

DBMS ARCHITECTURE

Ir.H Vierry Kurniawan (TI2B) 2131740011

Soal

1. Apa yg dimaksud dengan Arsitektur Client Server dan apa keuntungannya? Bandingkan Client-server dengan 2 arsitektur lainnya
2. Bandingkan dan beri perbedaan pada arsitektur DBMS tradisional dengan Three Tier Client Server dan mana yang lebih cocok untuk Web?
3. Apakah Server aplikasi lebih baik dibandingkan dengan File Server? Mengapa?

Jawaban

1. CLIENT SERVER

Client Server adalah metode penyelesaian masalah pada aplikasi (software) yang menggunakan sistem database sehingga perangkat komputer tidak perlu diinstall database. Database client server bisa diinstall pada komputer server sedangkan aplikasinya diinstall pada pihak client.

Keuntungan Client Server

- Memungkinkan akses luas ke database yang ada
- Peningkatan kinerja: jika klien server berada di komputer berbeda maka CPU dapat memproses aplikasi secara paralel
- Biaya Hardware dapat dikurangi (hanya perlu biaya besar di Server)
- Biaya komunikasi berkurang: Aplikasi dijalankan di Klien dan hanya mengirim jika ada permintaan akses database.
- Peningkatan konsistensi : Server menangani pemeriksaan, definisi dan validasi di satu tempat

TELEPROCESSING

Teleprocessing merupakan arsitektur tradisional yang terdiri satu CPU dan banyak terminal. Terminal tidak mampu berfungsi sendiri dan harus disambungkan ke pusat.

Kelemahan :

- Beban berat pada computer pusat krn harus menjalankan Aplikasi dan DBMS.

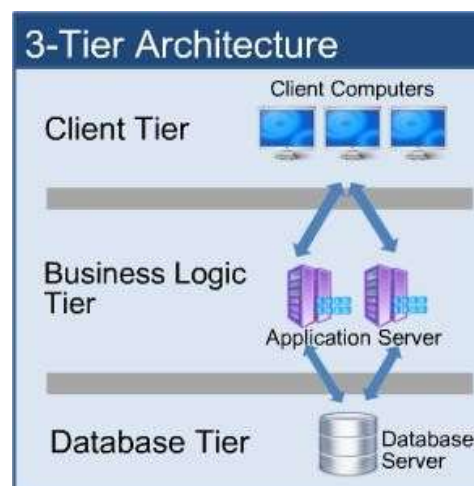
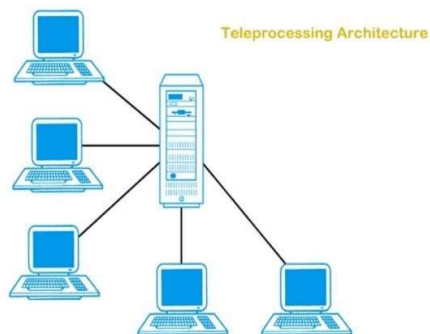
APPLICATION SERVER

Application Server merupakan antar muka untuk API pada bagian Business Logic dan Business Process untuk digunakan aplikasi lain.

Masalah yang harus dihandle di titik ini :

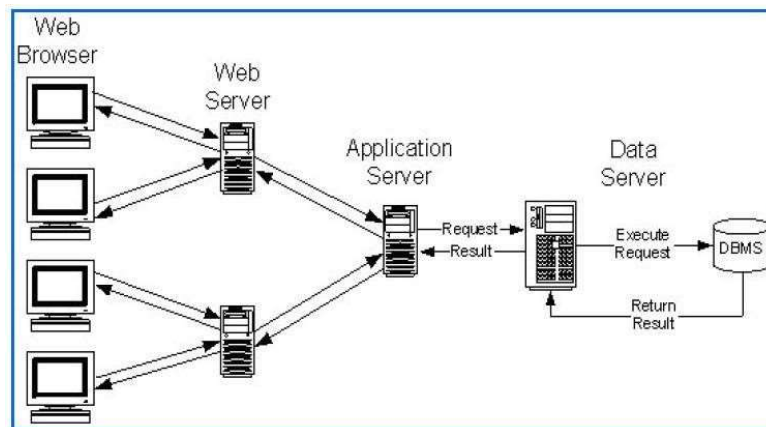
- Konkurensi
- Manajemen Koneksi Jaringan
- Menyediakan Akses ke Semua Server DB
- Penyatuan Koneksi DB
- Legacy DB Support, Clustering Support,
- Load Balancing dan Fail over

2. Perbandingan dan Perbedaan pada arsitektur DBM Tradisional (Teleprocessing) dengan Three Tier Client Server.



Dari dua gambar arsitektur di atas terdapat perbedaan yang signifikan yaitu terdapat pembagian dari arsitektur *three tier architecture*, sedangkan tidak terdapat pembagian di DBMS tradisional. Jika kedua arsitektur di atas dibandingkan maka sudah jelas bahwa *three tier architecture* lebih unggul dari segi pemeliharaan database dan server. Karena *three tier architecture* merupakan pengembangan Arsitektur DBMS yang sangat baik dari arsitektur Teleprocessing.

3. Server aplikasi dan file server lebih baik dipisah karena untuk menyeimbangkan muatan (*load balancing*) yang efisien. Seperti yang kita tahu bahwa tugas dari file server dan aplikasi server sangatlah berbeda. Oleh karena itu, membedakan kedua jenis tahapan (*tier*) dalam kasus ini sangatlah penting. Contoh sebagai berikut,



Jika kita lihat gambar di atas, aplikasi server bertugas untuk melakukan *request* data ke file server dan menghasilkan sebuah *result*. Perlu dilihat sebelumnya bahwa, aplikasi server menerima *request* dari banyak web server atau tidak hanya dari satu web server. Jika aplikasi server menjadi satu dengan file server maka kemungkinan yang terjadi adalah *overloading* pada file server.

Jika *overloading* terjadi, maka dampak yang akan ditimbulkan adalah lambatnya pengiriman data *result* dan *overheat* pada aplikasi server maupun data server. Namun, membedakan file server dan aplikasi server harus memperhitungkan beberapa hal.

Contohnya biaya atau *costs* yang banyak dan kerumitan pengkoneksian antar server.

1. Beri contoh dari masing-masing type of middleware

Jawab:

- A. Message Oriented Middleware (MOM)
- B. Object Middleware
- C. Remote Procedure Call (RPC) Middleware
- D. Database Middleware
- E. Transactional Middleware
- F. Robotic Middleware
- G. Middleware Integrasi

2. Beri contoh dari Web Service

Jawab :

- A. Apache HTTP Server
- C. Microsoft Internet Information Service (IIS)
- D. Lighttpd

3. Teknologi dan standar apa yang digunakan untuk mengembangkan layanan web dan hubungannya satu sama lain ?

Jawab :

Ada 4 macam teknologi yang digunakan untuk mengembangkan layanan web dan hubungannya satu sama lain.

- 1) **SOAP**, merupakan protokol standar untuk saling bertukar informasi antar aplikasi.
- 2) **REST**, merupakan arsitektur dari sebuah sistem yang terstandarisasi antar sistem komputer, dengan teknologi ini memungkinkan antar sistem saling berkomunikasi.

3) **WSDL (*Web Service Definition Language*)**, merupakan sebuah file yang berformat XML yang fungsi utamanya menjelaskan sisi client apa saja isi dari sebuah website.

4) **WDDX (*Web Distributed Data Exchange*)**, teknologi ini memungkinkan pertukaran dengan format yang berbeda.