

Pengenalan Sistem Terdistribusi

2019/2020

Objectives

- Pemahaman konsep Sistem Tersebar
- Keuntungan dan kerugian Sistem Tersebar
- Mampu membedakan perbedaan Sistem Tersebar dengan Sistem Terpusat
- Memahami jenis-jenis Sistem Tersebar

Pengantar

- Era tahun 1945, komputer era modern mulai ada dimana **ciri** komputer dengan bentuk cukup besar secara **fisik** dan harga yang sangat **mahal**
- Awal tahun 1980, **dua** perkembangan teknologi merubah paradigma komputer yang ada pada saat itu
 - Mikroprocessor: arsitektur 8-bit, 32-bit, 64-bit
 - Jaringan Komputer Lokal: LAN, WAN
- Dengan teknologi ini tidak hanya mungkin, tetapi **mudah menggabungkan** beberapa komputer dalam jumlah yang besar yang **terhubung satu sama lain**. Hal tersebut kita sebut sebagai **sistem terdistribusi atau sistem tersebar**.

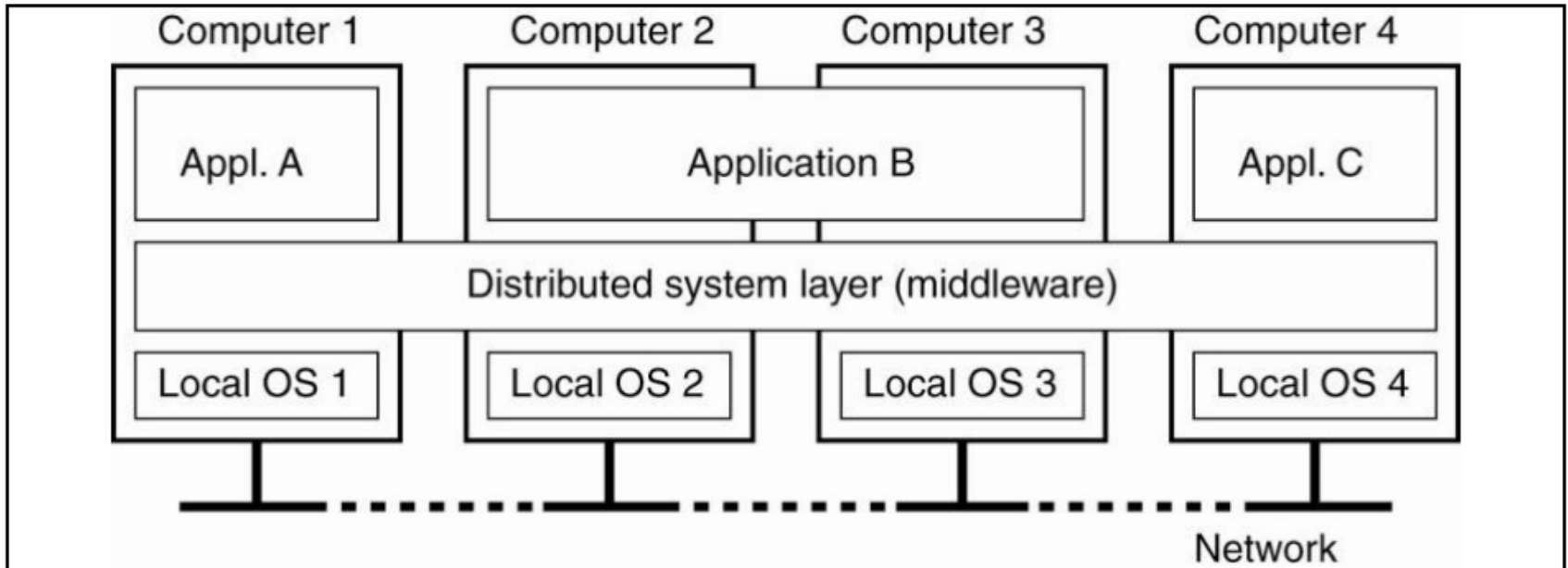
Sistem Terdistribusi

Definisi:

Sistem Terdistribusi adalah Kumpulan komputer **otonom** yang dihubungkan oleh **jaringan** dengan **software** yang dirancang untuk menghasilkan fasilitas **komputasi terintegrasi** sehingga **tampak** oleh user sebagai **satu sistem komputer**.



Middleware

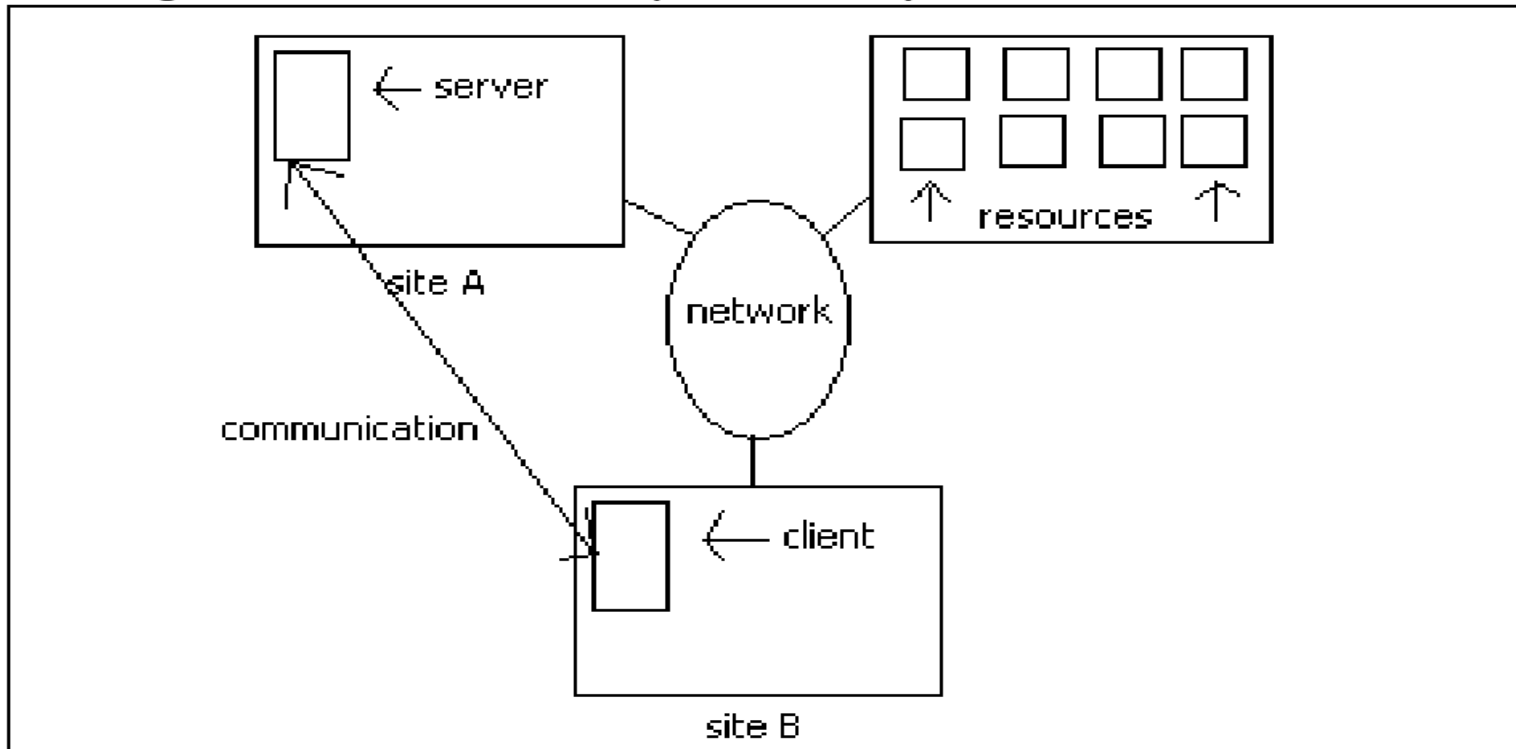


- Sumber: [Andrew S. Tanenbaum & Maarten van Steen, Distributed Systems: Principles and Paradigms, Prentice Hall](#)

Middleware

- Middleware merupakan **komponen perantara** yang memungkinkan client dan server (lapisan aplikasi dan sistem operasi) saling terhubung dan berkomunikasi satu sama lain. (1)
- Tujuan utama layanan middleware adalah untuk membantu memecahkan **interkoneksi** beberapa aplikasi dan masalah **interoperabilitas**. Middleware sangat dibutuhkan untuk bermigrasi dari aplikasi mainframe ke aplikasi client/server dan juga untuk menyediakan komunikasi antar platform yang berbeda.
- Middleware memiliki peran yang cukup penting karena dengan Middleware, maka client dapat **mengirimkan** pesan atau permintaan ke server, **menerjemahkan** pesan dari client agar dapat **dimengerti** oleh server dan demikian pula sebaliknya.
 - Contoh: Java's : Remote Procedure Call, Object Management Group's : Common Object Request Broker Architecture (CORBA), Microsoft's COM/DCOM (Component Object Model/ .NET Remoting), ActiveX controls (in-process COM components)

Struktur Sistem Tersebar



- Sumber: **Andrew S. Tanenbaum & Maarten van Steen, Distributed Systems: Principles and Paradigms, Prentice Hall**

Manfaat Sistem Terdistribusi

- Data sharing:
 - Mengizinkan pengguna untuk bisa mengakses data yang sama.
- Device sharing:
 - Mengizinkan pengguna untuk bisa mengakses perangkat keras yang sama.
- Communication:
 - Memungkinkan pengguna bisa melakukan komunikasi jauh lebih mudah
- Multiuser Computing :
 - Menerapkan banyak user yang dapat login pada saat yang bersamaan untuk mengakses sistem.

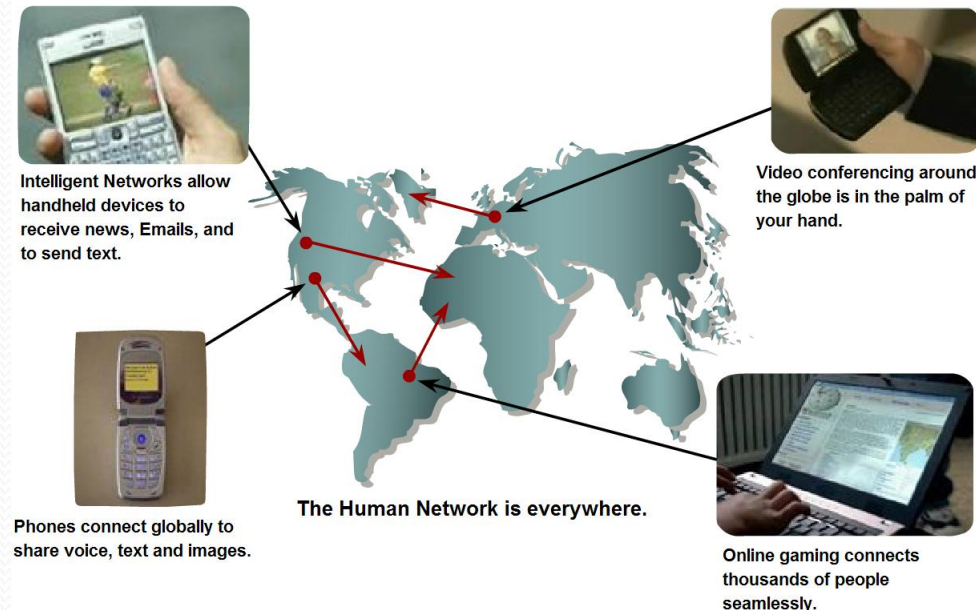
Manfaat Sistem Terdistribusi

- Flexibility:

- Membagi beban kerja pada perangkat yang tersedia dengan cara yang efektif.
- Dapat menambah komponen secara individu tanpa harus menduplikasi sistem
- Fasilitas local dapat disesuaikan dengan kebutuhan local
- Memungkinkan pertumbuhan sistem secara terus menerus.
- Susunan sistem bisa disesuaikan dengan pola organisasi perusahaan
- Memungkinkan beberapa bagian/local mengadakan percobaan konsep baru dan fasilitas baru untuk mengurangi resiko kegagalan sistem secara keseluruhan

Manfaat Sistem Terdistribusi

- Dari manfaat tersebut dapat diperoleh kesimpulan bahwa perbedaan sistem tersebar dengan sistem terpusat, sehingga perlu adanya sistem tersebar adalah:
 - Resource Sharing.
 - Computation Speedup
 - Reliability
 - Communication.





Pertanyaan

Sistem Terdistribusi

Vs

Jaringan Komputer

Kerugian Sistem Terdistribusi

- **Complexity** : Secara umum secara rancangan, implementasi dan maintenance (*error handling*) jadi **lebih kompleks** daripada arsitektur terpusat.
- **Security** : Asumsi terhadap sistem yang terhubung jaringan artinya telah masuk ke **jaringan publik** ini artinya setiap orang diperbolehkan untuk bisa mengakses data pada jaringan publik. Asumsi ini bisa **menjadi lubang keamanan** terhadap pengaksesan data secara tidak sah.
- **Manageability** : Membutuhkan **banyak usaha** untuk mengorganisasi sistem.
- **Unpredictability** : **Sulit diprediksi** terhadap ketidakstabilan sistem tergantung dari sistem organisasi dan banyaknya akses terhadap jaringan.

Karakteristik Sistem Terpusat

- Komputer otonom yang secara eksplisit terlihat (diberikan alamat ip)
- Berbagi resource **setiap saat** untuk seluruh user
- Aplikasi berjalan diatas **prosesor tunggal**
- Sumber daya terlihat/nampak
- Pengendalian tunggal
- *Single point of failure*

Karakteristik Sistem Terdistribusi

- **Gabungan** komputer bersifat otonom
- Sumber daya tidak dapat di akses secara langsung
- Aplikasi berjalan secara **bersamaan** pada **prosessor yang berbeda**
- *Concurrency of components.*
- *No global clock.*
- *Independent failures of components.*
- *Components are not shared by all users*
- *Multiple Points of control*

Tantangan

- Making resources accessible
- Distribution Transparency
- Openness
- Scalability
- Concurrency
- Fault Tolerance

Making Resources Accesible

- Hal utama dalam sistem tersebar adalah **mudahnya** pengguna **melakukan akses** terhadap **sumber daya**.
- Sumber daya dapat berbentuk *hardware* maupun *software*.
- Menghubungkan beberapa user dengan sumber daya memudahkan untuk **proses penggabungan dan pertukaran informasi**
 - Sebagai contoh printer, fasilitas penyimpanan, data, files, halaman web dan lain-lain.

Distribution Transparency

- Tujuan penting dari sistem terdistribusi adalah **menyembunyikan** bahwa fakta pemrosesan suatu *job* dilakukan secara **tersebar** melalui beberapa perangkat fisik yang berbeda (*Transparency*).
- *Transparency* membuat beberapa aspek distribusi **tidak nampak** oleh pengguna.

Aspek-aspek Transparency

- Beberapa aspek-aspek *Transparency* dalam sistem terdistribusi, meliputi:
 - *Transparency of access*; Pemetaan drive di samba, NFS, remotepriinter
 - *Transparency of location*; Transaksi perbankan ATM, Web Pages dll
 - *Transparency of relocation*; www, and Distributed database
 - *Transparency of replication*; Distributed DBMS, Mirroring Web Pages
 - *Transparency of concurrency*; NFS, Automatic teller machine network, Database management system
 - *Transparency of failure*; Database management system
 - *Transparency of Performance*;
 - *Transparency of Migration*; NFS, Web pages

Openness

- Memastikan sistem dapat diperluas dan mudah dalam pemeliharaan
- Perlu **integrasi berbagai komponen** yang dibuat oleh programmer atau vendor yang berbeda.
- Kunci aspek dari openness :
 - Antarmuka standar dan protocol (seperti protokol komunikasi internet)
 - Mendukung keberagaman (heterogeneity)

Scalability

- Bertambahnya kebutuhan terhadap sistem, memungkinkan untuk memperbesar dan memperkecil infrastruktur dari sistem tersebar itu sendiri
- Sistem tetap efektif meskipun terdapat peningkatan resource dan pengguna secara signifikan

Scalability

- Karakteristik algoritma tersebar :
 - Tidak ada mesin mempunyai **informasi lengkap** sekitar status sistem.
 - Mesin membuat **keputusan** berdasarkan pada **informasi lokal**.
 - **Kegagalan** dari satu mesin **tidak merusak** algoritma yang sedang berjalan.
 - **No global clock**, Terdapat batasan pada ketepatan proses sinkronisasi clock pada sistem terdistribusi, oleh karena asynchronous message passing dan pada sistem terdistribusi, tidak ada satu proses tunggal yang mengetahui global state sistem saat ini (disebabkan oleh concurrency dan message passing)

Scalability

- Scalability problems:
 - **Centralized services** : satu service untuk semua requests
 - **Centralized data** : satu data (data point) untuk semua services
 - **Centralized algorithms** : satu komputasi untuk semua requests
- Teknik menyelesaikan permasalahan skala
 - Hiding Communication Latencies (mengoptimalkan transmisi)
 - Distribution (contoh: DNS)
 - Replication (copy dan cache, masalah konsistensi)

Concurrency

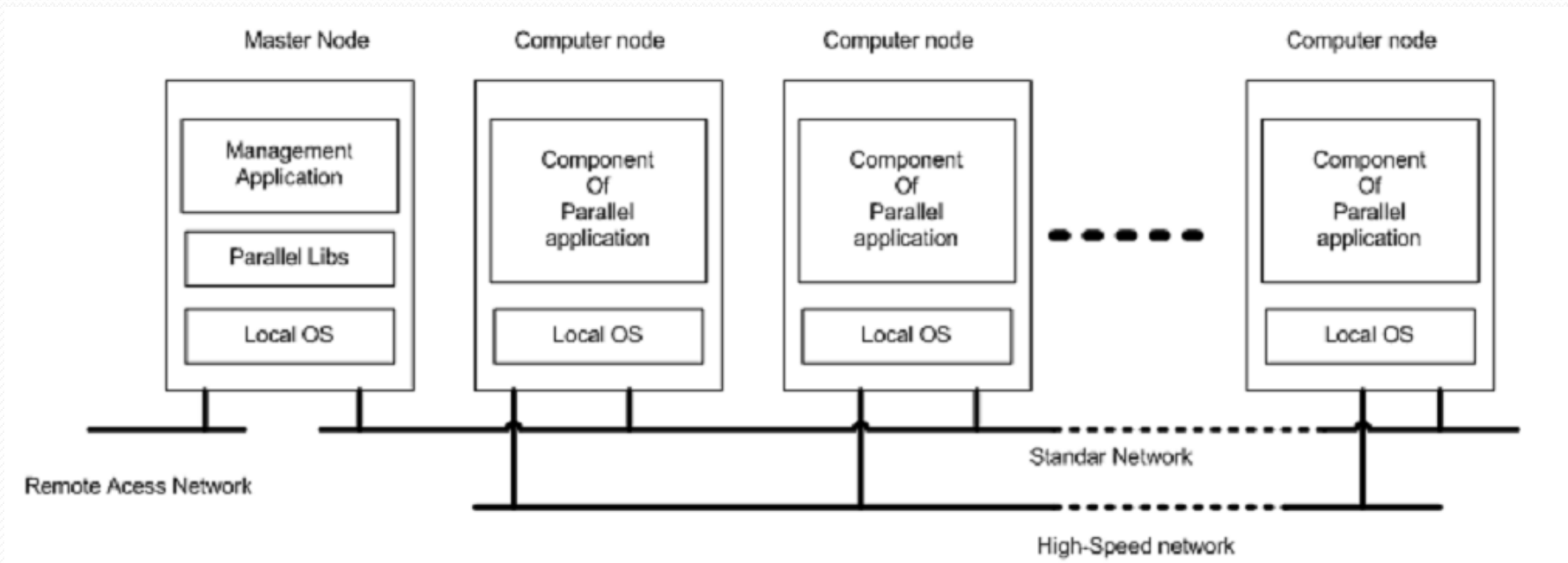
- Banyak client yang mengakses banyak data dalam waktu yang bersamaan, sedangkan data harus tetap konsisten
 - Proses dilakukan secara **simultan** dan membagi sumber daya sehingga bisa digunakan secara bersama
 - Menghindari masalah **deadlock**

Fault Tolerance

- Kemampuan untuk tetap melakukan komputasi ketika terjadi kesalahan
 - Deteksi/mask/toleransi kesalahan
 - Recovery terhadap kesalahan
 - Redundancy

Jenis Sistem Terdistribusi

- Distributed Computing Systems
 - Cluster Computing Systems
 - Grid Computing Systems
- Distributed Information Systems
 - Transaction Processing Systems
 - Enterprise Application Integration (Exchange info via RPC or RMI)
- Distributed Pervasive Systems (usually small, batterypowered systems, Mobile & wireless)
 - Home Systems (e.g. Smart phones, PDAs)
 - Electronic Health care systems (Heart monitors, BAN: Body Area Networks)
 - Sensor Networks (distributed Databases connected wirelessly)



- **Mesin-mesin terintegrasi** yang menjalankan proses secara bersama-sama dengan **membagi tugas** kedalam beberapa bagian yang di jalankan masing masing mesin.
- Jika prosesnya telah selesai maka hasil pengerjaan masing-masing mesin **digabung** menjadi satu (Parallel)

Grid Computing System

- Sebagai bandingan, arsitektur lain yang telah lebih dulu ada (**parallel computer** maupun **Cluster computing**) umumnya bekerja di atas lingkungan yang **homogenous dan terdedikasi**.
- Kondisi saat ini:
 - Komputer-komputer penyedia sumber daya bersifat **heterogenous** karena terdiri dari berbagai jenis **perangkat keras, sistem operasi** maupun **aplikasi** yang terpasang.
 - Komputer-komputer terhubung ke jaringan yang luas dengan **kapasitas bandwidth** yang **beragam**.
 - Komputer maupun jaringan **tidak terdedikasi**, bisa hidup / mati tidak sewaktu-waktu tanpa jadwal yang jelas.
- Untuk mengaktifkan Grid Computing, diperlukan perangkat lunak **Middleware khusus**



Terima Kasih