



Building a Network

(Lecture 2)

Requirements & Network Architecture

**Jaringan Komputer
dan
Komunikasi Data**
(Betty Purwandari MSc.)
Magister Teknologi Informasi
Fakultas Ilmu Komputer
Universitas Indonesia

Agenda

■ Dasar: Membangun sebuah Jaringan Komputer

■ Kebutuhan (Bab 1.1)

- Konektivitas (Bab 1.1.1)
- Pemakaian Sumber Daya Bersama (Bab 1.1.2)
- Dukungan untuk Pelayanan Aplikasi (Bab 1.3)
- **Kinerja (Performance) (Bab 1.4) ← (hal. 20)**

■ Arsitektur Jaringan (Bab 1.2)

- Lapisan dan Protokol
- Arsitektur OSI
- Arsitektur Internet

Review (..to remember)

■ Computer Networks:

- Generality (fungsi network yang sangat umum)

■ Bagaimana membangun jaringan komputer?

- Perspektif requirement: network manager, user, programmer

■ Requirement: Connectivity

- Dua atau lebih nodes terhubung **langsung** dengan links (fisik)
- Besarnya network dapat ditingkatkan dengan menggunakan: **switch**.
- Membangun jaringan di atas jaringan yang telah ada: Interkoneksi

■ Requirement: Resource Sharing

- Pemakaian bersama resource jaringan => cost effective
- Prasyarat rancangan & struktur jaringan.
Contoh **Multiplexing**: koneksi dapat digunakan secara bersama.

Requirement IV: Performance

■ Kinerja

- Selain fungsi terdapat kebutuhan jaringan untuk kinerja (**perform**) sesuai “**requirement**” aplikasi.
- Misalkan: transfer data dalam batasan waktu tertentu (on-line services), jumlah koneksi serentak (concurrent) dll.
- Membedakan kelas jaringan => kinerja tinggi (mis. jaringan kecepatan rendah vs kecepatan tinggi).
- Salah satu kebutuhan jaringan: kinerja (kecepatan) tinggi dengan “cost” yang rendah.

Bandwidth & Latency

■ Kinerja jaringan diukur dalam dua kategori:

- **Bandwidth (throughput):** jumlah bits yang dapat di-transfer dalam satu periode waktu
 - Misalkan: 1 Mbits/detik => 1 Mbps, berarti dapat mengirimkan data 1 juta bit setiap detik;
 - Bandwidth 1 Mbps, diperlukan waktu 1 mikro-detik untuk mengirimkan 1 bit.
- **Latency (delay):** berapa lama waktu yang diperlukan untuk mengirimkan “message” dari satu ujung (end) ke ujung lainnya.
 - Ukuran latency adalah satuan waktu.
 - Misalkan: latency untuk jaringan JKT – SBY: 20 milidetik (one-way).
 - Pengukuran lain Round-Trip Time (RTT): latency message bolak balik (two way).

Latency

■ Tiga komponen latency:

- **Propagasi**: waktu yang diperlukan untuk **perambatan** gelombang sinyal (misalkan batasan fisik kecepatan cahaya), ditentukan:
 - Kecepatan gelombang sinyal cahaya, misalkan pada fiber optik: 2.0×10^8 m/detik (konstan).
 - Jarak antara kedua titik ujung (pengirim dan penerima)
 - Waktu propagasi = jarak / kecepatan sinyal
- **Transmisi**: waktu yang diperlukan untuk mengirimkan semua bit data, misalkan message (tergantung **bandwidth** dan **besarnya data**).
 - Waktu transmisi = Besarnya Data / Bandwidth.
- **Antrian (Que)**: **delay** prosesing yang terjadi pada peralatan jaringan (misalkan switch atau multiplexer).

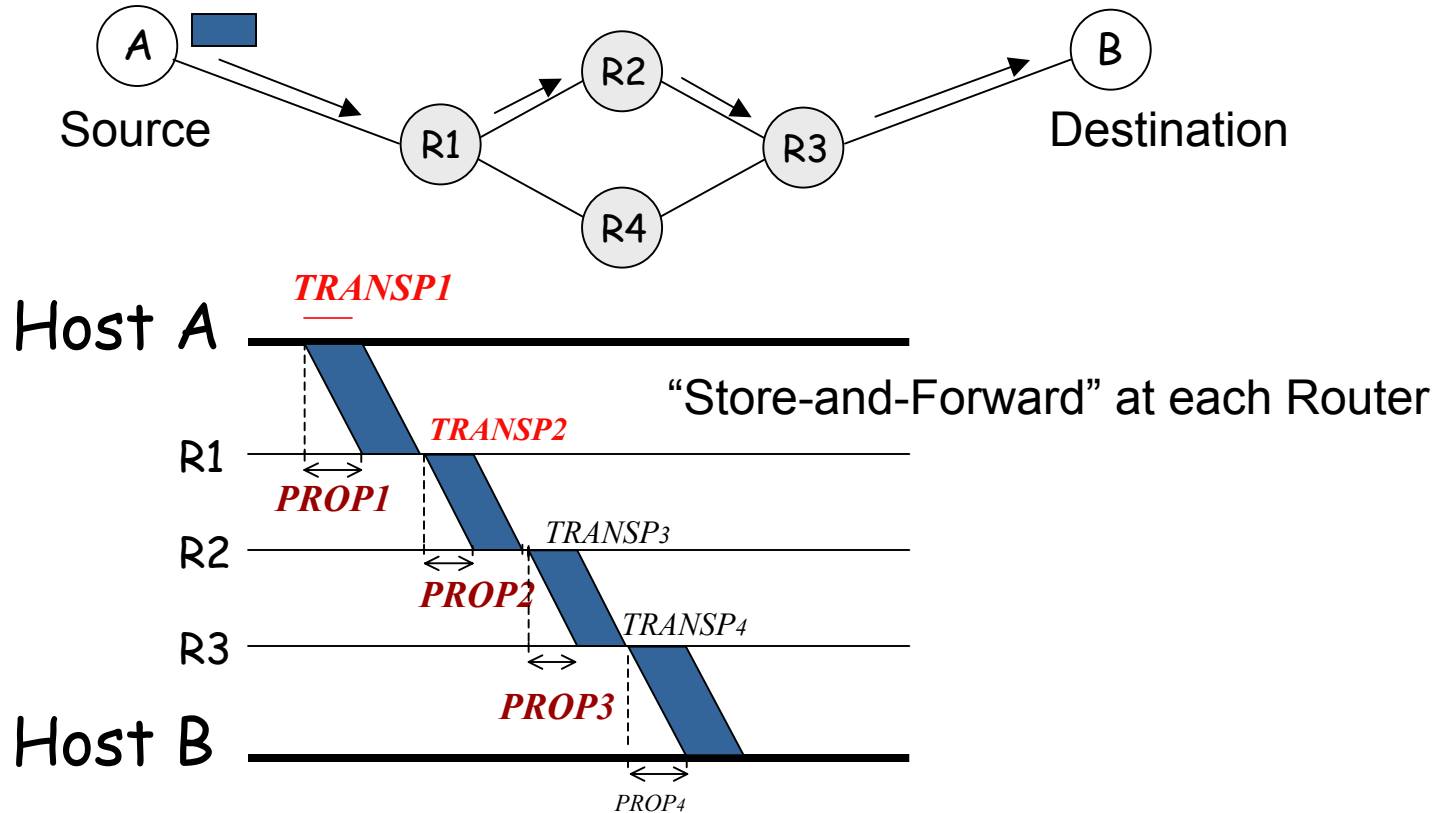
Latency



■ Misalkan

- P = panjang paket dalam bits
- L = panjang link dalam meter
- R = kecepatan bits dapat dikirim dalam bits/second
- c = konstanta kecepatan rambatan gelombang sinyal dalam meter/detik
- **Propagation delay: Waktu Propagasi = L/c**
- **Transmission time: Waktu Transmisi = P/R**

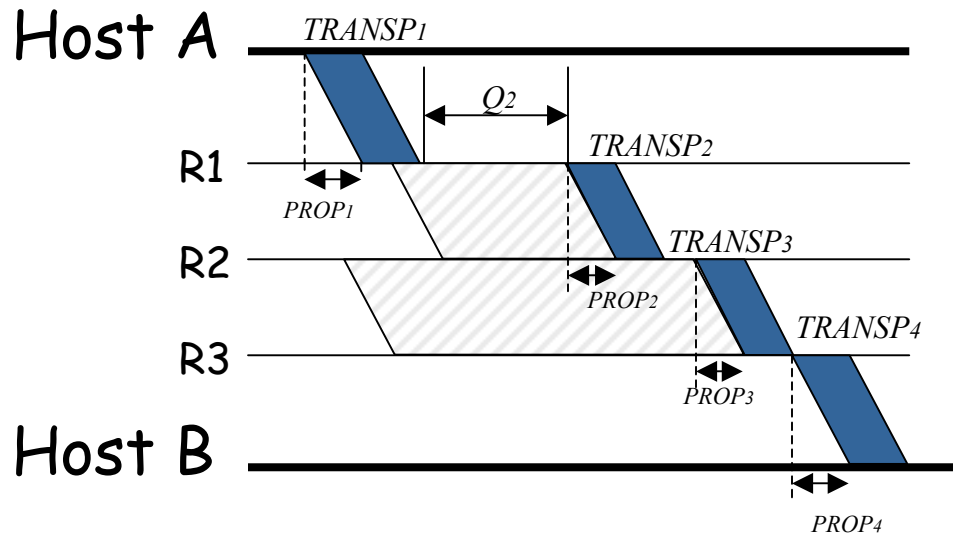
Example: Latency Network



$$\text{Minimum end to end latency} = \sum_i (TRANSP_i + PROP_i)$$

Example: Latency Network

Kemungkinan output link sedang digunakan, maka paket harus antri (queued) di dalam buffer => delay antrian



$$\text{Actual end to end latency} = \sum_i (TRANSP_i + PROP_i + Q_i)$$

Latency: Formula

$$\text{Latency} = (\text{Jarak/Kecepatan Sinyal}) + (\text{Besar Data/Bandwidth}) + \text{Antrian Delay}$$

Catatan (variabel penting):

- Propagasi: sesuai dengan media dan jarak, jadi jarak makin jauh waktu propagasi makin lama.
- Pengaruh besarnya bandwidth (teknologi transmisi data, cost link channel): makin besar bandwidth => waktu transmisi makin kecil.
- Pengaruh besarnya data: makin besar data makin lama waktu transmisi.
- Delay-Antrian: sesuai kondisi/load jaringan (traffic, node etc).

Summary (... latency)

- **Ukuran Kinerja jaringan ditentukan oleh faktor:**
 - Bandwidth dan Latency
- **Umumnya kita menggunakan Latency: berapa lama transmisi data/paket/message**
 - Waktu Transmisi Data => faktor dominan untuk “slower links or longer messages”
 - Propagation delay adalah faktor penting untuk link yang panjang
 - Antrian prosesing delay dapat mendominasi latency di dalam jaringan dengan load yang besar.

Agenda

- **Dasar: Membangun sebuah Jaringan Komputer**
 - **Kebutuhan (Bab 1.1)**
 - Konektivitas (Bab 1.1.1)
 - Pemakaian Sumber Daya Bersama (Bab 1.1.2)
 - Dukungan untuk Pelayanan Aplikasi (Bab 1.3)
 - Kinerja (Bab 1.4)
 - **Arsitektur Jaringan (Bab 1.2) ← (hal. 29)**
 - **Lapisan dan Protokol**
 - **Arsitektur OSI & Internet**

Network Arch.: requirements

■ Prasyarat networks:

- Interkoneksi: berbagai jenis/banyak komputer, komunikasi antar aplikasi secara cost-effective
- Perubahan kebutuhan (aplikasi), teknologi dan kapasitas

■ Diperlukan suatu arahan => rancangan jaringan (blue-prints)

- Arsitektur jaringan: Struktur dan model dalam membangun jaringan.
- Arsitektur => dasar rancangan dan implementasi jaringan.
 - Umum dapat memenuhi semua kebutuhan dan bentuk/ragam jaringan.
 - Akomodasi/abstraksi kerumitan (complexity) dan perubahan teknologi.

Layering (Abstraction)

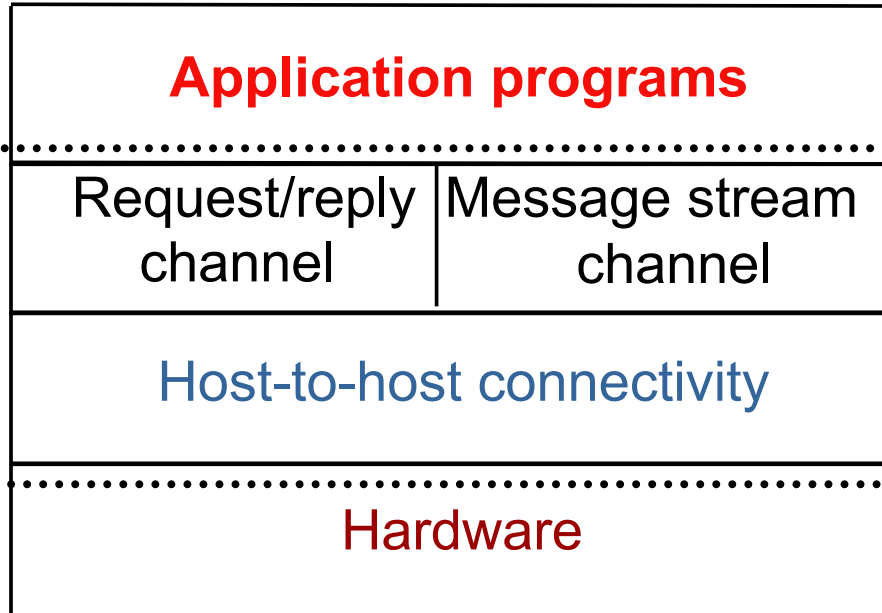
■ Secara sistem

- Divide and Conquer !!!
- Pendekatan pembagian moduler => services dan fungsionalitas
- Modul dasar: koneksi pada tingkat fisik (mis. kabel dan perangkat keras).
- Penambahan services di atas modul tersebut => pengiriman paket (resource sharing), koreksi kesalahan dst.

■ Lapisan (layering)

- Bentuk susunan modul mengarah pada lapisan (layer) => layer menggunakan services dari layer bawah

Layering (Abstraction)



Services kepada user

Abstraksi transfer data pada communication channel?

Abstraksi koneksi komputer dengan ragam bentuk jaringan

Services koneksi fisik (media)

- **Keuntungan struktur jaringan dalam bentuk lapisan (layer)**
 - **manageable components (function): dekomposisi**
 - **modular design: perubahan hanya pada layer**

Protocols

■ Implementasi struktur jaringan

- Sebagai “building blocks” => komponen/objek realisasi fungsi pada suatu lapisan, tentang tatacara berkomunikasi.
- Objek tersebut dikenal dengan sebutan **protokol**.

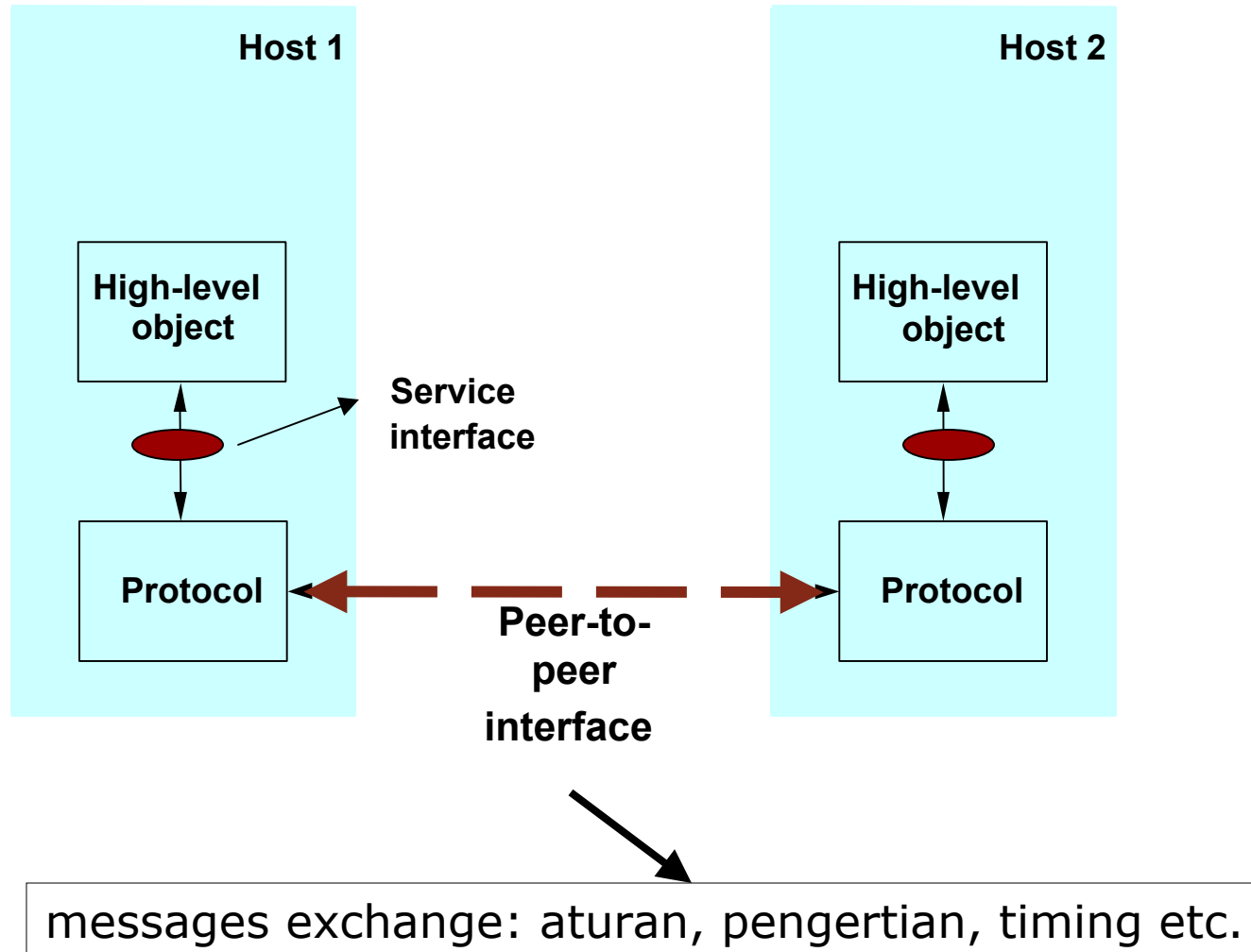
■ Dua sisi dari protokol:

- Service interface: operasi dan akses fungsi protokol pada entitas sama (s/w modul).
- Peer-to-peer interface: pertukaran messages untuk pasangan (peer, antar komputer).

■ Istilah “**protocol**”

- spesifikasi dari interface “peer-to-peer” (mis. standard Internet Protocol).
- modul implementasi service (mis. s/w & h/w yang “compliant” dgn IP)

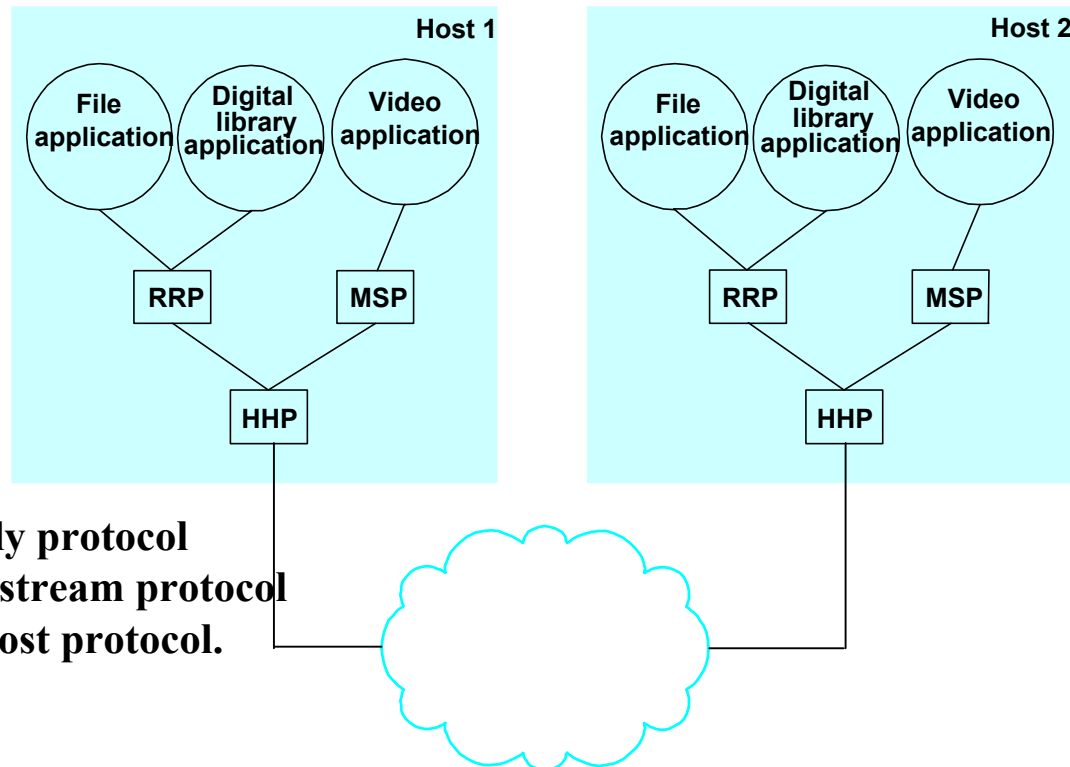
Interfaces



Protocol Machinery

■ Protocol Graph

- Ketergantungan antar protokol (services) pada protokol bawah.
- Gambaran kombinasi protokol untuk pertukaran data.



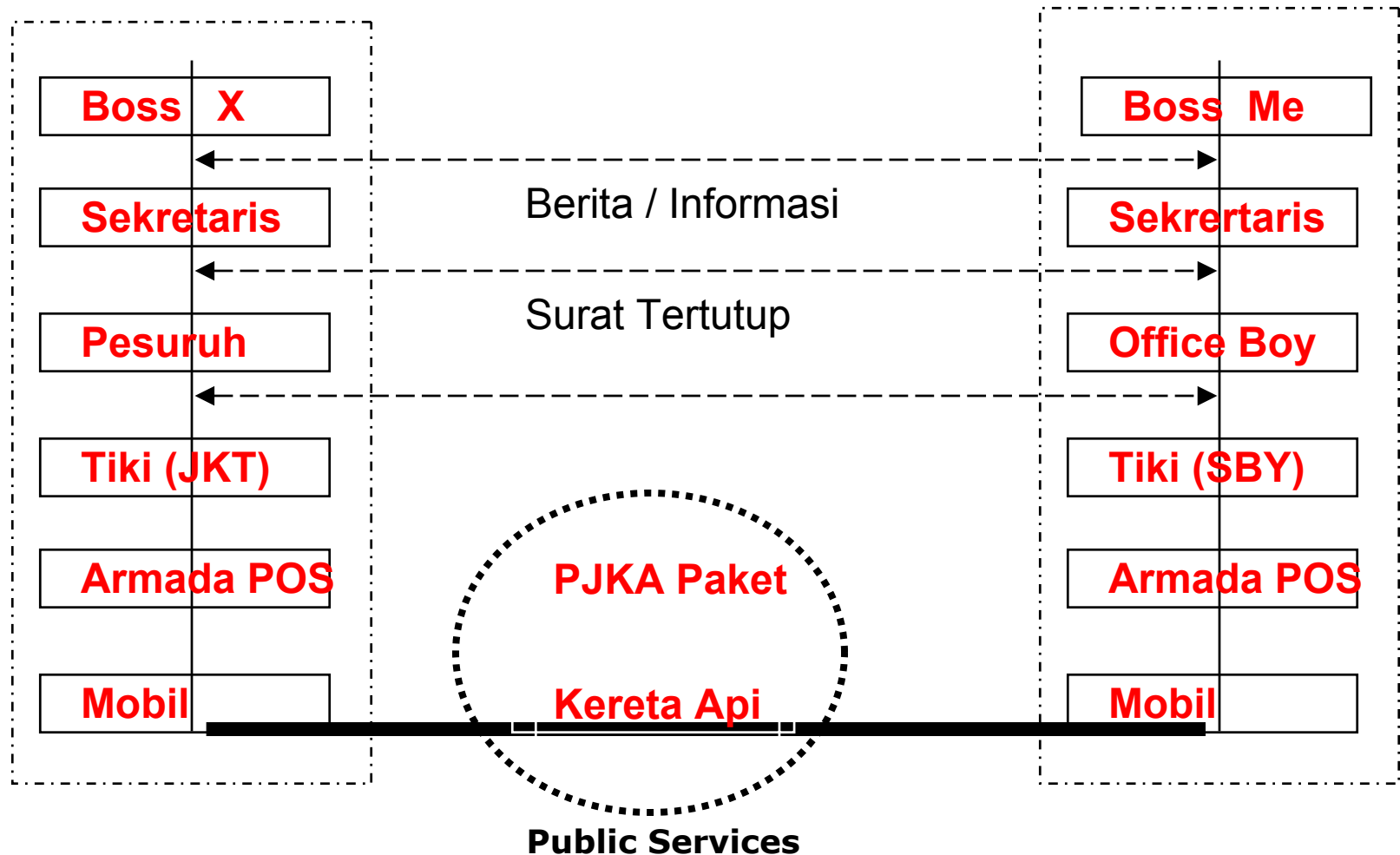
RRP = req./reply protocol
MSP= message stream protocol
HHP= host to host protocol.

Agenda

- **Dasar: Membangun sebuah Jaringan Komputer**
 - **Kebutuhan (Bab 1.1)**
 - Konektivitas (Bab 1.1.1)
 - Pemakaian Sumber Daya Bersama (Bab 1.1.2)
 - Dukungan untuk Pelayanan Aplikasi (Bab 1.3)
 - Kinerja (Bab 1.4)
 - **Arsitektur Jaringan (Bab 1.2) ← (hal. 29)**
 - Lapisan dan Protokol
 - **Arsitektur OSI & Internet**

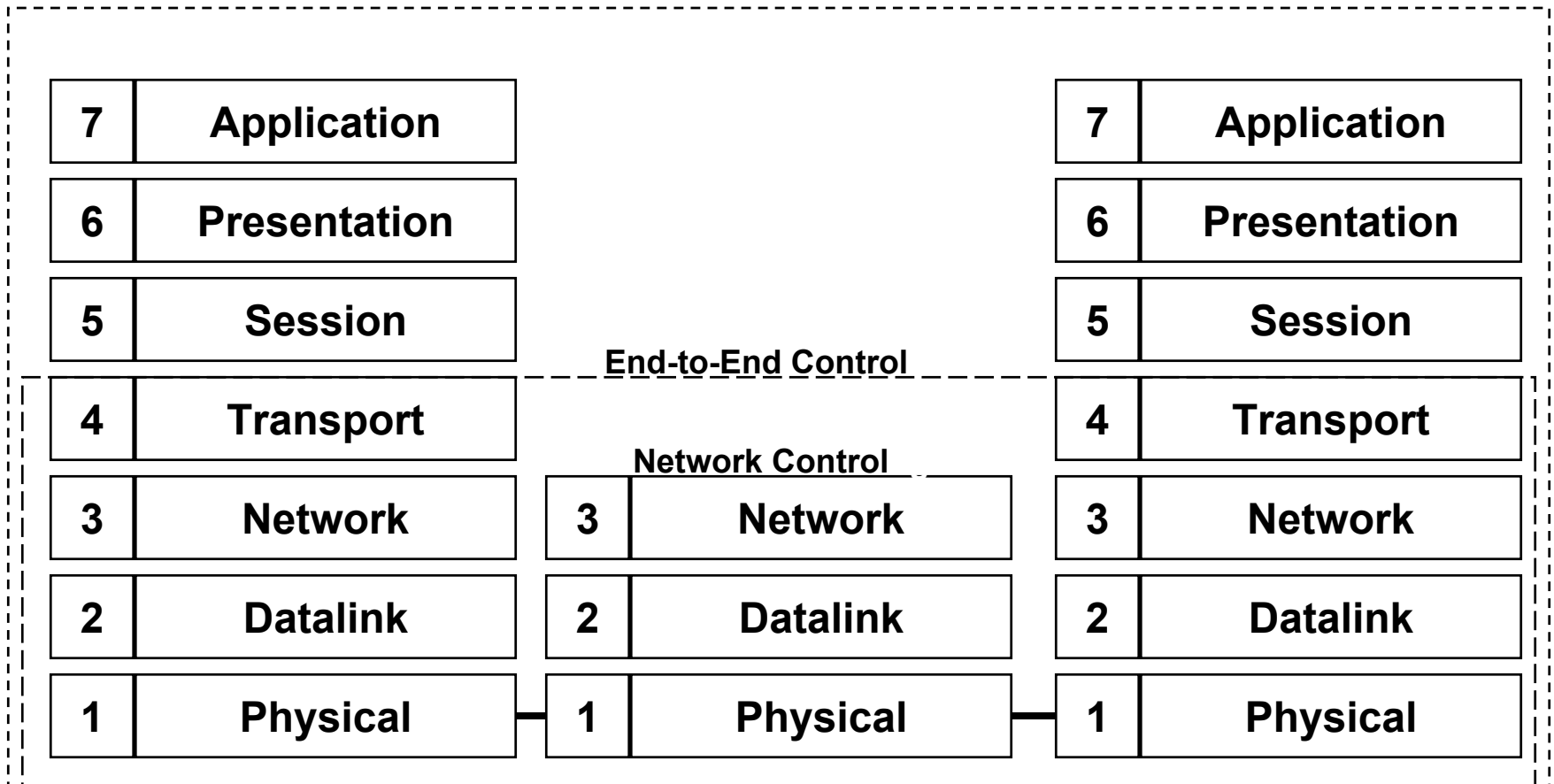
Analogy Layered (mail)

Review: Layered model untuk pertukaran messages mail



OSI Model

Acuan arsitektur OSI (Open System Interconnection)



Subnetwork (public data networks)

■ Mencakup 3 lapisan terbawah:

- Physical (Fisik): interkoneksi secara fisik
- Data link: transmisi data handal dan bebas kesalahan
- Network: relay dan routing untuk skala jaringan
 - Realisasi protokol, standard ITU (X dot series): X.25, X.400, X.500 dll.

■ Rasional: jaringan services publik (standard)

- Implementasi spesifik : menyediakan sarana umum untuk transportasi data (bisnis komunikasi data).
- Public service (public data networks) : digunakan oleh umum (berbagai pihak)
- Regulasi : diatur oleh pemerintah atau penyedia layanan (secara global)

End System

■ 4 Lapisan teratas fungsi end-to-end control

- Transport : reliable end-to-end data comm.
- Session : mengatur session antar host
- Presentation : mengatur cara data direpresentasikan
- Application : interaksi dengan user, program etc.

■ Rasional: Kendali akhir diserahkan ke komputer asal dan tujuan => host

- Host mengetahui aplikasi yang dijalankan
- Host relatif mempunyai kemampuan komputasi/storage yang lebih
- User lebih percaya pada host-nya sendiri

Physical Layer

- **Transmisi bit pada saluran fisik (i.e telekomunikasi)**
- **Fungsi**
 - Pelayanan ke Data Link : arus bit antara dua node
 - Melakukan konversi bit (data) ke sinyal yang sesuai dengan kanal fisik, misalnya menjadi tegangan pulsa, gelombang radio etc.
 - Representasi/definisi bit 0 dan 1 :
 - karakteristik elektrik (tegangan, frekuensi etc)
 - karakteristik waktu (timing), kecepatan transmisi bit

Data Link

- **Pengiriman paket data yang handal untuk dua node terhubung langsung (direct)**
- **Fungsi**
 - Service Lapisan Jaringan: reliable packet
 - Memperbaiki kesalahan bit pada lapisan fisik:
 - Membuat blok data: paket atau frame
 - Deteksi dan koreksi (retransmisi) frame yang rusak/salah.
 - Manajemen Link
 - Link set-up : awal nomor frame, test dll.
 - Akses kontrol pemakaian link (giliran).

Network

■ Pengiriman paket data pada subnet

■ Fungsi:

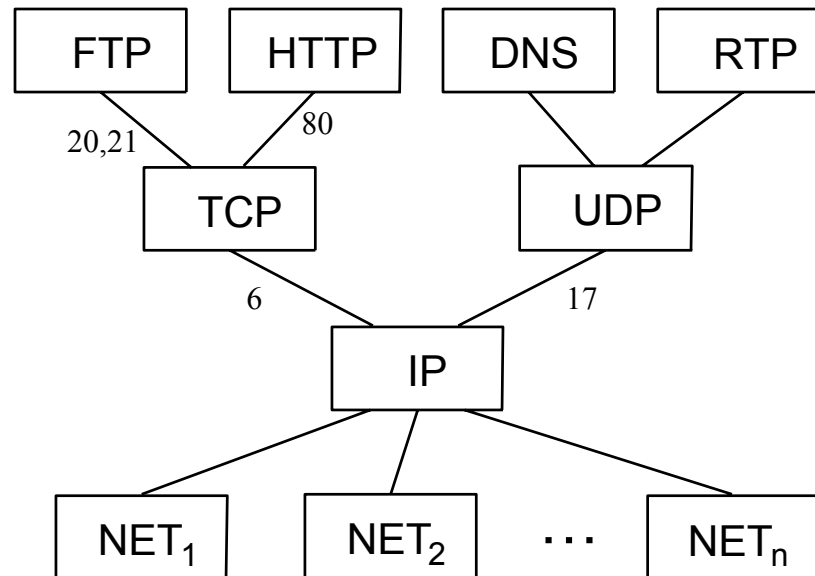
- Service Lapisan Transport: virtual network (teknologi & implementasi subnet)
- Penentuan rute paket: switch paket
 - Penentuan cara paket dikirimkan: connection vs connectionless
- Kontrol dan manajemen jaringan
 - Stabil : tidak macet (congestion), distribusi load traffic
 - Pengaturan arus paket (flow control) dalam jaringan.
- Internetworking: gabungan dari berbagai jaringan.

Transport

- **Transfer data dari end-to-end (host-to-host)**
- **Fungsi:**
 - Service lapisan sesi/aplikasi
 - Menjamin koneksi yang handal (mis. error control) untuk transmisi data ujung ke ujung.
 - Melakukan buffer dan kontrol kesalahan terhadap paket yang dikirimkan.
 - Mengatur arus pengiriman paket dari pengirim.

Example: Internet Protocols Stack

- Protokol dikembangkan: Internet Engineering Task Force (IETF) (<http://www.ietf.org>)
- Berpusat pada IP (IP over everything)



Example: Internet Protocols Stack

■ **Tingkat bawah**

- Variasi dukungan untuk berbagai protokol jaringan (bentuk dan jenis koneksi): Net1, Net 2.
- Gabungan h/w dan s/w (device drivers), mis. Ethernet, frame relay etc.

■ **Tingkat menengah**

- Protokol tunggal: Internet protocol (IP).

■ **Tingkat atas**

- Dua protokol utama: TCP (Transmission Control Protocol) dan UDP (User Datagram Protocol).

■ **Tingkat Aplikasi: protokol aplikasi.**

Summary

- **Struktur dan model jaringan => arsitektur jaringan**
 - Kita dapat membangun jaringan berdasarkan model jaringan secara berlapis => dekomposisi fungsi dan services.
- **Realisasi jaringan dalam bentuk sekumpulan protokol (stack) yang saling bekerja sama (protocol graph).**
- **Karakteristik dasar protokol: multiplexing dan encapsulation.**

Summary (... layering)

■ **Struktur bertingkat (layering)**

- Berguna sebagai model: spesifikasi jelas (fungsional dan interface)
- Tapi menambahkan overhead (mis. waktu dan proses)

■ **Korelasi antar OSI layering dan TCP/IP layering => fungsional**

■ **Implementasi “layering” => faktor kemudahan dan performance**

- Batasan antar layer sering tidak diikuti
- Fungsional dan efisiensi implementasi menjadi faktor penentu (realisasi software & hardware)

Additional Slides: Bandwidth

■ Sinyal digital

- Lingkup: transmisi sinyal digital (pulsa), sesuai dengan kecepatan clock => Hz, MHz (1 juta Hz).
- Umumnya 1 MHz clock mampu mengirikan sinyal atau bit sebesar 1 Mbps (juta bit per detik).

■ Bandwidth

- Jumlah bit yang dapat dikirimkan per satuan waktu, mis bps, Kbps, Mbps, Gbps dst.
- Notasi dan besaran mengikuti konvensi Hz, 1 Mbps = 10^6 bps (1 Kbps = 10^3 bps) (bandingkan dengan ukuran data, satuan bits, KB dan MB?).
- Makin besar bandwidth makin kecil pulsa digital => teknik sender/receiver makin rumit (cost lebih mahal).

Example: Exercise 1.5 (Page 61)

■ **Hitung waktu transfer 1000 KB file, asumsi RTT=100ms, ukuran paket 1KB data, dan diperlukan 2 RTT untuk handshaking awal.**

a). Bandwidth 1.5 Mbps, dan paket data dikirim secara kontinyu (tidak terputus)

- Gunakan rumus latency dan perhitungkan semua faktor yang memberikan kontribusi terjadinya delay dari sender ke receiver.
- Latency = [handshaking] + waktu propagasi [paket 1, one way] + waktu transmisi
- Latency = $[2 * RTT] + [RTT/2] + [BesarData/Bandwidth]$
- Latency = $[200ms] + [50 ms] + [1000KB/1.5Mbps]$
- Latency = $[200ms] + [50 ms] + [(1000*1024*8)/(1.5 * 10^6) s]$
- Latency = $0.25 s + 5.46 s = 5.71 \text{ second}$

Example: Exercise 1.5 (Page 61)

- **Hitung waktu transfer 1000 KB file, asumsi RTT=100ms, ukuran paket 1KB data, dan diperlukan 2 RTT untuk handshaking awal.**
- b). Bandwidth 1.5 Mbps, dan paket data tidak dikirim secara kontinyu, tapi setiap satu paket dikirimkan sender harus menunggu 1 RTT, kemudian mengirim paket berikutnya.**
 - Dengan cara ini terdapat overhead 1 RTT pada paket kedua, ketiga, dst sampai paket ke-1000; paket pertama tidak perlu menunggu sehingga total delay dari 1000 paket tsb adalah 999 RTT.
 - Latency = [handshaking] + waktu propagasi [paket 1, one way] + waktu transmisi + [total delay overhead menunggu]
 - Latency = 5.71 s + [999 * RTT]
 - Latency = 105.61 second.