**MODUL MATA KULIAH**

**PENGANTAR SISTEM BASIS DATA**

**KP213 - 2 SKS**

FAKULTAS TEKNOLOGI INFORMASI

**UNIVERSITAS BUDI LUHUR**



JAKARTA SEPTEMBER 2019

T I M

P E N Y U S U N

Safrina Amini

Sri Mulyati

Gandung Triyono

|  |  |  |
| --- | --- | --- |
|  | UNIVERSITAS BUDI LUHUR  FAKULTAS TEKNOLOGI INFORMASI |  |

MODUL PERKULIAHAN #5

**ARSITEKTUR SISTEM BASIS DATA**

|  |  |  |
| --- | --- | --- |
| Capaian Pembelajaran | : | Mahasiswa mampu menjelaskan berbagai bentuk arsitektur dari sistem basis data |
| Sub Pokok Bahasan | : | 1.1. Centralized and Client-Server Architecture  1.2. Server System Architecture  1.3. Parallel Systems  1.4. Distributed Systems |
| Daftar Pustaka | : | 1. Database System A Practical Approach to Design, Implementation, and Management by Thomas M. Connolly, Carolyn E. Begg  2. An Introduction to Database Systems by C.J. Date |

**1.1. Centralized and Client-Server Architecture**

Pentingnya memahami arsitektur database

* Memiliki gambaran tentang pola kerja database
* Mampu menentukan arsitektur database yang tepat sesuai kondisi lapangan

Arsitektur Aplikasi database

* Arsitektur yang umum digunakan ialah arsitektur client-server. Dengan ciri:
  + Secara logic, dbms dan aplikasi yang menggunakan terpisah ( misalnya software aplikasi menggunakan vb dan database menggunakan SQLServer)

Client-Server

1. Dalam implementasinya, ada 3 bagian yang berbeda saat mengolah data yang diperlukan user:
   1. Bagian penyimpanan data (logika penyimpanan)
   2. Bagian proses (logika pemrosesan)
   3. Bagian tampilan (logika presentasi)
2. Logika penyimpanan

* Bertanggung jawab untuk melakukan proses baca-tulis ke database
* Manifestasi dalam bentuk perintah DML

1. Logika pemrosesan

* Bertanggung jawab untuk melakukan pengolahan data yang akan ditulis atau ditampilkan
* Misalnya menghitung nilai total dan rata-rata dari data hasil query.
* Termasuk pengolahan data sesuai aturan bisnis yang berlaku

1. Logika presentasi

* Bertanggung jawab untuk menampilkan data dan informasi yang dibaca atau akan disimpan pada database
* Pada praktek arsitektur client –server, dikenal 3 jenis pemusatan sumber daya.
  1. Fat Client

Proses berat di client, ringan di server

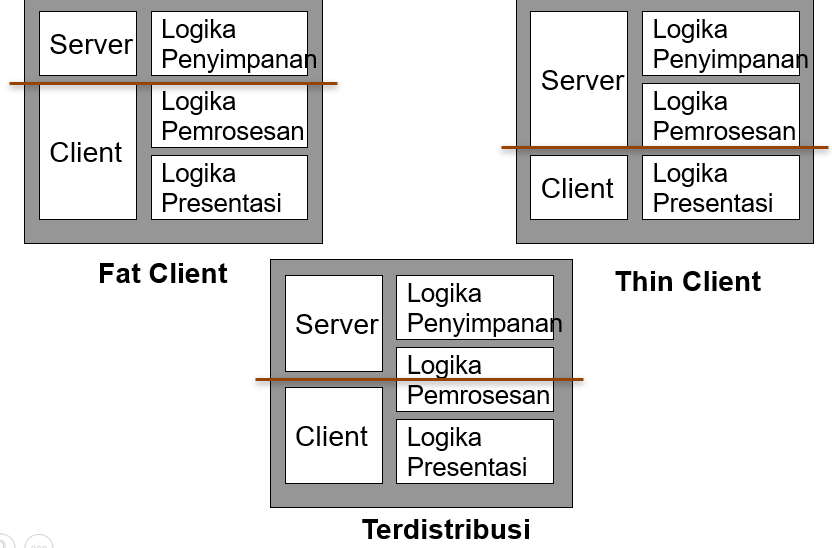
* 1. Thin client

Proses berat di server, ringan di client

* 1. Terdistribusi

Proses berimbang di client dan server

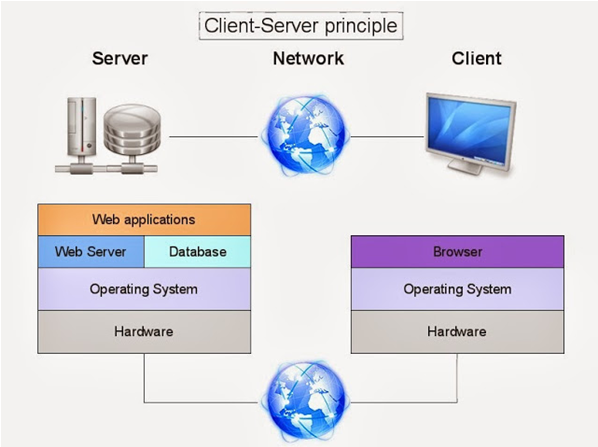
Beban kerja jenis-jenis client



Konsep mana yang lebih tepat?

* Relatif berdasarkan kebutuhan dan kondisi lapangan

**Arsitektur Klien-Server (Client-Server)**

[](https://mildsend.files.wordpress.com/2013/11/client-server.png)

**Klien-Server** atau **Client-Server** merupakan sebuah paradigma dalam [teknologi informasi](http://id.wikipedia.org/wiki/Teknologi_informasi" \t "_top" \o "Teknologi informasi) yang merujuk kepada cara untuk mendistribusikan aplikasi ke dalam dua pihak : Pihak Klien dan Pihak Server. Client-server sebagai arsitektur yang paling banyak digunakan saat ini. Dimana client dapat melakukan proses sendiri, ketika client meminta data, server akan mengirimkan data sesuai yang diminta, kemudian proses akan dilakukan di client.

* **Client** adalah sembarang sistem atau proses yang melakukan suatu permintaan data atau layanan ke server.
* **Server** adalah sistem atau proses yang menyediakan data atau layanan yang diminta oleh client. Secara fisik, sebuah server dapat berupa komputer mainframe, mini-komputer, workstation, ataupun PC atau peranti lain seperti printer, server tidak harus berupa sistem fisik, tetapi juga suatu proses.

Dalam model Klien/Server, sebuah aplikasi dibagi menjadi dua bagian yang terpisah, tapi masih merupakan sebuah kesatuan yakni komponen Klien dan komponen Server. Komponen klien juga sering disebut sebagai *front-end*, sementara komponen server disebut sebagai *back-end*. Komponen klien dari aplikasi tersebut dijalankan dalam sebuah workstation dan menerima masukan data dari pengguna. Komponen klien tersebut akan menyiapkan data yang dimasukkan oleh pengguna dengan menggunakan teknologi pemrosesan tertentu dan mengirimkannya kepada komponen server yang dijalankan di atas mesin server, umumnya dalam bentuk *request* terhadap beberapa layanan yang dimiliki oleh server. Komponen server akan menerima request dari klien dan langsung memprosesnya lalu mengembalikan hasil pemrosesan tersebut kepada klien. Klien pun menerima informasi hasil pemrosesan data yang dilakukan server dan menampilkannya kepada pengguna, dengan menggunakan aplikasi yang berinteraksi dengan pengguna.

Sebuah contoh dari aplikasi client/server sederhana adalah aplikasi web yang didesain dengan menggunakan [Active Server Pages](http://id.wikipedia.org/wiki/Active_Server_Pages" \t "_top" \o "Active Server Pages) (ASP) atau [PHP](http://id.wikipedia.org/wiki/PHP" \t "_top" \o "PHP). Skrip PHP atau ASP akan dijalankan di dalam web server ([Apache](http://id.wikipedia.org/wiki/Apache_HTTP_Server" \t "_top" \o "Apache HTTP Server) atau [Internet Information Services](http://id.wikipedia.org/wiki/Internet_Information_Services" \t "_top" \o "Internet Information Services)), sementara skrip yang berjalan di pihak klien akan dijalankan oleh *[web browser](http://id.wikipedia.org/wiki/Web_browser" \t "_top" \o "Web browser)* pada komputer klien. Klien-server merupakan penyelesaian masalah pada software yang menggunakan database sehingga setiap komputer tidak perlu diinstall database, dengan metode klien-server database dapat diinstal pada suatu komputer sebagai server dan aplikasinya diinstal pada client.

**Implementasi client-server :**

* Aplikasi pesan, misalnya surat elektronik (email).
* Penyebaran basis data pada beberapa jaringan komputer.
* Memungkinkan berbagi berkas atau periferal atau pengaksesan komputer melalui jarak jauh.
* Pemrosesan aplikasi yang intensif dengan suatu pekerjaan (*job*) dibagi menjadi tugas-tugas (*task*) yang masing-masing dilaksanakan pada komputer yang berbeda.

**Kelebihan Arsitektur client-server :**

1. Pemrosesan dapat dilakukan di computer client, sehingga data dapat diproses sesuai dengan kebutuhan client.
2. Proses bisnis tetap akan berjalan meskipun terjadi kemacetan mesin.
3. Pada arsitektur client-server hanya dibutuhkan mesin-mesin yang sederhana, sehingga dapat mengurangi biaya dalam membangun sistem.
4. Mudah dalam melakukan up-grade pada perangkat sistem.
5. Dapat menggunakan berbagai platform aplikasi pada client.

**Keuntungan Client-Server :**

|  |  |
| --- | --- |
| **Fitur** | **Keuntungan** |
| * Jaringan mesin-mesin yang kecil tetapi berdaya guna | * Jika sebuah mesin macet, bisnis tetap berjalan |
| * Kumpulan komputer dengan ribuan MIPS (million instruction per second) | * Sistem memberikan kekuatan dalam melaksanakan suatu tugas tanpa memonopoli sumber-sumber daya. Pemakai akhir diberi hak untuk bekerja secara local |
| * Beberapa workstation sangat handal seperti mainframe, tetapi dengan biaya 90% lebih rendah | * Menawarkan keluwesan untuk melakukan pembelian pada hal-hal lain atau untuk meningkatkan keuntungan |
| * Sistem terbuka | * Bebas memilih perangkat keras, perangkat lunak, dan layanan dari berbagai vendor |
| * Sistem tumbuh dengan mudah dan dapat diperluas secara tak terbatas | * Mudah untuk memperbaharui system |
| * Lingkungan operasi client yang bersifat individual | * Dapat mencampur dan mencocokkan platform komputer yang gsesuai dengan kebutuhan masing-masing departemen dan pemakai |

Arsitektur Client-Server ini dibagi dalam 2 bagian arsitektur yaitu :

**1) Arsitektur Client Side**

Merujuk pada pelaksanaan data pada browser sisi koneksi HTTP. JavaScript

adalah sebuah contoh dari sisi eksekusi client dan contoh dari sisi penyimpanan

pada client adalah cookie.

**Karakteristik :**

* Pihak klien selalu memulai permintaan/permohonan ke pihak server
* Setelah mengirim permintaan, klien menunggu jawaban atas permintaannya dari server
* Menerima balasan dari server atas permintaannya
* Biasanya klien akan terhubung ke sejumlah kecil dari server pada satu waktu
* Biasanya berinteraksi langsung dengan end-user (pengguna akhir) dengan menggunakan user interface (antarmuka pengguna)
* Khusus jenis klien mencakup web browser, email klien dan online chat klien

**2)** **Arsitektur Server Side**

Pada server side, ada sebuah server Web khusus yang bertugas mengeksekusi perintah dengan menggunakan standar metode HTTP. Misalnya penggunaan CGI script pada sisi server yang mempunyai tag khusus yang tertanam di halaman HTML. Tag ini memicu terjadinya perintah untuk mengeksekusi.

**Karakteristik :**

* Sebagai penyedia layanan, sisi server akan selalu menunggu permintaan dari sisi klien
* Sesuai dengan tugasnya, melayani dan menjawab permintaan data yang diminta oleh klien
* Sebuah server dapat berkomunikasi dengan server lain untuk melayani permintaan klien
* Jenis server khusus mencakup web server, FTP server, database server, email server, file server, print server. Mayoritas dari web layanan tersebut juga merupakan jenis server.

Server yang terlibat dalam jaringan Client-Server

**1). File Servers**

* File server vendors mengklaim bahwa mereka pertama menemukan istilah client-server.
* Untuk sharing file melalui jaringan

**2) Database Servers**

* Client mengirimkan SQL requests sebagai pesan pada database server,selanjutnya hasil perintah SQL dikembalikan.
* Server menggunakan kekuatan proses yang diinginkan untuk menemukan data yang diminta dan kemudian semua record dikembalikan pada client.

**3) Transaction Servers (Transaksi Server)**

* Client meminta remote procedures yang terletak pada server dengan sebuah SQL database engine.
* Remote procedures ini mengeksekusi sebuah grup dari SQL statement
* Hanya satu permintaan / jawaban yang dibutuhkan untuk melakukan transaksi

**4) Groupsware Servers**

* Dikenal sebagai Computer-supported cooperative working
* Manajemen semi-struktur informasi seperti teks, image, bulletin boards dan aliaran kerja
* Data diatur sebagai dokumen

**5) Object Application Servers**

* Aplikasi client/server ditulis sebagai satu set objek komunikasi
* Client objects berkomunikasi dengan server objects melalui Object Request Broker (ORB)
* Client meminta sebuah method pada remote object

**6) Web Application Servers (Aplikasi Web Servers)**

* World Wide Web adalah aplikasi client server yang pertama yang digunakan untuk web.
* Client dan servers berkomunikasi menggunakan RPC seperti protokol yang disebut HTTP.

**Model Arsitektur Klien-Server :**

**I. Arsitektur Mainframe**

Pada arsitektur ini, terdapat sebuah komputer pusat (host) yang memiliki sumber daya yang sangat besar, baik memori, processor maupun media penyimpanan. Melalui komputer terminal, pengguna mengakses sumber daya tersebut. Komputer terminal hanya memiliki monitor/keyboard dan tidak memiliki CPU. Semua sumber daya yang diperlukan terminal dilayani oleh komputer host. Model ini berkembang pada akhir tahun 1980-an.

**II. Arsitektur File Sharing**

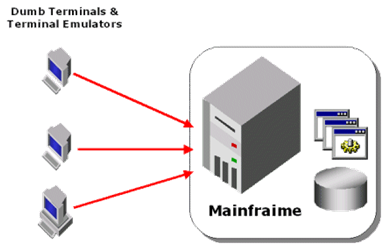
Pada arsitektur ini komputer server menyediakan file-file yang tersimpan di media penyimpanan server yang dapat diakses oleh pengguna. Arsitektur file sharing memiliki keterbatasan, terutama jika jumlah pengakses semakin banyak serta ukuran file yang di shaing sangat besar. Hal ini dapat mengakibatkan transfer data menjadi lambat. Model ini populer pada tahun 1990-an.

**III. Arsitektur Client/Server**

Karena keterbatasan sistem file sharing, dikembangkanlah arsitektur client/server. Salah satu hasilnya yaitu berupa software database server yang menggantikan software database berbasis file server. Dikenalkan pula RDBMS (Relational Database Management System). Dengan arsitektur ini, query data ke server dapat terlayani dengan lebih cepat karena yang ditransfer bukanlah file, tetapi hanyalah hasil dari query tersebut. RPC (Remote Procedure Calls) memegang peranan penting pada arsitektur client/server.

Berdasarkan Kolaborasinya, Arsitektur Client/Server dibagi menjadi 3, yaitu :

1) **Arsitektur Single- Tier (Satu lapis)**

[](https://mildsend.files.wordpress.com/2013/11/one-tier.png)

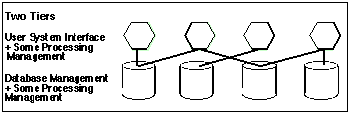
2) Arsitektur Two Tier

Model Two-tier terdiri dari tiga komponen yang disusun menjadi dua lapisan : Client (yang meminta serice) 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.

[](https://mildsend.files.wordpress.com/2013/11/two-tier-1.png)

Hubungan Antar Lapisan dalam Arsitektur Two Tier

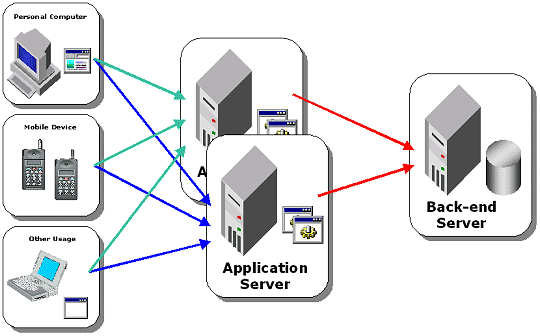
Pada gambar tersebut, user interface yang merupakan bagian dari program aplikasi melayani input dari user. Input tersebut diproses oleh Manajemen Proses dan melakukan query data ke database (dalam bentuk perintah SQL). Pada database server juga bisa memiliki Manajemen Proses untuk melayani query tersebut, biasanya ditulis ke dalam bentuk Stored Procedure.

Pengolahan informasi pada arsitektur ini dapat dibagi menjadi dua, yaitu sistem user interface (antarmuka pengguna) lingkungan dan lingkungan server manajemen database. Arsitektur two tier memiliki tingkat kemanan yang lebih tinggi dan terukur daripada arsitektur single-tier. Arsitektur ini memiliki database pada computer yang terpisah dan hal tersebut menyebabkan arsitektur ini dapat meningkatkan kinerja keseluruhan situs.

**Kelemahan :** biaya mahal, arsitekturnya kompleks, tidak ada pembaruan kode, skalabilitasnya kurang dan tingkat keamanan kurang.

**Kelebihan :** mudah digunakan oleh pengguna, dapat menangani database server secara khusus dan bisnis lingkup kecil sangat cocok menggunakan arsitektur ini.

**3)** **Arsitektur Three-tier (Tiga Lapis)**

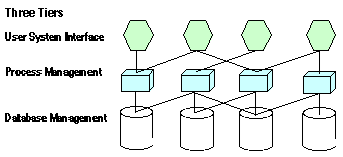
[](https://mildsend.files.wordpress.com/2013/11/tree-tier.png)

Arsitektur Three Tier

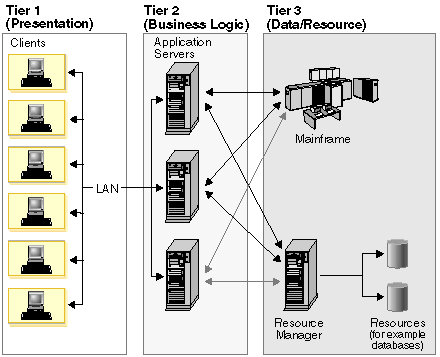
Karena arsitektur sebelumnya memiliki cukup banyak kelemahan, maka dikembangkanlah arsitektur three tier ini yang akan membantu mengatasi kelemahan dari arsitektur two-tier.

Arsitektur three-tier memiliki 3 lapisan.

Pada model ini disisipkan satu layer tambahan diantara user interface tier dan database tier. Tier tersebut dinamakan middle-tier. Middle-Tier terdiri dari bussiness logic dan rules yang menjembatani query user dan database, sehingga program aplikasi tidak bisa mengquery langsung ke database server, tetapi harus memanggil prosedur-prosedur yang telah dibuat dan disimpan pada middle-tier. Dengan adanya server middle-tier ini, beban database server berkurang. Jika query semakin banyak dan/atau jumlah pengguna bertambah, maka server-server ini dapat ditambah, tanpa merubah struktur yang sudah ada. Ada berbagai macam software yang dapat digunakan sebagai server middle-tier. Contohnya MTS (Microsoft Transaction Server) dan MIDAS.

[](https://mildsend.files.wordpress.com/2013/11/tree-tier-1.png)

Hubungan Antar Lapisan dalam Arsitektur Three Tier

[](https://mildsend.files.wordpress.com/2013/11/tree-tier-2.png)

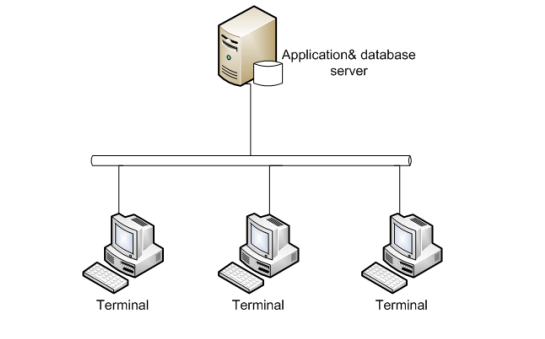
Implementasi Three Tier dalam skala Enterprise

**Kelebihan :**Dapat menangani proses skala besar, transfer informasi antara web server dan server database optimal, tidak akan menyebabkan lapisan lain terkontaminasi jika salah satu lapisan terdapat kesalahan.

**Kekurangan :** Sulit untuk dirancang, sulit untuk di atur dan mahal.

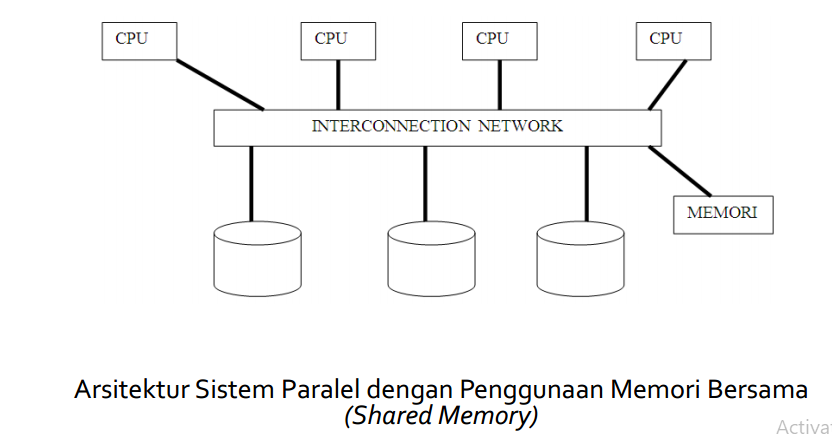
**1.2. Server System Architecture**

* Sistem komputer multi-guna: satu ke banyak CPU dan beberapa alat pengendali yang terhubung melalui sebuah bus yang memungkinkan akses pembagian memori.
* Sistem Multi-user : lebih banyak penyimpan, memori lebih besar, multiple CPU, dan menggunakan SO multi-user. Melayanai banyak pengguna yang terhubung dengan sistem melalui terminal. Biasa disebut sistem *server*.

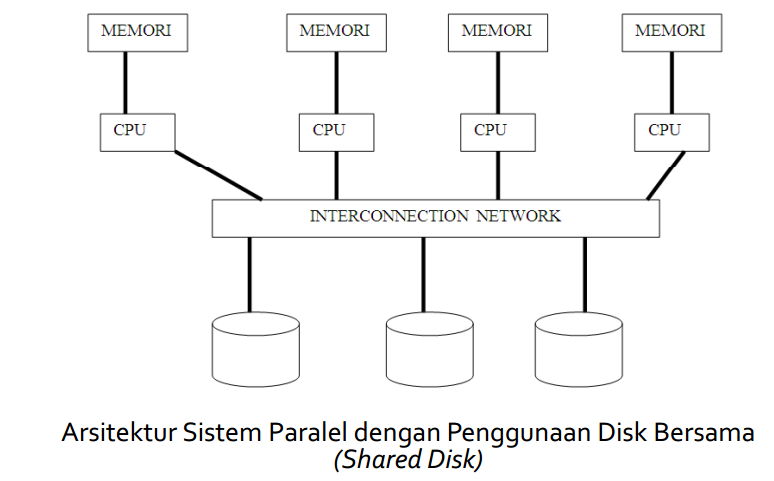


**1.3. Parallel Systems**

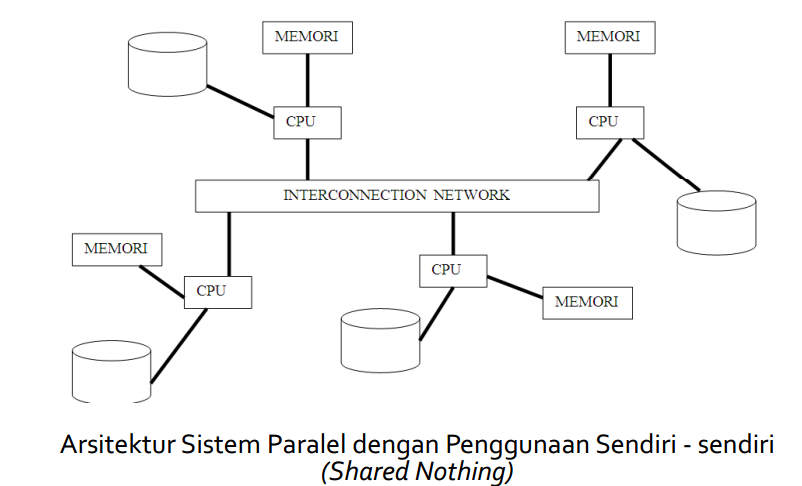
* Sistem ini menggunakan beberapa prosesor dan disk yang dirancang untuk dijalankan secara parallel selama hal tersebut digunakan untuk memperbaiki kinerja database
* Teknologi parallel ini biasanya digunakan untuk basis data yang berukuran sangat besar (terabites) atau system yang memproses ribuan transaksi per detik
* Arsitektur yang digunakan pada system DBMS parallel :
  + Penggunaan memory bersama (share memory)
    - Sebuah arsitektur yang menghubungkan beberapa prosesor di dalam system tunggal yang menggunakan memori secara bersama-sama
    - Arsitektur ini menghasilkan pengaksesan data yang sangat cepat yang dibatasi oleh beberapa prosesor
    - Dekenal dengan nama SMP (Symmetric Multiprocessing) metode ini sering digunakan dalam bentuk workstation personal yang mensupport beberapa mikroprosessor dalam parallel DBMS



* + Penggunaan Disk bersama (Share Disk)
    - Arsitektur yang mengoptimalkan jalannya suatu aplikasi yang tersentralisasi dan membutuhkan keberadaan data dan kinerja yang tinggi
    - Setiap prosesor dapat mengakses langsung semua disk, tetapi prosesor tersebut memiliki memori sendiri-sendiri
    - Arsitektur ini menghapus masalah pada penggunaan memori bersama tanpa harus mengetahui sebuah basis data di partisi. Arsitektur ini dikenal dengan cluster



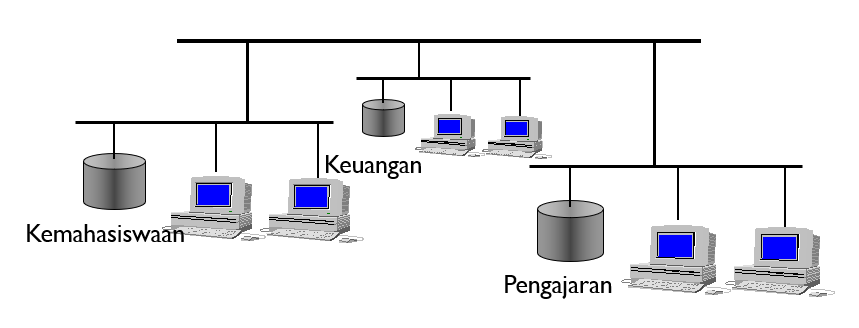
* + Penggunaan secara sendiri-sendiri (Share Nothing)
    - Sering dikenal dengan massively parallel processing yatiu arsitektu dari beberapa prosesor dimana setiap prosesor adalah bagian dari system yang lengkap yang memiliki memory dan disk
    - Arsitektur hamper sama dengan DBMS terdistribusi, namun pendistribusian data pada parallel DBMS hanya berbasis pada kinerja saja



**1.4. Distributed Systems**

Sistem database terdistribusi

* DBMS secara fisik dapat dipisah-pisah, dan diletakkan pada lokasi yang berbeda (sesuai dengan lokasi bagian-bagian organisasi) tetapi saling berhubungan.



Kenapa menggunakan database terdistribusi?

* Organisasi modern yang terdiri dari unit usaha yang terpisah dan berotonomi
  1. Kebutuhan untuk bertukar data antar bagian/departemen
  2. Biaya komunikasi data terlalu mahal jika menggunakan database terpusat
* Integrasi berbagai jenis database perusahaan
* Kebutuhan database duplikat untuk *recovery*

Keuntungan database terdistribusi dibanding terpusat

* Ketersediaan dan Keandalan
  1. Gangguan pada satu bagian masih memungkinkan sistem untuk berfungsi
* Kewenangan Administrasi Lokal
  1. Pembagian tugas dan tanggung-jawab pemeliharaan database lokal bagian
* Berkembang secara Modular
  1. Penambahan komponen baru lebih ekonomis tanpa mengganggu bagian yang lain.

Keuntungan database terdistribusi dibanding terpusat

* Biaya komunikasi rendah
  1. Mayoritas akses adalah akses ke database lokal bagian
* Respon sistem umumnya cepat
  1. Juga, mayoritas akses adalah akses ke database lokal bagian
  2. Tergantung kesesuaian antara pembagian/ distribusi data dengan pola penggunaanya.

Kekurangan database terdistribusi

* Kompleksitas dan Harga DBMS tinggi
* Beban komputasi tinggi untuk koordinasi antar database
* Kompleksitas penjaminan integritas data, terutama untuk pengubahan data
* Respon sistem bisa lambat (jika harus mengakses database lain lewat jaringan)

Bagaimana mendistribusikan data pada database terdistribusi?

1. Replikasi Data

* + Setiap perubahan pada satu database dikopi ke database lainnya secara periodik

2. Partisi Horisontal

* + Distribusi record dari tabel-tabel sesuai dengan pemakaian/kepemilikannya

3. Partisi Vertikal

* + Distribusi kolom/field dari tabel-tabel sesuai dengan pola pemakaian/kepemilikannya

4. Gabungan dari 1, 2 dan 3

**Rangkuman**

* Jadi *arsitektur client-server* adalah pemrosesan data oleh sebuah aplikasi dalam sebuah jaringan yang terdiri dari client dan server yang saling berkomunikasi ketika mengakses sebuah server. Dengan menggunakan arsitektur ini, sistem informasi dapat digunakan dan dibangun dengan perangkat lunak *client-server* yang bermacam-macam dan berbeda-beda.
* **Fungsi 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
* **Fungsi Server**
  + Menerima dan memproses basis data yang diminta dari client
  + Memeriksa autorisasi
  + Menjamin tidak terjadi pelanggaran terhadap integrity constraint
  + Melakukan query/pemrosesan update dan memindahkan response ke client
  + Memelihara data dictionary
* Sistem Multi-user : lebih banyak penyimpan, memori lebih besar, multiple CPU, dan menggunakan SO multi-user. Melayanai banyak pengguna yang terhubung dengan sistem melalui terminal. Biasa disebut sistem *server*.
* Paralel database adalah Sistem yang menggunakan beberapa prosesor dan disk yang dirancang untuk dijalankan secara parallel selama hal tersebut digunakan untuk memperbaiki kinerja database
* Sistem database terdistribusi adalah ketika DBMS secara fisik dapat dipisah-pisah, dan diletakkan pada lokasi yang berbeda (sesuai dengan lokasi bagian-bagian organisasi) tetapi saling berhubungan

**Latihan**

1. Apa perbedaan antara 1 tier, 2 tier dan 3 tier ?
2. Sebut dan jelaskan apa saja yang termasuk dalam arsitektur Paralel system !
3. Apa saja kelebihan dan kekurangan system database terdistribusi ?
4. Apa saja kelebihan dan kekurangan system database client server ?
5. Jelaskan jenis server pada jaringan database client server !

**HALAMAN TERAKHIR MODUL**



**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS BUDI LUHUR**

Jl. Raya Ciledug, Petukangan Utara, Pesanggrahan

Jakarta Selatan, 12260

Telp: 021-5853753 Fax : 021-5853752

http://fti.budiluhur.ac.id