sistem jaringan kerja, dengan bantuan Server dan Jaringan.

e. **APLIKASI CLIENT SERVER**

Pengaplikasian dari CLIENT – SERVER adalah seperti LAN . Contoh aplikasi dalam CLIENT-SERVER adalah SKYPE ,kita mengirimkan data ke SKYPE dan Server SKYPE menerima data yang CLIENT kirim dimana SERVER SKYPE menerjemahkan kode yang kita kirim ke servernya setelah diterjemahkan SERVER mengirimkan data ke CLIENT sesuai apa yang SERVER sudah terjemahkan sebelumnya dan sebaliknya, adapun contoh aplikasi client server yang lain seperti Whatsapp Messenger, We Chat dan sebagainya.

## 2.2 **OPEN DATABASE CONNECTIVITY**

*Open Database Connectivity (disingkat menjadi ODBC) adalah sebuah standar terbuka untuk konektivitas antar mesin basis data.* Standar ini menyediakan API yang dapat digunakan untuk menjalankan dan mengoneksikan sebuah aplikasi dengan sebuah sistem manajemen basis data (SMBD). Para desainer ODBC membuatnya dengan tujuan agar ODBC terbebas dari penggunaan bahasa pemrograman tertentu, sistem manajemen basis data tertentu, dan sistem operasi tertentu.

*ODBC merupakan protokol standard yang dapat menghubungkan segala jenis database eksternal server atau file eksternal server.* Driver ODBC yang digunakan oleh Mesin database Microsoft Jket dapat mengakses *Microsoft SQL Server* dan beberapa database eksternal lainnya. ODBC dapat mengakses sumber data baik lokal maupun jarak jauh (remote) dengan menggunakan driver ODBC. Driver ODBC 16 bit atau 32 bit keduanya dapat digunakan untuk mengakses sumber data.

*Open Database Connectivity (ODBC) adalah Application Programming interface (API) database yang khusus digunakan untuk mengakses database relasional.* ODBC terdapat dalam setiap komputer yang menggunakan sistem operasi windows, karena ODBC merupakan bagian dari Windows Open System Architecture (WOSA).

Dalam ODBC disediakan berbagai Application Programming Interface (API) yang berguna untuk menyediakan dan memberikan stkitar bagi berbagai kegiatan pemrograman. Keuntungan utama menggunakan ODBC ini adalah fleksibilitas, fleksibel disini artinya perubahan jenis database yang dipergunakan oleh sebuah aplikasi tidak akan mempengaruhi kode program aplikasi tersebut.

ODBC terdapat dalam Control Panel Windows, yang ditunjukkan dengan icon Data Source (ODBC). Hanya saja tempat tampilan ODBC tidak sama untuk beberapa jenis windows, seperti Windows98, 2000, NT, XP, Me masing-masing akan mempunyai tampilan yang berbeda untuk lokasi ODBC.

Pada ODBC terdapat **2 hal penting yang berkaitan dengan koneksi database**, yaitu :

1. Database Driver

Database Driver adalah software/driver yang berfungsi untuk membuat koneksi ke database yang sesuai dengan database tersebut. Setiap database memiliki driver yang spesifik, sehingga dalam sebuah ODBC data berisi lebih dari 1 database driver.

2. Data Source Name ( DSN )

Data Source Name (DSN) adalah nama/alias dari database yang digunakan oleh program, yang koneksinya melalui ODBC.

Dalam ODBC terdapat beberapa **jenis Data Source Name** yang akan menentukan otoritas pemakai database, yaitu :

1. User DSN

User DSN bersifat private bagi user pembuatnya, artinya DSN jenis ini hanya dapat dipergunakan/tampak oleh user yang membuatnya (yaitu user yang login dengan username pembuat DSN tersebut), yang perlu menjadi catatan adalah bahwa sifat user DSN ini hanya berjalan pada Windows NT.

2. Sistem DSN

Sistem DSN bersifat public dalam sebuah komputer, artinya semua user yang menggunakan komputer dimana Sistem DSN di-install dapat menggunakannya walaupun bukan pembuatnya.

### 3. File DSN

File DSN adalah satu-satunya yang bisa di-share dan dapat dipergunakan komputer lain yang memiliki driver database yang sama, dengan demikian setting untuk koneksi database cukup dilakukan pada 1 komputer saja dan komputer yang lain cukup me-load-nya saja. File DSN ini umumnya dipergunakan untuk aplikasi multiuser pada suatu LAN.

ODBC memiliki beberapa ***komponen utama***, yakni sebagai berikut :

#### 1. ODBC API

Sekumpulan panggilan fungsi, kode-kode kesalahan dan sintaksis SQL yang mendefinisikan bagaimana data dalam sebuah DBMS diakses.

#### 2. Driver database ODBC

Driver (yang berupa dynamic link library) yang mampu memproses panggilan fungsi ODBC untuk sebuah DBMS tertentu.

#### 3. ODBC Driver Manager

Yang bertugas untuk memuat driver basis data ODBC yang dibutuhkan oleh aplikasi.

Pengguna sistem operasi Windows dapat mengonfigurasi ODBC dengan menggunakan utilitas ODBC yang terdapat di dalam Control Panel. Utilitas ini mengizinkan mereka untuk mendefinisikan DSN (Data Source Name) untuk basis data yang hendak diakses

dan juga driver yang digunakan untuk mengakses basis data tersebut. Untuk mengonfigurasi ODBC dengan utilitas ini, mereka dapat mengasosiasikan sebuah DSN dengan sebuah driver ODBC, yang mengizinkan aplikasi untuk dapat berinteraksi dengan sebuah basis data yang disimpan baik secara lokal di dalam mesin yang sama maupun di dalam server jaringan.

## **BAB II**

### **PENUTUP**

#### **3.1 KESIMPULAN**

Dalam konteks basis data, client mengatur interface berfungsi sebagai workstation tempat menjalankan aplikasi basis data. Client menerima permintaan pemakai, memeriksa sintaks dan generate kebutuhan basis data dalam SQL atau bahasa yang lain. Kemudian meneruskan pesan ke server, menunggu response dan bentuk response untuk pemakai akhir.

Model Two-tier terdiri dari tiga komponen yang disusun menjadi dua lapisan yaitu client (yang meminta service) dan server (yang menyediakan service).

Tiga komponen tersebut yaitu :

1. User Interface. Adalah antar muka program aplikasi yang berhadapan dan digunakan langsung oleh user.
2. Manajemen proses.
3. Database. Model ini memisahkan peranan user interface dan database dengan jelas, sehingga terbentuk dua lapisan.

Model three-tier atau multi-tier dikembangkan untuk menjawab keterbatasan pada arsitektur client/server. Dalam model ini, pemrosesan disebarkan di dalam tiga lapisan (atau lebih jika diterapkan arsitektur multitier).

Sebuah sistem 3-tier menyediakan support multi-user yang stabil, bahkan saat pada client menjalankan aplikasi yang berbeda, juga dapat mendayagunakan beberapa database yang digunakan secara bersamaan.

Diantara keuntungan-keuntungan yang dapat diperoleh dari arsitektur n-tier (atau 3-tier pada umumnya), yang terutama adalah :

1. Kemudahan perubahan business logic di masa yang akan datang.
2. Business logic yang mudah diimplementasi dan dipelihara.
3. Aplikasi client dapat mengakses berbagai tipe DBMS yang berbeda-beda secara transparan.

## DAFTAR PUSTAKA

Zainal hakim. Pengertian Client server

<http://www.zainalhakim.web.id/pengertian-client-server.html>

Luviyani. client – server

<http://luviyani.blogspot.com/2010/04/client-server.html>

Muhammad Abdullah. Arsitektur Client Server

<http://achmad89.wordpress.com/2010/07/11/arsitektur-client-server.html>

Slamet ridwan. Arsitektur Client Server

<http://slametridwan.wordpress.com/arsitektur-client-server.html>