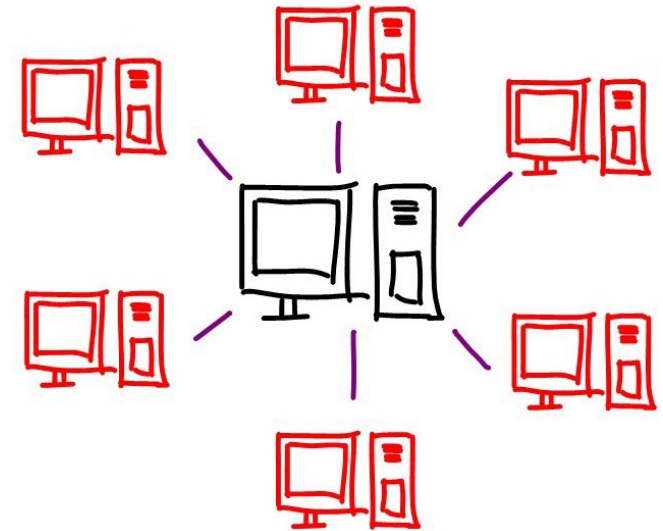


Sistem Terdistribusi (Distributed System)

Pengantar sistem terdistribusi

Henry Saptono, M.Kom



Course/Slides Credits

- **Catatan:** *semua slide presentasi didasarkan pada buku yang dikembangkan oleh Andrew S. Tanenbaum dan Maarten van Steen. Judul buku mereka "Distributed Systems: Principles and Paradigms" (edisi 1 & 2). Dan juga berdasarkan slide presentasi yang dibuat oleh Maarten van Steen (VU Amsterdam, Dept. Computer Science)*

Definisi sistem terdistribusi

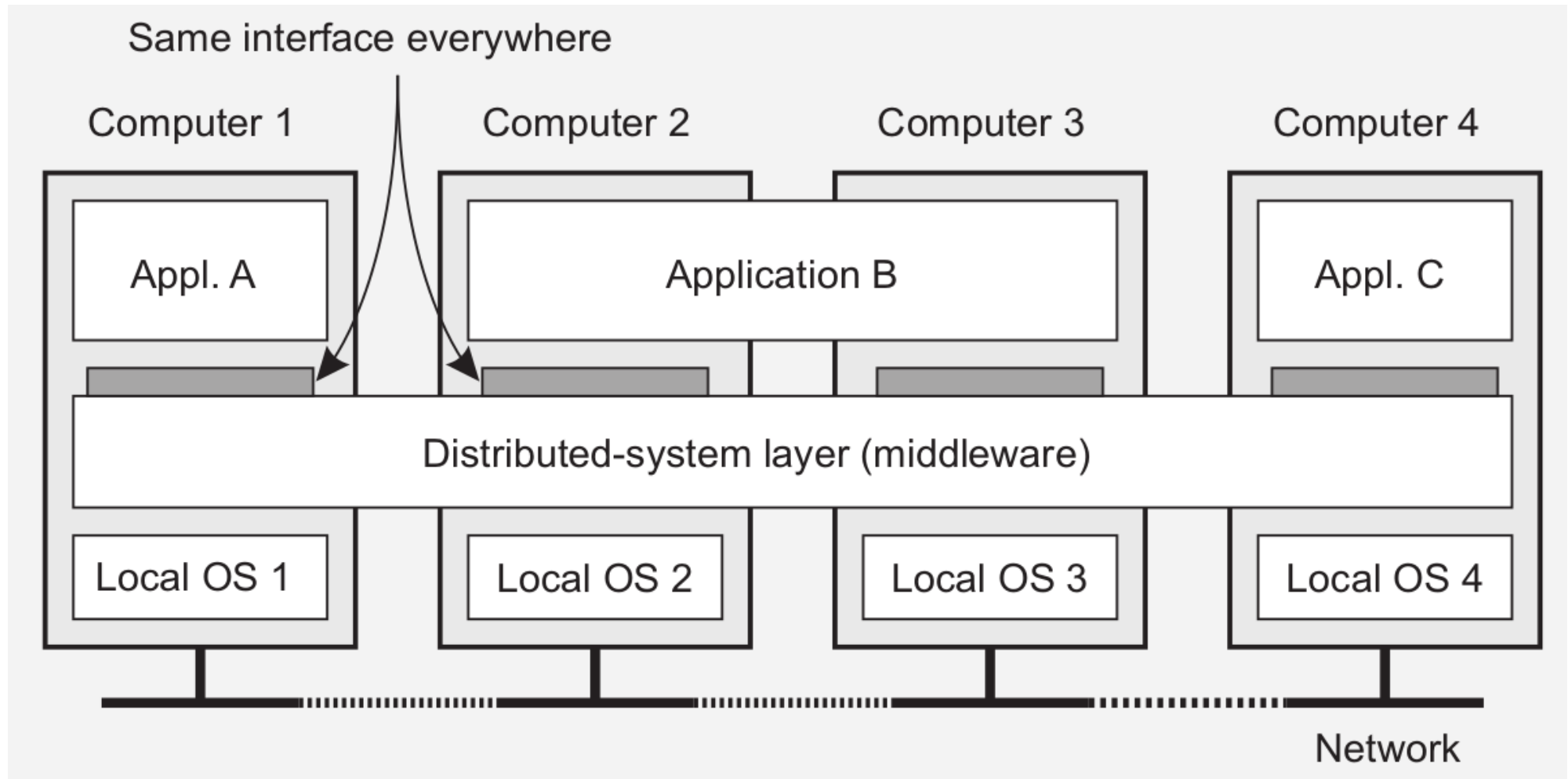
- **Sistem terdistribusi** adalah :
*kumpulan **elemen komputasi otonom** yang muncul atau tampak bagi penggunanya sebagai **sistem koheren tunggal***
(Distributed Systems: Principles and Paradigms)
- Dua aspek:
 - (1) elemen komputasi independen dan
 - (2) sistem tunggal \Rightarrow middleware.

Definisi sistem terdistribusi

- Definisi lain, **Sistem terdistribusi** adalah :
 - *dimana komponen perangkat keras atau perangkat lunak yang terletak di jaringan komputer berkomunikasi dan mengoordinasikan tindakan mereka hanya dengan mengirimkan pesan.*

(DISTRIBUTED SYSTEMS - Concepts and Design 5th edition)

Definisi sistem terdistribusi



Tujuan sistem terdistribusi

- **Menyediakan sumber daya (*Making resources available*).**
 - membuat banyak sumber daya tersedia untuk banyak pengguna / aplikasi;
 - Dan harus membagikannya secara terkontrol & efisien;
- **Transparansi distribusi (*Distribution transparency*)**
 - harus menyembunyikan aspek fisik dari proses / distribusi sumber daya DS;
 - mis. akan muncul sebagai satu sistem;

Tujuan sistem terdistribusi

- **Keterbukaan (*Openness*)**
 - harus menawarkan layanan sesuai dengan aturan standar;
 - Dan harus menjelaskan sintaks dan semantik layanan layanan tersebut;
- **Skalabilitas (*Scalability*)**
 - ukuran (jumlah pengguna & / atau proses)
 - geografis (jarak antar node)
 - admin (jumlah domain administratif)

Transparansi distribusi

Transp.	Description
Access	Hide differences in data representation and how an object is accessed
Location	Hide where an object is located
Relocation	Hide that an object may be moved to another location while in use
Migration	Hide that an object may move to another location
Replication	Hide that an object is replicated
Concurrency	Hide that an object may be shared by several independent users
Failure	Hide the failure and recovery of an object

Transparansi distribusi

Perlu diperhatikan, tujuan transparansi distribusi penuh mungkin bisa jadi bukan ide yang bagus:

- Pengguna mungkin terdapat di berbagai benua
- Menyembunyikan kegagalan total jaringan dan node adalah (teoritis dan praktis) tidak mungkin
 - Anda tidak dapat membedakan komputer yang lambat dari yang gagal
 - Anda tidak pernah dapat memastikan bahwa server benar-benar telah melakukan operasi sebelum crash
- Transparansi penuh akan membebani kinerja

Keterbukaan sistem terdistribusi

- Sistem terdistribusi terbuka
 - Mampu berinteraksi dengan layanan dari sistem terbuka lainnya, terlepas dari lingkungan yang mendasarinya:
 - Sistem harus sesuai dengan antarmuka yang terdefinisi dengan baik
 - Sistem harus mendukung portabilitas aplikasi
 - Sistem harus mudah dioperasikan



Keterbukaan sistem terdistribusi

- Mencapai keterbukaan
 - Setidaknya membuat sistem terdistribusi bebas dari heterogenitas dari lingkungan yang mendasarinya:
 - Perangkat keras
 - Platform (os)
 - Bahasa

Skala dalam sistem terdistribusi

- **Skalabilitas** setidaknya terdiri dari tiga komponen:
 - Jumlah pengguna dan / atau proses (**skalabilitas ukuran**)
 - Jarak maksimum antar node (**skalabilitas geografis**)
 - Jumlah domain administratif (**skalabilitas administratif**)

Teknik untuk penskalaan (scaling)

- **Menyembunyikan latensi komunikasi**
 - Menghindari menunggu tanggapan (respon); dengan melakukan sesuatu yang lain:
 - Memanfaatkan komunikasi bersifat asinkron
 - Memiliki penanganan terpisah untuk tanggapan yang datang
 - **Masalah:** tidak semua aplikasi cocok dengan model ini

Teknik untuk penskalaan

- **Distribusi**
 - Membagi data (partisi) dan proses komputasi ke beberapa mesin:
 - Memindahkan proses komputasi ke klien (**Applet Java**)
 - Layanan penamaan terdesentralisasi (**DNS**)
 - Sistem informasi terdesentralisasi (**WWW**)

Teknik untuk penskalaan

- **Replikasi / caching**
 - Membuat salinan data tersedia di mesin yang berbeda:
 - Replikasi File dan database
 - Mirroring web
 - Web cache (di browser dan proxy)
 - Caching file (di server dan klien)

Scaling dan Permasalahannya

- Menerapkan teknik scaling (penskalaan) itu mudah, kecuali untuk satu hal:
 - Dengan memiliki banyak salinan (di-cache atau direplikasi), mengarah pada inkonsistensi: dengan memodifikasi satu salinan menjadikan salinan itu berbeda dari yang lain.
 - Selalu menjaga salinan konsisten dan secara umum mengharuskan sinkronisasi global pada setiap modifikasi.
 - Sinkronisasi global menghalangi solusi-solusi skala besar.

Mengembangkan sistem terdistribusi: Perangkat

- Sistem terdistribusi bisa menjadi rumit dan menghasilkan kekurangan-kekurangan yang perlu ditambah di kemudian waktu. Ini terjadi akibat ada **asumsi yang salah** ketika merancang sistem terdistribusi, yaitu asumsi bahwa:
 - Jaringanannya andal
 - Jaringanannya aman
 - Jaringanannya homogen
 - Topologi tidak pernah berubah
 - Latensi adalah nol
 - Bandwidth tidak terbatas
 - Transport cost is zero
 - Hanya ada satu pengelola (pengelolaan)



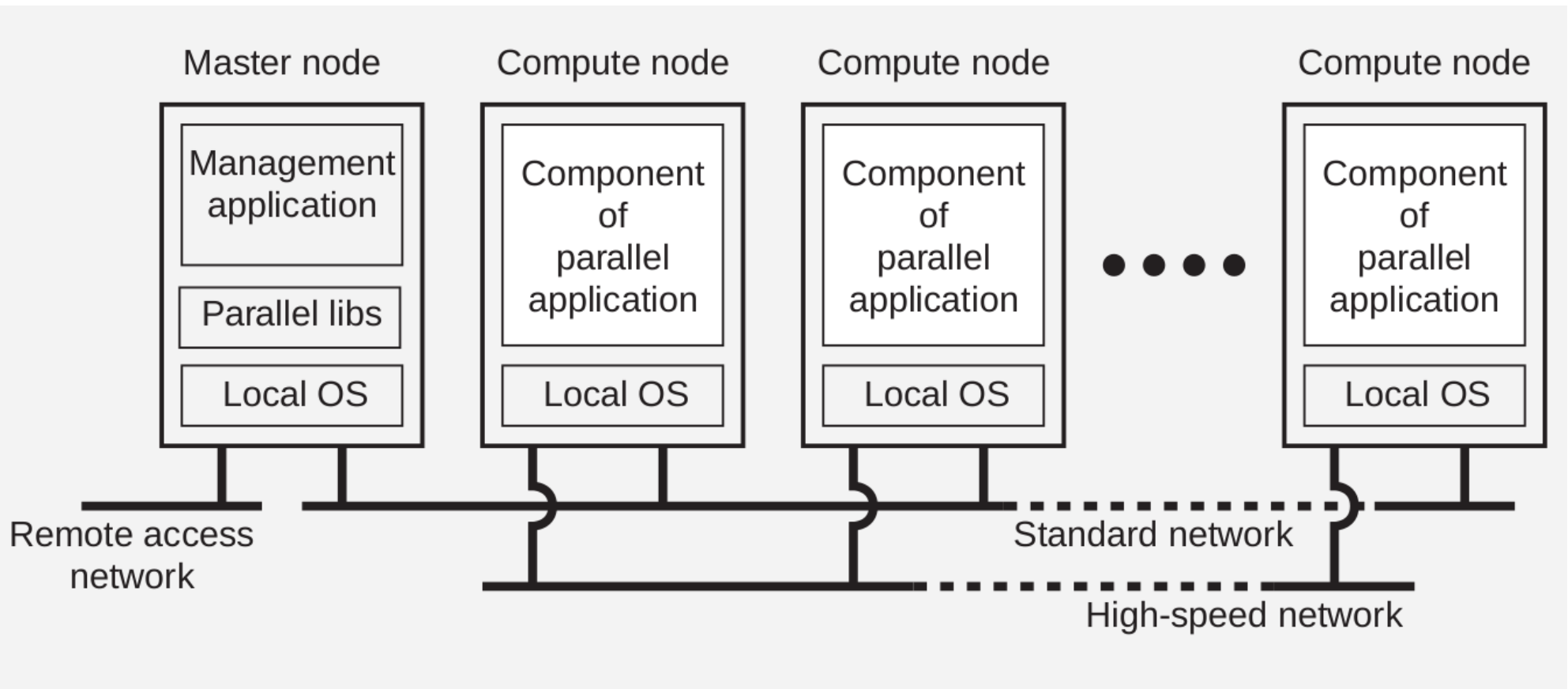
Jenis-jenis sistem terdistribusi

- Distributed computing systems
- Distributed information systems
- Distributed pervasive systems

Distributed computing systems

- Banyak sistem terdistribusi yang dikonfigurasi untuk Komputasi Kinerja Tinggi (High Performance Computing = HPC)
- **Komputasi Cluster (*Cluster Computing*)**
 - Pada dasarnya sekelompok sistem *high-end* yang terhubung melalui LAN:
 - **Homogen**: OS yang sama, perangkat yang hampir identik
 - **Node pengelolaan tunggal**

Distributed computing systems: cluster computing

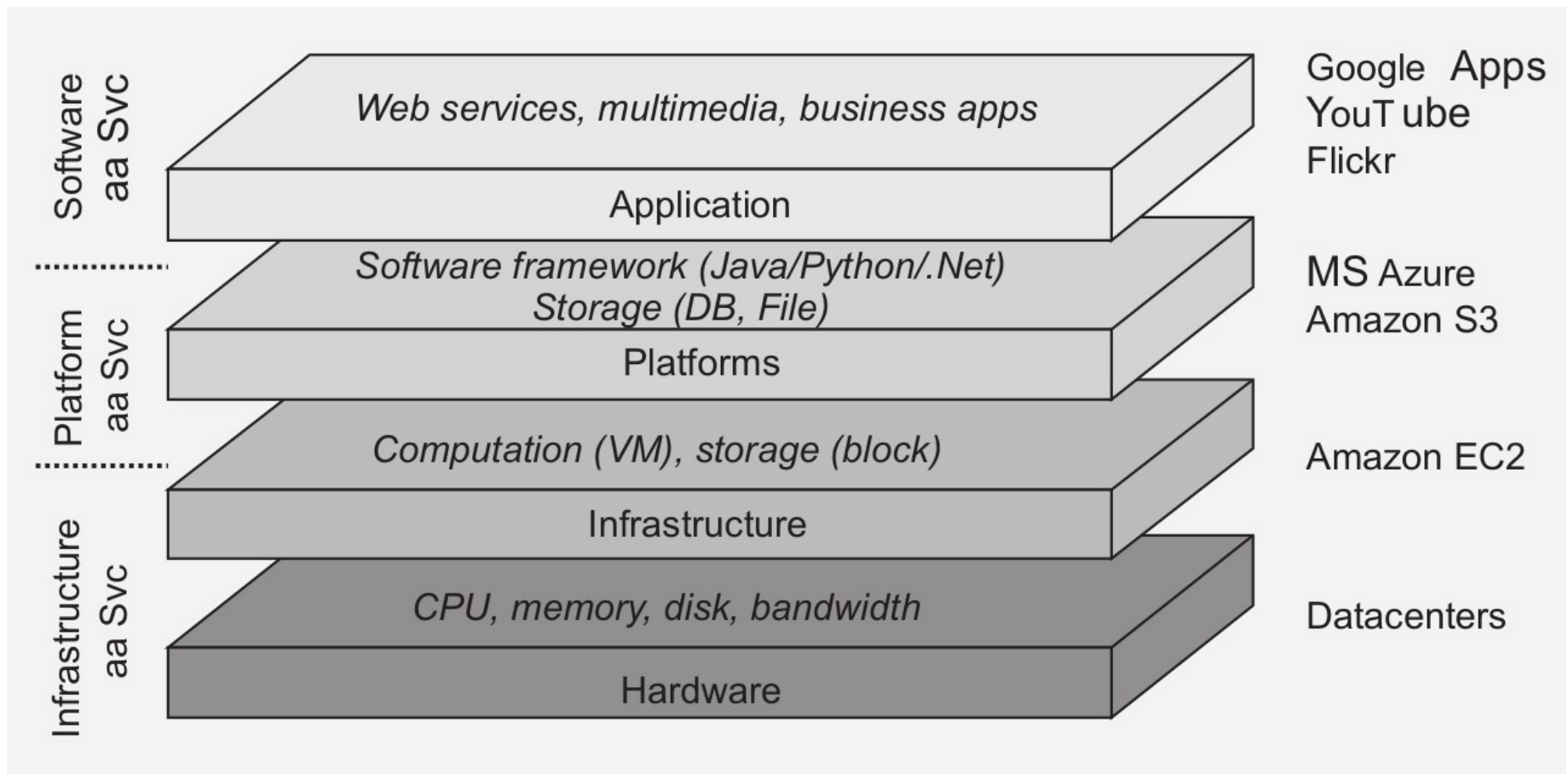


Distributed computing systems

- **Komputasi Grid (*Grid Computing*)**
 - Terdiri dari banyak node (from everywhere):
 - **Heterogen**
 - Tersebar di beberapa organisasi
 - Dapat dengan mudah menjangkau jaringan area yang luas

Distributed computing systems

- **Komputasi Cloud (*Cloud Computing*)**



Distributed Information Systems: Transaction processing system

- Banyak sistem terdistribusi yang digunakan saat ini adalah bentuk-bentuk dari sistem informasi tradisional, yang sekarang mengintegrasikan sistem lama (*legacy*). Contoh: Sistem pemrosesan transaksi (*Transaction processing systems*)
- Transaksi adalah kumpulan operasi pada objek (database, komposisi objek, dll.) Yang memenuhi properti berikut (ACID)

Distributed Information Systems: Transaction processing system

- **Atomicity:** Semua operasi berhasil, atau semuanya gagal. Ketika transaksi gagal, keadaan objek akan tetap tidak terpengaruh oleh transaksi.
- **Consistency:** Transaksi menetapkan transisi status yang valid. Ini tidak mengecualikan kemungkinan kondisi perantara yang tidak valid selama eksekusi transaksi.
- **Isolation:** Transaksi bersamaan tidak saling mengganggu. Tampaknya untuk setiap transaksi T bahwa transaksi lain terjadi baik sebelum T, atau setelah T, tetapi tidak pernah keduanya.
- **Durability:** Setelah pelaksanaan transaksi, efeknya dibuat permanen



Distributed Information Systems: Transaction processing system

BEGIN TRANSACTION(server, transaction)

READ(transaction, file-1, data)

WRITE(transaction, file-2, data)

newData := MODIFIED(data)

IF WRONG(newData) THEN

 ABORT TRANSACTION(transaction)

ELSE

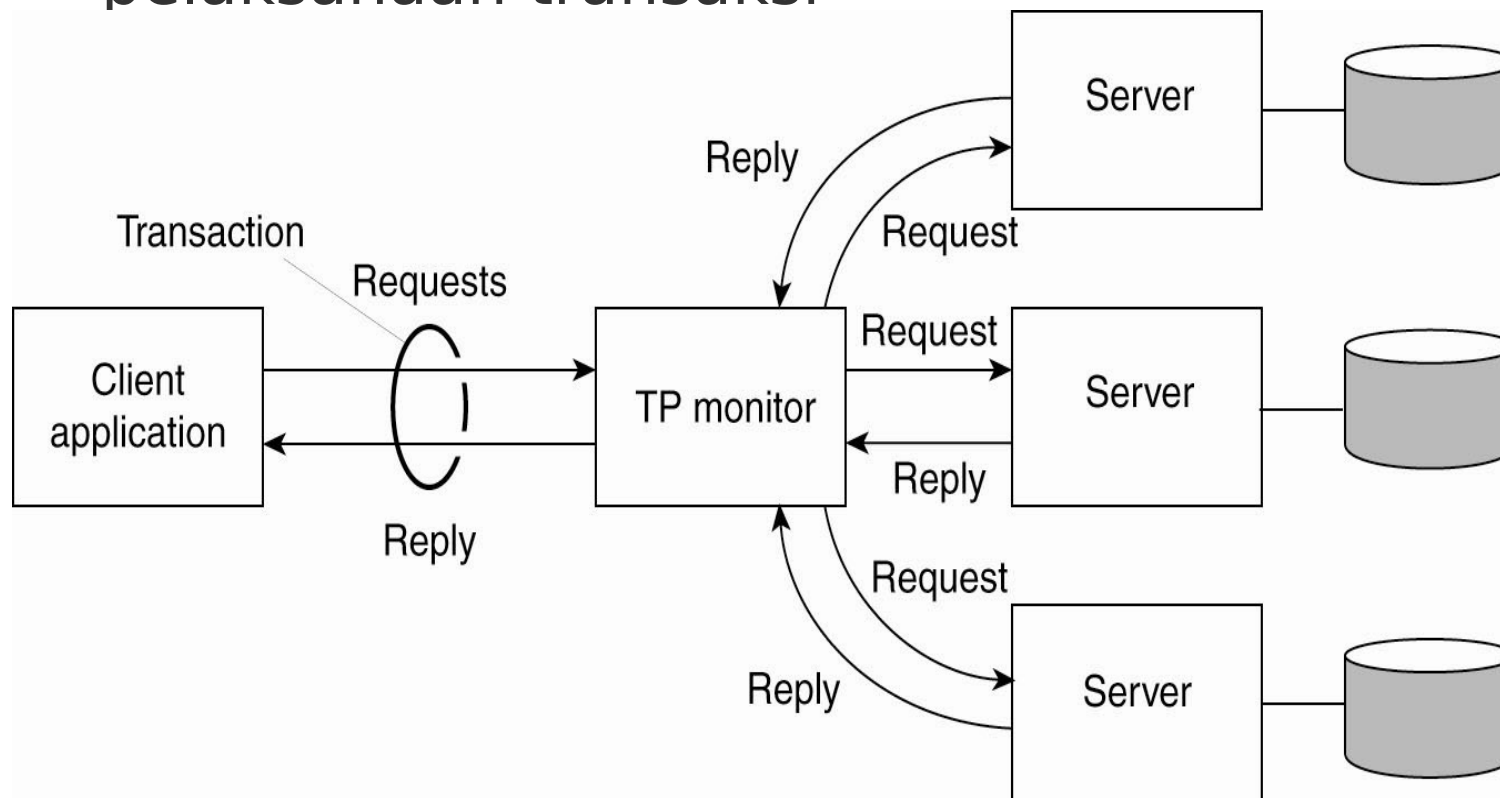
 WRITE(transaction, file-2, newData)

 END TRANSACTION(transaction)

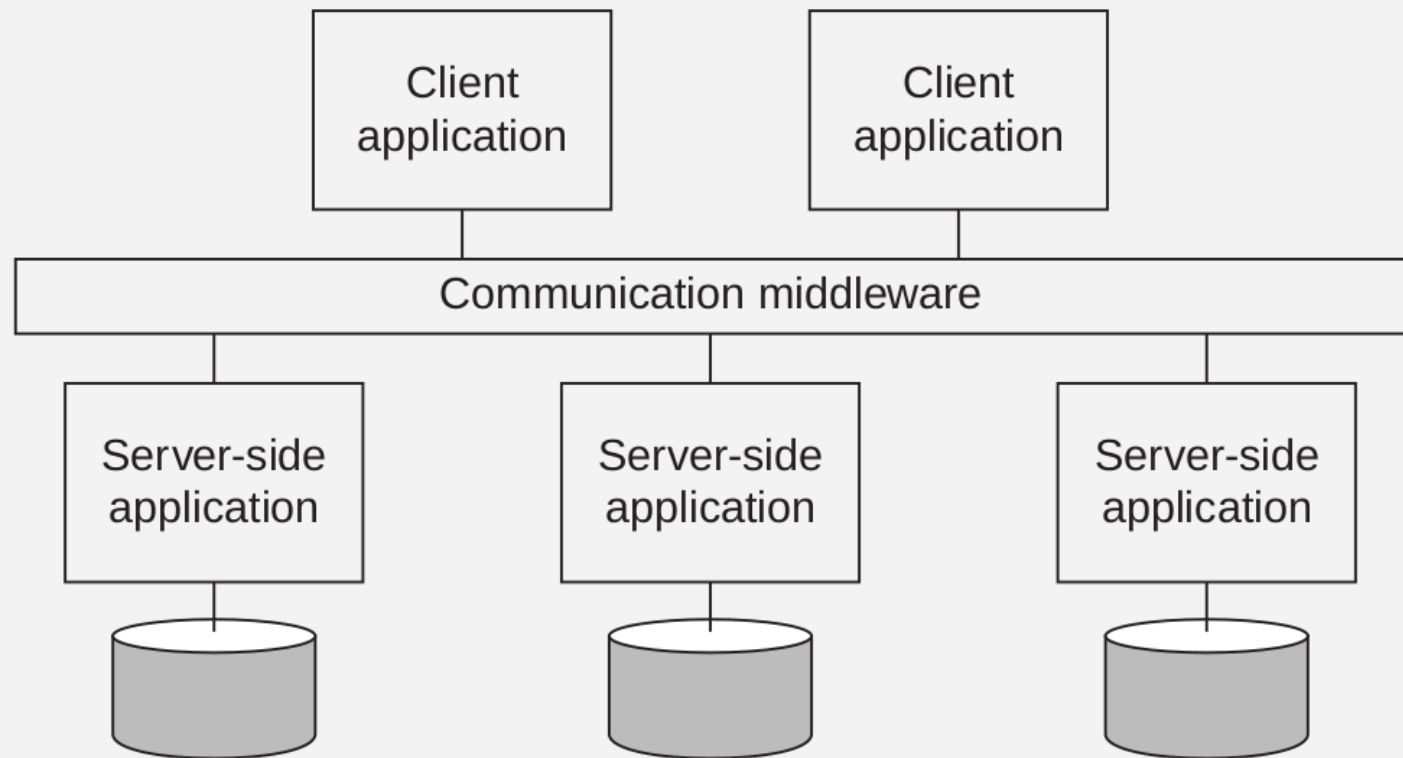
END IF

Distributed Information Systems: Transaction processing system

- Dalam banyak kasus, data yang terlibat dalam transaksi didistribusikan di beberapa server. TP monitor bertanggung jawab untuk mengoordinasikan pelaksanaan transaksi



Distributed Information Systems: Enterprise application integration



- Remote Procedure Call (RPC)
- Message-Oriented Middleware (MOM)



Communication Middleware Models/Paradigm

- Distributed File Systems
- Remote Procedure Call (RPC)
- Distributed Objects (RMI)
- Distributed Documents

Distributed pervasive systems

- Muncul generasi berikutnya dari sistem terdistribusi di mana **node kecil, mobile**, dan seringkali **tertanam** dalam sistem yang lebih besar, ditandai oleh fakta bahwa sistem secara alami menyatu dengan Lingkungan pengguna.
- Terdapat **tiga sub tipe**:
 - Ubiquitous computing systems
 - Mobile computing systems
 - Sensor (and actuator) networks

Distributed pervasive systems

- Sistem komputasi di mana-mana (***Ubiquitous computing systems***): pervasive (meresap) dan terus menerus hadir, mis., Ada interaksi berkelanjutan antara sistem dan pengguna.
- Sistem komputasi seluler (***Mobile computing systems***): pervasive, tetapi penekanannya adalah pada fakta bahwa perangkat pada dasarnya mobile.
- Jaringan sensor (dan aktuator): pervasive, dengan penekanan pada pengindraan aktual (kolaboratif) dan aktuasi lingkungan.

Ubiquitous computing systems

- Karakteristik dasar:
 - (**Distribution**) Perangkat terhubung jaringan, didistribusikan, dan dapat diakses secara transparan
 - (**Interaction**) Interaksi antara pengguna dan perangkat sangat tidak mengganggu
 - (**Context awareness**) Sistem menyadari konteks pengguna untuk mengoptimalkan interaksi
 - (**Autonomy**) Perangkat beroperasi secara mandiri tanpa campur tangan manusia, dan karenanya dikelola sendiri
 - (**Intelligence**) Sistem secara keseluruhan dapat menangani berbagai aksi dan interaksi dinamis

Mobile computing systems

- Mobile computing systems pada umumnya adalah subkelas dari Ubiquitous computing systems dan memenuhi semua lima persyaratan/karakteristik.
- Karakteristik khas:
 - Banyak jenis devices seluler: ponsel pintar, remote control, peralatan mobile, dan sebagainya
 - Komunikasi nirkabel
 - Perangkat dapat terus mengubah lokasi mereka → pengaturan rute mungkin bermasalah, karena rute dapat sering berubah
 - perangkat mungkin mudah terputus sementara → jaringan yang toleran terhadap gangguan

Sensor networks

- Karakteristik:
 - Node tempat sensor terpasang adalah:
 - Banyak (10-1000-an)
 - Sederhana (kapasitas memori / komputasi kecil / komunikasi)
 - Seringkali bertenaga baterai (atau tanpa baterai)

Sensor networks as distributed systems

