



INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

HEART & MIND TOWARDS EXCELLENCE

Sistem Terdistribusi

Proses

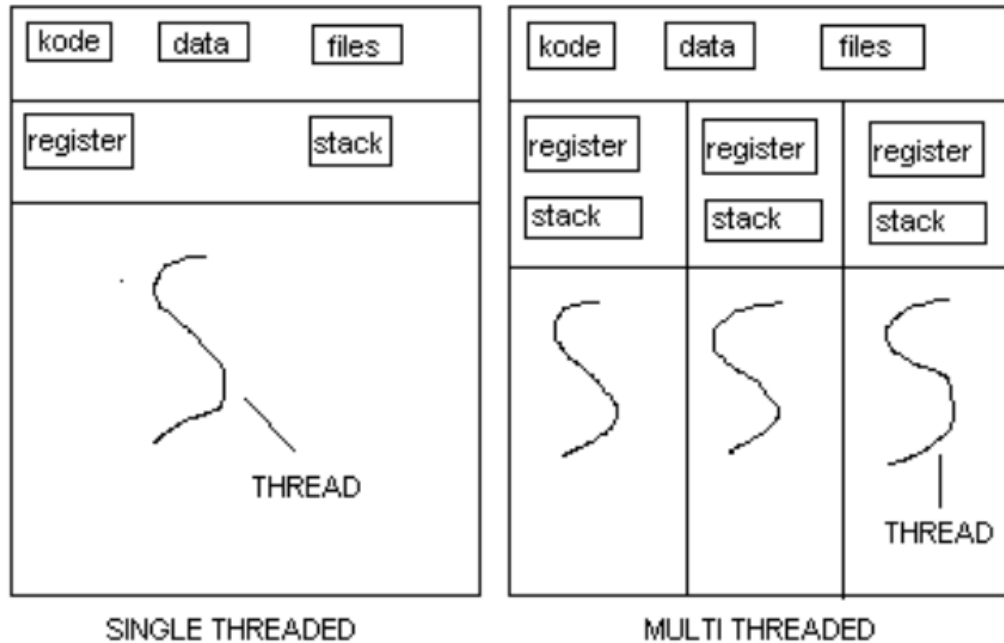
S1 Sistem Komputer
Musayyanah, S.ST, MT

Apa itu Thread ?

- Sebuah alur control dari sebuah proses.
- Contohnya : Sebuah web browser mempunyai thread untuk menampilkan gambar atau tulisan sedangkan thread yang lain berfungsi sebagai penerima data dari network
- Diklasifikasikan menjadi :
 - 1. Single Thread
 - 2. Multithread

perbedaan thread dengan proses

- thread berbeda dengan proses karena thread memiliki address space yang sama yang artinya mereka berbagi variable global. karenanya masing2 thread dapat saling mengakses virtual address, membaca, menulis bahkan menghentikan thread lain. sedangkan proses bisa dimiliki oleh user
- proses menunggu atau ditahan proses yang lain. proses selalu dimiliki oleh satu user sehingga mereka dapat saling bekerja sama tidak saling berbenturan.



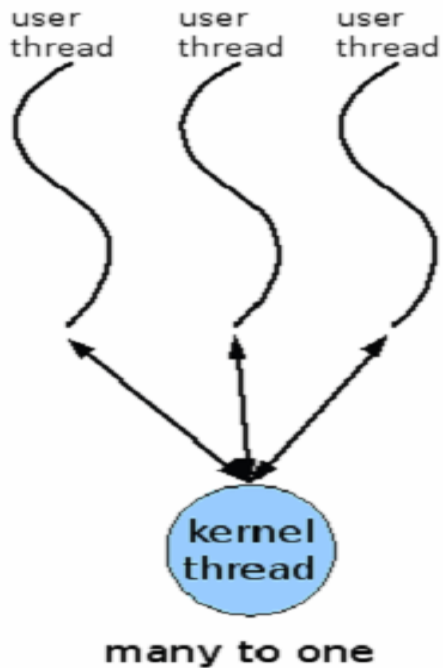
- Banyak perangkat lunak yang berjalan pada PC modern dirancang secara multithread
- Contohnya : sebuah web server diakses oleh ratusan client. Jika web server sbg proses single thread, maka web server hanya bisa melayani satu client pada satu satuan waktu.
- Satu client yang lain harus menunggu untuk pengajuan permintaan, sampai permintaan client sebelumnya terselesaikan. (Solusi : membuat web server menjadi multi-threading)

Keuntungan Multithread

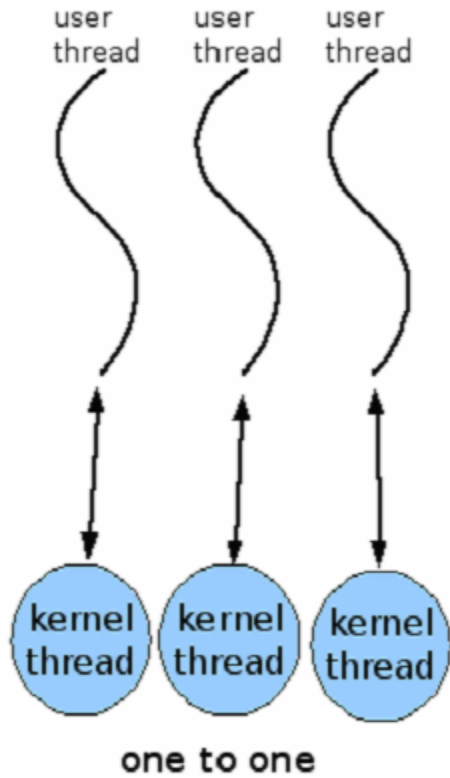
- Responsi : membuat sebuah program berjalan meskipun sebagian dari program tersebut diblok.
Contoh : web browser akan memiliki sebuah thread melayani permintaan pengguna sementara thread yang lain berusaha menampilkan image
- Berbagi sumber daya : memungkinkan sebuah aplikasi untuk mempunyai beberapa thread yang berbeda dalam lokasi memori yang sama
- Ekonomi : menghemat sumber daya proses
- Utilasi Arsitektur Multiprocessor : setiap thread dapat berjalan secara parallel di atas prosesor yang berbeda

Model-Model Multithread

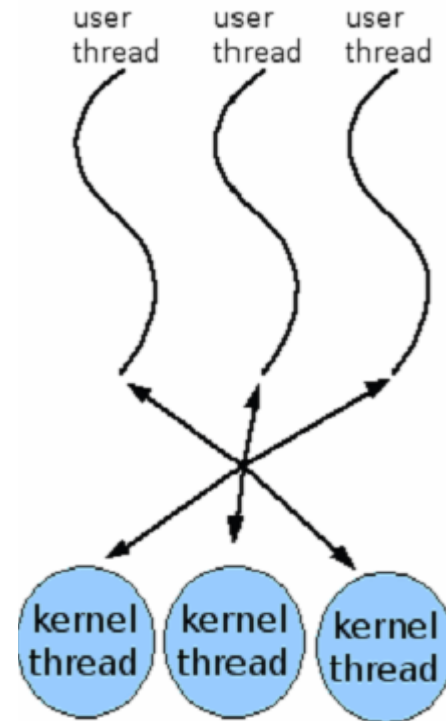
- Many to One : tidak dapat berjalan secara parallel pada multiprocessor



- Model One to One



Model Many to Many



Client Server

- Sistem client/server harus terdapat satu atau beberapa server yang menyediakan layanan atau satu atau beberapa klien yang meminta layanan tersebut (tidak peduli apakah kondisi tersebut berada pada sebuah sistem jaringan ataupun stand alone).



- Server : komputer yang dapat memberikan service ke client.
- Client : komputer yang mengakses beberapa service yang ada di server

- Karakteristik Server

- a. Pasif

- b. Menunggu request

- c. Menerima request, memproses dan mengirimkan balasan berupa service

- Karakteristik Client

- a. Aktif

- b. Mengirim request

- c. Menunggu dan menerima balasan dari server

Socket

- Sebuah endpoint untuk komunikasi di dalam jaringan
- Dibuat menyambungkan dua buah alamat IP melalui port tertentu.
- Digunakan dalam sistem client/server
- Jika ada client yang menghubungi server maka server akan menyetujui komunikasi dengan client melalui socket yang dibangun

Model Client Server

- **Arsitektur Mainframe**

- terdapat sebuah host yang memiliki sumber daya yang sangat besar, baik memori, processor maupun media penyimpanan.
- Client mengakses sumber daya tersebut melalui komputer terminal (hanya memiliki keyboard dan monitor dan tidak memiliki CPU).
- Semua sumber daya yang diperlukan terminal dilayani oleh komputer host.

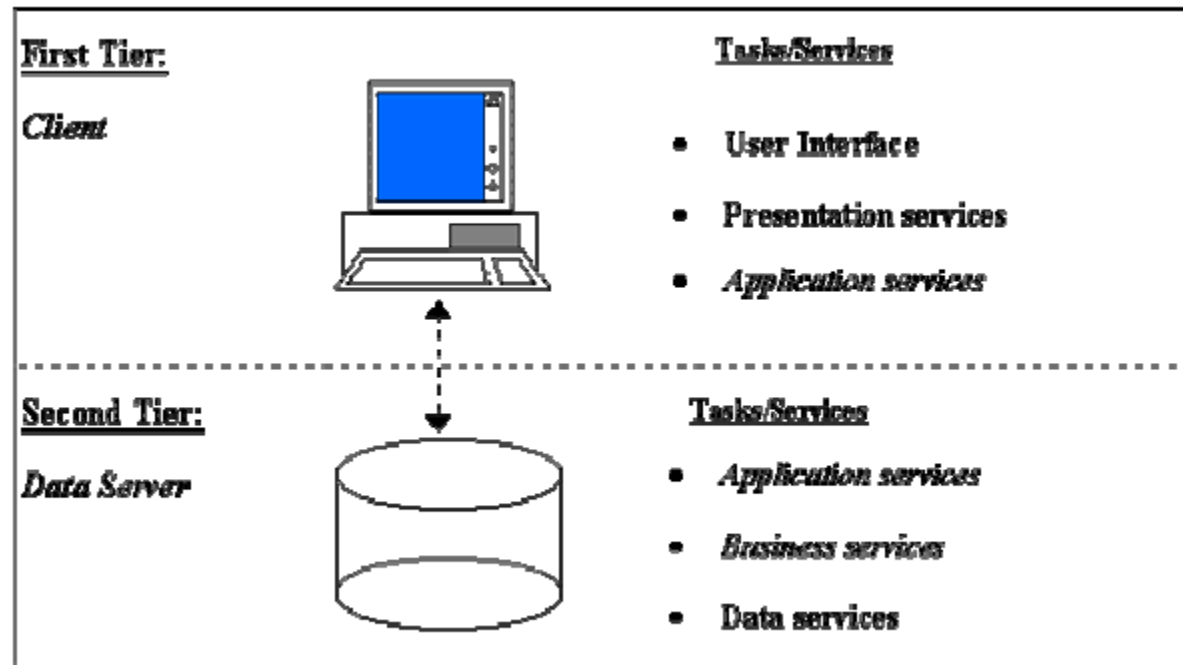
- **Arsitektur File Sharing**

- Komputer server menyediakan file-file yang tersimpan di media penyimpanan server yang dapat diakses oleh pengguna.
- Transfer data lambat jika jumlah pengakses banyak dan ukuran file besar

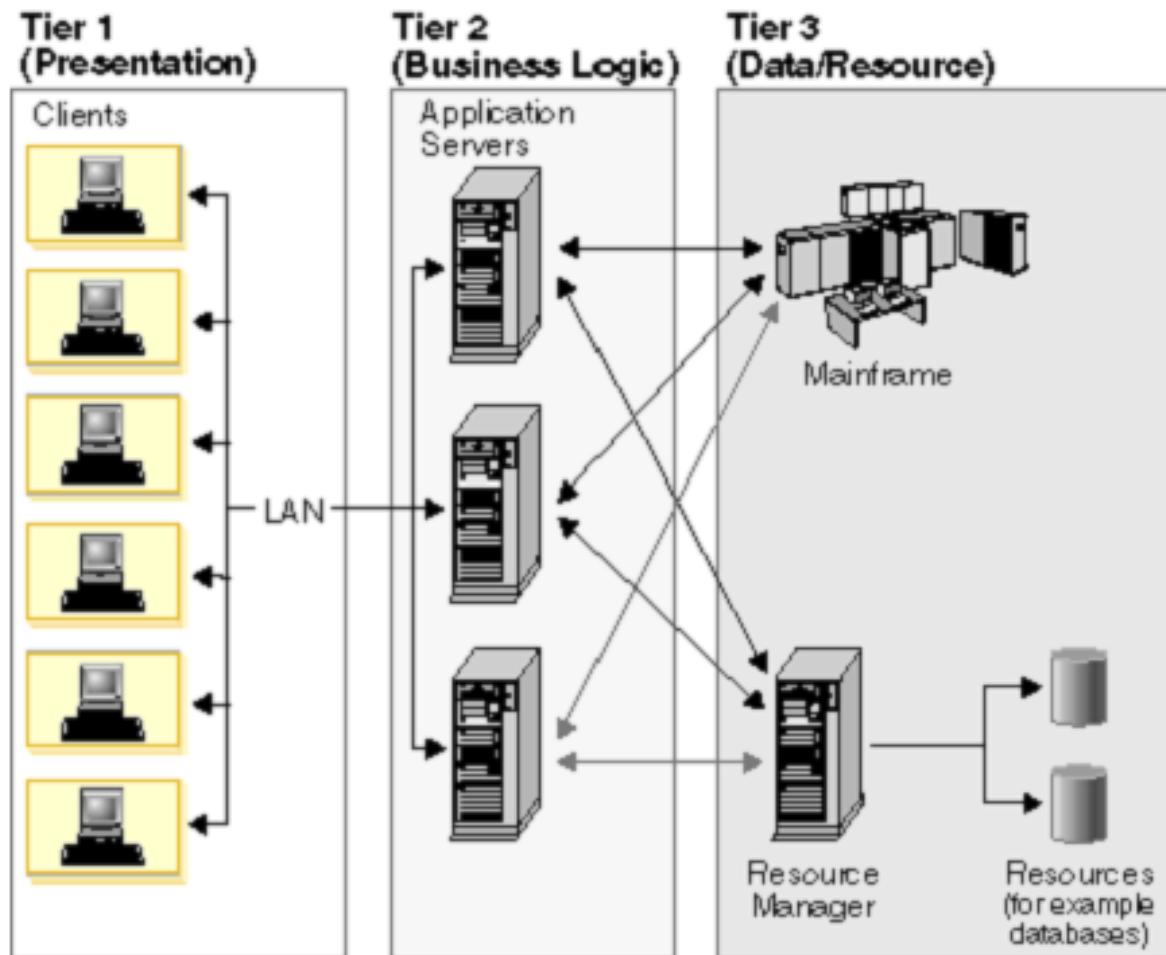
- **Arsitektur Client/Server**

- Seperti software database server yang menggantikan software database berbasis file server.
- Query data ke server dapat terlayani lebih cepat karena yang ditransfer bukan file, tetapi hasil dari query tersebut.

- **Model Two Tier**
- Terdiri dari 3 komponen :
 1. User Interface
 2. Manajemen Proses
 3. Database



- Model Three-Tier
- Disebut juga **middle-tier**



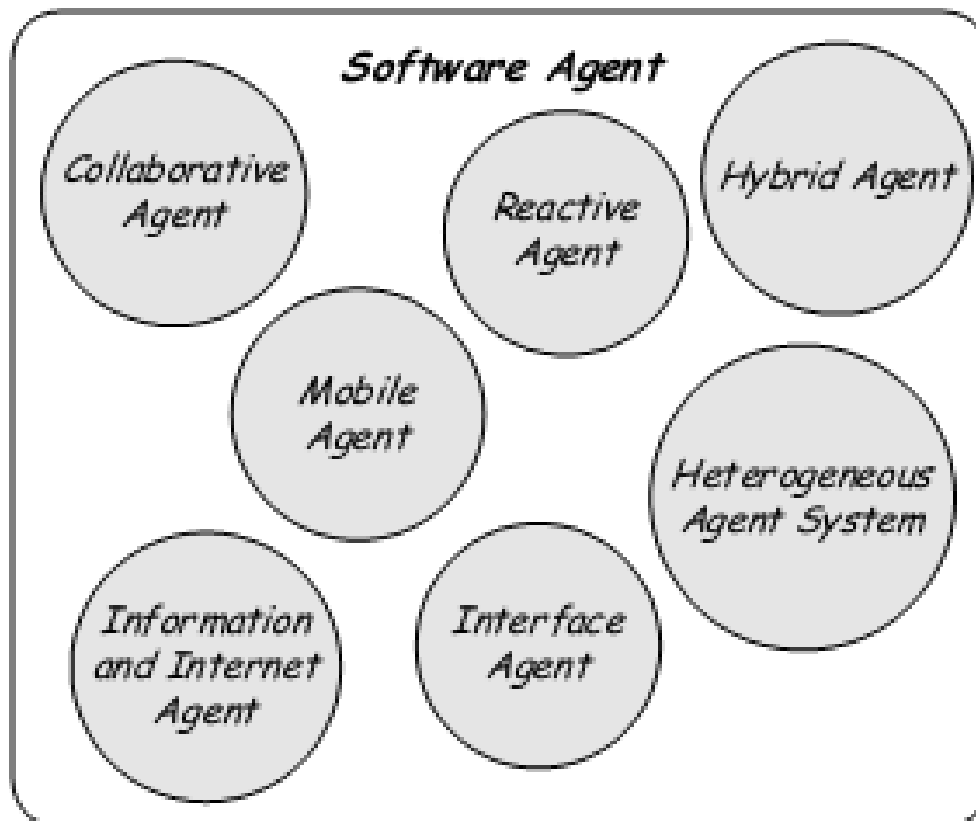
Agent

- Disebut software agent
- Entitas perangkat lunak yang didedikasikan untuk tujuan tertentu yang memungkinkan user untuk mendelegasikan tugasnya secara mandiri.
- **Karakteristik dari Agent**
- **Autonomy** : melakukan tugas sendiri dan tidak dipengaruhi oleh user, agent lain ataupun oleh lingkungan
- **Intelligence, Resoning, dan Learning**
- **Mobility dan Stationary** : agent mobility dan agent stationary memiliki kemampuan untuk mengirim pesan dan berkomunikasi dengan agent lain

- Delegation : agent bergerak dalam rangka menjalankan tugas yang diperintahkan oleh user
- Reactivity : kemampuan untuk bisa cepat beradaptasi dengan adanya perubahan informasi yang ada dalam suatu lingkungan
- Proactivity dan Goal Oriented: Agent harus didesain memiliki goal yang jelas, dan selalu berorientasi kepada tujuan yang diembannya.
- Communication and Coordination Capability : Agent harus mempunyai kemampuan untuk dapat berkomunikasi dengan agent yang lain yang harus memperhatikan masalah user interface dan perangkatnya.

Klasifikasi Software Agent

- Klasifikasi menurut Karakteristik yang dimiliki



- Klasifikasi menurut Lingkungan dimana dijalankan
 - a. Dekstop Agent : bertugas di dalam lingkungan PC (Personal Computer) dan berjalan di atas suatu Operating System (OS)
 - b. Internet Agent : bertugas yang bertugas di lingkungan jaringan Intrenet, melakukan tugas memanage infrmasi yang ada di Internet.
 - c. Intranet Agent : bertugas dalam lingkungan jaringan Intranet

Bahasa Pemrograman yang digunakan

- Object Orientedness : bahasa pemrograman yang berbasis object disingkat OOP (Object Oriented Programming) ex : JAVA, C++, Pascal, PHP, Delphi, Python
- Platform Independence : program harus bisa dijalankan di platform manapun. Ex : bahasa Java dapat dengan mudah dipindahkan antar berbagai jenis sistem operasi dan berbagai jenis arsitektur komputer.
- Communication Capability : bahasa pemrograman yang dapat mendukung pemrograman yang berbasis network dan komunikasi

- Security : bahasa pemrograman yang mensupport level-level keamanan yang bisa membuat agent bergerak dengan aman.
- Code Manipulation : memerlukan manipulasi kode program secara runtime, shg diperlukan bahasa pemrograman untuk software yg dpt menangani masalah runtime tersebut

Referensi

- 1. <http://bebas.vlsm.org/v06/Kuliah/SistemOperasi/BUKU/SistemOperasi-4.X-1/ch11.html#c31101>
- 2. <http://bebas.vlsm.org/v06/Kuliah/SistemOperasi/2002/riene101/produk/Materi/Threads.html>
- 3. Rofiq Siregar, Definisi Client/Server, <http://rofiqsiregar.wordpress.com/2007/05/09/defenisi-clientserver/>
- 4. <http://bebas.vlsm.org/v06/Kuliah/SistemOperasi/BUKU/SistemOperasi-4.X-1/ch17s05.html>
- 5. Rofiq Siregar, Model Client/Server, <http://rofiqsiregar.wordpress.com/2007/05/29/model-clientserver/>
- 6. Ayu Anggriani dkk., Tugas Kuliah Pengantar Sistem Terdistribusi, 2008.
- 7. Romi Satria Wahono, Pengantar Software Agent: Teori dan Aplikasi, [http://ilmukomputer.com/2006/08/21/pengantar-**software-agent**/](http://ilmukomputer.com/2006/08/21/pengantar-software-agent/)