

BAB

12

Jaringan Komunikasi Komputer

12.1 EVOLUSI JARINGAN KOMUNIKASI KOMPUTER

Jaringan komunikasi komputer diperoleh dari suatu kombinasi komputer dan telekomunikasi. Menurut sejarah, dua teknologi ini telah berevolusi secara terpisah, masing-masing dengan tujuan yang berbeda. Namun, pada tahun-tahun terakhir ini, cepatnya perkembangan teknologi digital telah memungkinkan penggabungan keduanya.

Penurunan harga dan peningkatan kemampuan hitung perangkat keras digital yang dihasilkan oleh teknologi LSI dan VLSI telah melihat telekomunikasi berkembang, mulai dari noncomputerized system switching voice-only, melalui suatu arsitektur uniprosesor yang terpusat, sampai ke sistem suara/data terinte-

grasi dan berbasis multiprosesor dewasa ini. Hal yang sama juga ada di belakang proses evolusi sistem komputer: mulai dari sistem dengan operasi komputer pusat dengan mode pemrosesan secara batch, melalui sistem *remote access*, interaktif, berbagi waktu (*time-sharing*), sampai ke sistem pemrosesan terdistribusi (*distributed processing system*) dewasa ini, yang melibatkan berbagai variasi sumberdaya komputer yang digabungkan bersama-sama untuk membentuk suatu sistem kerjasama di bawah skema pengendalian terdesentralisasi.

Teknologi VLSI dengan cepat mengubah segi ekonomis organisasi komputer. Komputer mini dan mikro dapat digunakan secara efektif untuk berbagai aplikasi. Komputer-komputer ini mempunyai kelompok instruksi yang baik, kemampuan pengalaman yang ekstensif dan perangkat lunak yang mampu memuaskan banyak pemakai komputer. Para pemakai akhir (*end-user*) juga memiliki perasaan yang sama—daripada membawa pekerjaan ke komputer, lebih baik komputer dibawa ke pemakai. Bagaimanapun juga, perpindahan kemampuan perhitungan dan sumberdaya ke pemakai akhir dan distribusi elemen pemrosesan tidak menyebabkan komputer besar menjadi usang. Lebih tepatnya mereka berfungsi untuk melukiskan area aplikasi dimana satu jenis sistem dapat digunakan secara lebih efektif daripada yang lainnya. Beberapa aplikasi, seperti pengolahan citra dan peramalan cuaca, memerlukan kemampuan pengolahan yang tidak dapat diberikan oleh komputer kecil. Demikian juga, beberapa aplikasi, seperti akuisisi data secara *realtime*, bisa saja tidak menggunakan sebuah komputer besar atau tidak memerlukannya dan dapat dikerjakan pada mesin yang lebih kecil.

Klasifikasi Jaringan Komunikasi Komputer

Jaringan komunikasi komputer dapat didefinisikan sebagai suatu kelompok sistem komputer independen yang saling berhubungan dan berkomunikasi satu dengan lainnya untuk tujuan pembagian sumberdaya perangkat keras dan perangkat lunak. Kita dapat mengklasifikasikan jaringan komputer atas dua kategori:

1. **Long-haul** atau **wide-area network**, menghubungkan sistem komputer yang berjauhan, secara geografis terpisah dan terpencar. Lebih lanjut, kelas ini terbagi atas tiga jenis:
 - (a) Jaringan komunikasi remote-access.
 - (b) Jaringan komunikasi pribadi (private).
 - (c) Jaringan komunikasi data umum.

2. **Local area network (LAN)**, menghubungkan sistem komputer yang berada dalam suatu lokal, secara geografis merupakan daerah yang terbatas.

Jaringan komunikasi pribadi dan jaringan remote-access terbatas bagi suatu komunitas pemakai yang tertutup, yang mempunyai kepentingan umum dan jaringan itu biasanya tidak hanya sekedar memberikan layanan komunikasi data. Sebagian besar dari mereka mempunyai fasilitas *packet-switching* dan mungkin juga fasilitas **pembagian sumberdaya** (*resource-sharing*), seperti ARPANET yang dikembangkan di Amerika Serikat, atau **bertujuan untuk suatu misi** (*mission-oriented*), seperti EURONET yang dibentuk oleh Masyarakat Ekonomi Eropa (EEC atau European Economic Community). ARPANET terentang ke seluruh penjuru Amerika Serikat dan berkembang ke beberapa bagian Eropa Barat, sedangkan EURONET melayani negara-negara anggota MEE. Meskipun jaringan remote-access mungkin saja tidak cocok untuk menggambarkan definisi suatu jaringan komunikasi komputer, tetapi jaringan tersebut merupakan jenis jaringan komunikasi data pribadi yang paling umum. Mereka menyediakan akses terminal secara remote yang meliputi daerah geografis yang luas ke satu atau lebih komputer induk (*host*). Terminal berkomunikasi dengan komputer induk untuk mendapatkan pelayanan batch dengan berbagi waktu (*time-sharing*) atau secara remote atau untuk mengakses database umum yang besar. Jaringan ini menggunakan konsentrator saluran (*line concentrator*) untuk mengurangi biaya sakuran komunikasi dan biasanya menyediakan fasilitas *dial-in-access* (akses melalui saluran telepon) sehingga pemakai dapat mengakses konsentrator lokalnya melalui saluran telepon. Contoh jaringan remote-access adalah INFONET dari Computer Sciences Corporation dan CYBERNET dari Control Data Corporation; keduanya terentang di Amerika Serikat, Eropa barat dan beberapa negara lain.

Jaringan data umum menyediakan pelayanan umum bagi berbagai macam pemakai dan dioperasikan oleh suatu perusahaan *common carrier*, meskipun peralatan-peralatan yang berhubungan tersebut merupakan milik banyak pelanggan yang berbeda. Mereka meliputi daerah geografis yang luas; namun pelayanan yang diberikan melalui sirkuit suara telpon mempunyai kelemahan pilihan tingkat data yang terbatas, biaya komunikasi yang tidak menguntungkan dan tingkat kesalahan yang terlalu tinggi untuk komunikasi data. Namun, tahun-tahun belakangan ini telah dikembangkan suatu jaringan umum yang baru yang dapat melayani lebih baik kebutuhan bagi komunikasi data. Semua pelayanan transmisi digital, seperti Dataphone Digital Service dari AT & T, menawarkan suatu pilihan tingkat data dengan harga yang pantas dan banyak pembaharuan tingkat kesalahan-

an. Jaringan umum lainnya, seperti Tymnet dan Telenet di Amerika Serikat, menyediakan pelayanan komunikasi data yang telah ditingkatkan dengan mengintegrasikan node packet-switched ke dalam sirkuit common carrier (leased) analog atau digital.

Local area network (LAN) tercakup dalam daerah geografis yang terbatas (kurang dari 24 km) dan berciri khas tingkat kecepatan data yang tinggi dan tingkat kesalahan yang rendah. Biasanya mereka menggunakan skema pengendalian terdesentralisasi dan sejumlah besar LAN menggunakan transmisi penyiaran. LAN menghubungkan berbagai macam piranti independen, seperti komputer, terminal, workstation, peripheral dan telepon melalui suatu media yang tidak mahal, sehingga menghasilkan biaya komunikasi yang jauh lebih rendah dibandingkan dengan jaringan long-haul. Ethernet mungkin merupakan jenis LAN yang paling terkenal. Ia dikembangkan oleh Xerox Corporation pada tahun 1970-an dan kemudian diperbaharui kembali secara kerja sama dengan Digital Equipment Corporation dan Intel, Ethernet menggunakan kabel koaksial dengan tingkat kehilangan yang rendah sebagai media transmisi di bawah pengendalian yang terdesentralisasi. Jarak maksimum antarstasiun 24 km dan jumlah maksimum stasiun dalam suatu jaringan adalah 1024 buah.

Hal-hal tentang Komunikasi Komputer

Jaringan komputer melibatkan sejumlah besar hal-hal yang kompleks yang tidak mungkin dibahas dalam satu bab. Karena itu, pembahasan kita akan dibatasi pada aspek pokok pembuatan jaringan komputer saja.

Suatu **arsitektur jaringan** merupakan suatu sistem logika yang rumit dan sangat terstruktur yang menggambarkan elemen-elemen fisik dan logika dari suatu sistem komunikasi dan menentukan hubungan serta interaksi yang mungkin di antara berbagai macam unsur jaringan. Untuk mengendalikan proses komunikasi, diperlukan beberapa tata cara untuk menentukan *kapan* dan *bagaimana* kita berkomunikasi dan *apa* yang dikomunikasikan. Tata cara ini, disebut sebagai **protokol komunikasi**, merupakan kumpulan aturan yang digunakan untuk mengendalikan pertukaran informasi antara unsur-unsur komunikasi suatu jaringan. Model arsitektur jaringan yang paling lazim akan dibahas pada Bagian 12.2.

Suatu jaringan komunikasi komputer tersusun atas sekumpulan node, disebut sebagai **node saklar** (*switching nodes*), yang saling dihubungkan oleh **saluran transmisi** (*transmission channels*). **Topologi** suatu jaringan adalah struktur yang

ditunjukkan oleh skema hubungan antar node (internodal). Berbagai macam topologi jaringan akan dibahas pada Bagian 12.3.

Pada Bagian 12.4, kita memperhatikan sejumlah hal yang berhubungan dengan teknik-teknik transmisi data. Data digital dapat ditransmisikan dalam tiga modus utama: **simpleks**, **setengah dupleks** dan **dupleks penuh**. Jika saluran transmisi merupakan **multipoint** maka bisa didapatkan beberapa kombinasinya. Sewaktu sebuah pengirim mentransmisikan data, si penerima harus mengsinkronisasikan dirinya pada data untuk pengambilan contoh yang tepat namun juga harus dapat menentukan awal dan akhir dari masing-masing blok data. Juga, karena fasilitas transmisi biasanya mahal dan sistem yang berkomunikasi tidak selalu menggunakan seluruh kapasitas saluran data, fasilitas tersebut harus mampu saling membagi kapasitas tersebut. Hal ini dilakukan dengan proses **multiplexing** dan kita membahas tiga skema yang biasanya digunakan dalam jaringan komunikasi komputer yaitu: **multiplexing dengan pembagian frekuensi (FDM atau frequency division multiplexing)**, **multiplexing dengan pembagian waktu (TDM atau time division multiplexing)** dan **multiplexing dengan pembagian waktu statistikal (STDM atau statistical time division multiplexing)**.

Media transmisi membawa-bawa data antara pengirim dan penerima. Ia dapat berupa penghantar secara fisik seperti kabel telepon twisted-pair, kabel koaksial dan kabel serat optik. Di lain pihak, media transmisi secara fisik tidak dapat dipisahkan seperti contohnya udara, air laut atau hampa udara. Karakteristik media-media ini akan dibahas secara terperinci pada Bagian 12.5.

Sebuah ciri khas utama suatu jaringan komunikasi komputer adalah arsitektur node saklar (*switching node*) dan jenis **teknik saklar** (*switching technique*) yang digunakan. Menurut teknik saklarnya, suatu jaringan komunikasi komputer dapat dibedakan atas dua jenis: **jaringan saklar** (*switched network*) dan **jaringan pemancar** (*broadcast network*). Teknik saklar yang digunakan oleh seorang pembuatnya dapat berupa **saklar sirkuit** (*circuit switching*), **saklar pesan** (*message switching*), **saklar paket** (*packet switching*) atau beberapa kombinasi atau variasi saklar-saklar tersebut. Jaringan pemancar menyertakan **radio**, **satelit** dan **jaringan lokal (LAN)**. Kita hanya membahas teknik-teknik jaringan saklar pada Bagian 12.6.

12.2 ARSITEKTUR JARINGAN KOMUNIKASI KOMPUTER

Suatu jaringan komunikasi komputer memungkinkan pertukaran informasi **antarelemen** (suatu piranti yang mampu mengirim atau menerima informasi) dalam

sistem yang berbeda-beda (sembarang kumpulan yang terdiri atas satu atau lebih elemen). Elemen-elemen yang dihubungkan dalam suatu sistem dapat berbeda-beda jenisnya. Proses komunikasi dikendalikan oleh suatu tata cara yang disebut sebagai protokol komunikasi. Suatu **protokol komunikasi** merupakan sekumpulan aturan, prosedur dan teknik yang biasanya diterapkan dalam perangkat lunak. Protokol tersebut menyatakan **format data**, disebut sebagai **sintaks** protokol, **format kendali**, yang disebut sebagai **semantiks** protokol, yang menyertakan informasi mengenai header, jejak (*trailer*), panjang data dan penanganan kesalahan (*error*); serta **informasi pengaturan waktu** bagai perangkaian dan pencocokan kecepatan. Transfer data diperoleh dari serangkaian pertukaran pesan tertentu antara elemen-elemen yang berkomunikasi. Dengan mengendalikan aliran pesan, jalur komunikasi logikal, yang mungkin (atau tidak mungkin) berhubungan ke hubungan fisik secara langsung, terbentuk di antara elemen yang berkomunikasi.

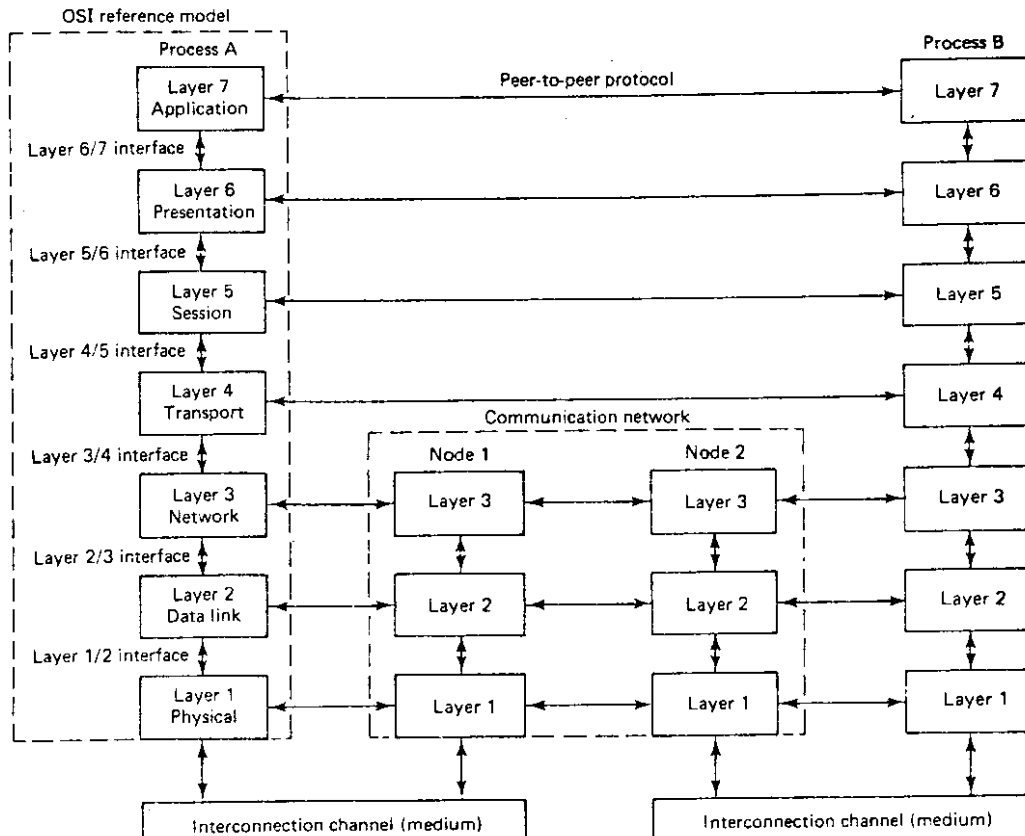
Arsitektur jaringan diorganisir dalam suatu bentuk **struktur bertingkat** (*layered structure*) yang memiliki suatu hirarki protokol jaringan dan memberikan suatu bentuk dekomposisi fungsional yang menyenangkan. Untuk menyelidiki beberapa konsep yang terlibat, sekarang mari kita perhatikan sebuah model arsitektur jaringan yang lazim digunakan, yaitu model pertalian hubungan sistem terbuka (*open system interconnection reference model*).

Model Pertalian Hubungan Sistem Terbuka

Model pertalian hubungan sistem terbuka (OSI atau open system interconnection) untuk arsitektur jaringan komunikasi komputer dibangun oleh International Standards Organization (ISO) dan diadaptasi oleh badan pembuat standar lainnya sebagai suatu langkah menuju terbentuknya standarisasi internasional bagi protokol jaringan. OSI memberikan kerangka kerja konseptual untuk mendefinisikan standar-standar bagi pertalian berbagai sistem komputer yang heterogen.

Hubungan sistem terbuka (OSI) berarti bahwa sistem itu dapat melakukan pertukaran informasi dengan dukungan kedua belah pihak dan pemakaian standar-standar yang dapat diaplikasikan; mereka saling *terbuka* satu sama lain untuk tujuan *interkoneksi*. Hal ini tidak mengandung arti adanya pemakaian bentuk implementasi sistem tertentu, teknologi maupun alat-alat interkoneksi. Hal ini terserah kepada vendor dimana peralatannya harus mengenali dan menunjang standar-standar yang dapat diterapkan.

Model pertalian OSI terdiri dari tujuh tingkatan (*layer*). Gambar 12-1 memperlihatkan secara skematis proses komunikasi baik langsung melalui saluran



Gambar 12-1 Model pertalian OSI dan proses komunikasi melalui sebuah saluran

interkoneksi (media transmisi) atau melalui satu atau lebih node jaringan perantara, seperti diperlihatkan pada Gambar 12-1. Node tersebut juga bertingkat namun, dalam contoh ini, masing-masing hanya terdiri dari tiga tingkatan. Akan dijelaskan nanti dalam bagian ini, kita tidak selalu perlu memakai ketujuh tingkatan tersebut.

Jumlah tingkatan, nama dan fungsi masing-masing tingkatan dapat bervariasi dari jaringan yang satu dengan yang lainnya. Namun demikian tujuan masing-masing tingkatan dalam jaringan manapun adalah untuk menyediakan beberapa

pelayanan bagi tingkatan di atasnya. Rincian sesungguhnya bagaimana pelayanan tersebut diterapkan tidak diketahui oleh tingkatan yang dilayaninya. Keuntungan pendekatan ini adalah bahwa masalah komunikasi elemen-ke-elemen yang nampaknya tidak dapat dikendalikan dapat kita dekomposisikan menjadi sejumlah subpermasalahan yang tetap dapat kita kendalikan.

Tingkatan-tingkatan yang saling berhubungan dalam sistem yang berbeda disebut sebagai **peer** dan berkomunikasi dengan menggunakan **protokol peer-ke-peer**. Penerapan fungsi dari sembarang tingkatan disebut sebagai suatu **entitas**. Suatu entitas dapat berupa sebuah subroutine atau sebuah proses di dalam sebuah sistem multiprosesing, sejumlah entitas yang sama atau berbeda dapat saling berhubungan melalui sebuah tingkatan. Jika komunikasi terjadi di antara entitas peer (misalnya dari komputer ke komputer) maka protokolnya disebut sebagai **simetris**. Jika komunikasi terjadi di antara entitas yang berbeda (misalnya dari komputer ke terminal) maka protokolnya disebut sebagai **asimetris**. Pengidentifikasian asimetris jelas berarti entitas yang lebih sederhana bagi elemen inferior secara logika. Sebagai contoh, perintah-perintah yang dikirimkan oleh sebuah terminal kepada sebuah komputer tidak sama dengan perintah-perintah yang dikirim oleh komputer ke terminal.

Perhatikan bahwa, dengan pengecualian bagi tingkatan fisik, tidak ada transfer data secara *langsung* antara tingkatan peer. Agar data dapat melewati tingkatan peernya maka setiap tingkatan menggunakan **komunikasi virtual** dimana ia mengirim data dan informasi kendali ke tingkatan di bawahnya. Proses ini berlangsung sampai tingkatan fisik tercapai, titik dimana mungkin (atau tidak mungkin) ada **komunikasi fisik** secara langsung antara kedua sistem komunikasi tersebut. Contoh pada Gambar 12-1 memperlihatkan bahwa proses A berhubungan dengan proses B melalui node 1 dan 2.

Suatu arsitektur jaringan biasanya memerinci protokol peer, menyatakan hubungan fungsional antara tingkatan-tingkatan tersebut dan menunjukkan informasi yang harus diteruskan antara tingkatan-tingkatan tersebut. Namun ia tidak menyatakan format **interface** antar tingkatan (lihat Gambar 12-1), yang berguna dalam memerinci operasi-operasi, disebut sebagai **primitif**, dimana setiap tingkatan disediakan bagi tingkatan di atasnya. Hal ini menjadi kebebasan pihak vendor peralatan tersebut untuk memutuskan, dengan memberikan suatu fleksibilitas terpasang (*built-in*), namun memerlukan sangkut paut yang ketat pada arsitektur jaringan. Satu-satunya pengecualian adalah batas tingkatan fisik pada peralatan, yang harus ada untuk melayani hubungan fisik. Sekarang mari kita perhatikan baik-baik fungsi dari masing-masing ketujuh tingkatan OSI.

Tingkatan 1: Tingkatan fisik (*physical layer*). Tingkatan fisik memberikan suatu alat untuk melekatkan media fisik dan mengendalikan pemakaiannya. Tingkatan ini berkepentingan dengan karakteristik elektris bagi pengaksesan ke media fisik, seperti tingkat voltase dan penetapan waktu bagi perubahan voltase (1/0 rasio tanda/ruang); karakteristik mekanis, seperti jenis-jenis konektor; dan karakteristik prosedural, seperti perangkatan kejadian (*event*) dan modus transmisi (contoh: setengah dupleks atau dupleks penuh; lihat Bagian 12.4). Contoh standar tingkatan 1 adalah RS-232-C, RS-449 dan IEEE 802 untuk LAN.

Tingkatan 2: Tingkatan hubungan data (*data link layer*). Aliran data melalui media fisik cenderung membuat kesalahan. Tingkatan hubungan data memberikan pendeteksian dan pengendalian kesalahan untuk meningkatkan kualitas pelayanan bagi tingkatan yang lebih tinggi dan agar kualitas hubungan dapat diandalkan. Tingkatan ini membagi-bagi aliran data mentah menjadi beberapa **kerangka** (*frame*), mengenali batas-batas kerangka, menggunakan jabat-tangan (*handshaking*) bagi komunikasi peer dan mentransmisi kembali kerangka-kerangka yang rusak/hancur. Ia juga bertanggung jawab atas pencocokan tingkatan data (kendali aliran) agar pengirim tidak membanjiri penerima dengan data. Contoh standar tingkatan 2 adalah High-Level Data Link Control (HDLC), prosedur Advanced Data Communication Control (ADCC) dan IEEE 802 untuk LAN.

Tingkatan 3: Tingkatan jaringan (*network layer*). Tingkatan ini melaksanakan fungsi-fungsi jaringan, seperti pensaklaran (saklar sirkuit, pesan atau paket; lihat Bagian 12.6) dan informasi routing antarpiranti akhir. Tingkatan ini menciptakan, mengelola dan memutuskan hubungan-hubungan dalam suatu jaringan dan dapat pula menjalankan fungsi sebuah node perantara (seperti diperlihatkan pada Gambar 12-1) atau fungsi menghubungkan dua jaringan yang terpisah. Suatu contoh standar tingkatan 2 adalah X.25.

Tingkatan 4: Tingkatan transportasi (*transport layer*). Karena tingkatan jaringan memelihara transmisi data dan teknik-teknik saklar, transfer data antara entitas transportasi menjadi transparan. Tingkatan transportasi dengan demikian dapat memberikan suatu mekanisme transmisi dengan kualitas pelayanan yang disyaratkan bagi tingkatan di atasnya. Ia memastikan tingkat kesalahan yang dapat diterima dan pengiriman data sekuensial tanpa kehilangan dan duplikasi dan menentukan prioritas dan penundaan. Suatu contoh standar tingkatan 4 adalah protokol transportasi (TP atau Transport Protocol).

Tingkatan 5: Tingkatan sesi (*session layer*). Sewaktu dua aplikasi akan bertukar data, ada beberapa persetujuan yang harus dicapai mengenai bagaimana bentuk pertukaran akan terjadi. Tingkatan 5 bertanggung jawab mengatur tugas liaison ini, yang disebut sebagai sebuah sesi, dan untuk mengatur, mengkoordinasi dan mensinkronisasi dialog tersebut. Ia juga memberikan pelayanan penyembuhan jika terjadi beberapa kesalahan. Penetapan standar bagi protokol sesi sampai saat ini belum tercapai.

Tingkatan 6: Tingkatan presentasi (*presentation layer*). Tingkatan presentasi memastikan agar pertukaran data antar piranti bagi entitas aplikasi dalam bentuk yang mereka ketahui. Ia memungkinkan memungkinkan komunikasi antar berbagai jenis peralatan dengan format data yang berbeda-beda pula. Ia memanipulasi dan mengkonversi data terstruktur untuk memecahkan perbedaan dalam format dan representasi data dan memberikan pelayanan-pelayanan seperti konversi kode (karakter), konversi format file, pemadatan pesan dan pengenkripsian data. Penetapan standar bagi protokol tingkatan 6 masih menjadi isu terbuka.

Tingkatan 7: Tingkatan aplikasi (*application layer*). Inilah tingkatan tertinggi yang dinyatakan dalam model tersebut. Ia berkonsentrasi pada dukungan aplikasi pemakai secara langsung dan terserah pemakai untuk mengisinya. Aplikasi meliputi pertukaran informasi antara pemakai akhir, program aplikasi atau piranti dan alokasi tugas. Pada dasar dibutuhkan dua jenis protokol oleh tingkatan 7. Yang pertama mendukung pelayanan bagi pemakai dan khusus untuk aplikasi tertentu. Jenis kedua lebih umum dan untuk tujuan pelayanan jaringan seperti fungsi-fungsi manajemen.

Ada dua hal yang diperhatikan dalam model pertalian OSI. Yang pertama bahwa model itu sendiri tidak menjamin adanya interkoneksi. Tiap-tiap tingkatan menyatakan fungsi dan pelayanan yang akan diterapkan dalam protokol. Beberapa protokol tidak tergantung pada jaringan, sementara yang lainnya tergantung pada jaringan dengan ekstensifikasi yang lebih besar atau kecil. Contoh protokol jenis pertama berhubungan dengan tingkatan 5, 6 dan 7. Sebuah contoh protokol jenis berikutnya adalah tingkatan fisik, yang akan diterapkan dengan lain bila medianya kabel koaksial atau jika jaringan merupakan packet-switched. Hal kedua adalah bahwa media fisik itu sendiri bukan merupakan bagian dari model tersebut. Pada jaringan jarak jauh, misalnya, ia terdiri dari jaringan publik, saluran yang disewa secara langsung dan modem atau peralatan pengendali lainnya yang diperlukan untuk memungkinkan pemakai akhir (komputer, pengendali, terminal,

dsb.) untuk menghubungkan mereka. Dalam LAN, media fisik terdiri atas kabel, pe-ngulang (repeater) dan penerima (transceiver) yang memungkinkan peralatan pemakai berhadapan muka dengan jaringan tersebut. (Suatu **pengulang** adalah sebuah piranti yang menerima data pada sebuah jalur komunikasi dan mentransmisikannya, bit per bit, pada jalur lainnya secepat ia menerimanya. Suatu **penerima** adalah sebuah piranti yang mengkombinasikan pengirim dan penerima).

Untuk menggambarkan prinsip-prinsip OSI dalam operasi, anggaplah bahwa aplikasi X dalam proses A pada Gambar 12-1 mempunyai sebuah pesan yang akan dikirim ke aplikasi Y dalam proses B. X mentransfer data ke sebuah entitas aplikasi pada tingkatan aplikasi A. **Header A** yang berisi informasi yang diperlukan bagi protokol tingkatan 7 peer ditambahkan pada data. Data aslinya dengan header tersebut sekarang sebagai sebuah unit lewat ke tingkatan 6 A. Entitas presentasi memperlakukan keseluruhan unit sebagai data dan menambahkan header-nya sendiri. Proses ini terus turun sampai tingkatan 2, yang biasanya menambahkan sebuah header dan **trailer** pada pesan tersebut. Unit tingkatan 2, disebut sebagai sebuah **kerangka**, kemudian melalui tingkatan fisik ke media komunikasi.

Sewaktu sebuah kerangka diterima oleh B, terjadi proses sebaliknya. Begitu data masuk, tiap-tiap tingkatan mengupas header paling luar, bekerja sesuai informasi protokol yang terkandung di dalamnya dan meneruskan sisanya ke tingkatan berikutnya. Perhatikan bahwa pada setiap tahapan, sebuah tingkatan mungkin saja memecah data menjadi beberapa bagian agar sesuai dengan persyaratannya sendiri. Unit data yang terbagi-bagi ini harus dibangun ulang lagi dengan menghubungkan tingkatan peer sebelum diteruskan.

Fungsi Protokol

Fungsi sebuah protokol tertentu tergantung pada sistem komunikasi dan karakteristik proses komunikasi yang diterapkan pada semua tingkatan. Mari kita teliti beberapa fungsi-fungsi dasar yang terlibat dalam perancangan protokol.

Segmentasi. Sistem komunikasi pada tingkat aplikasi mengatur transfer aliran data dari suatu entitas ke entitas lainnya. Untuk alasan seperti kendali kesalahan dan akses yang adil untuk pembagian fasilitas transmisi, tingkatan yang lebih rendah mengharuskan aliran data disegmentasikan (dipecah) menjadi beberapa blok berukuran maksimal. Fungsi protokol ini dapat dirujuk sebagai **segmentasi**. Karena data yang telah disegmentasikan harus dibangun ulang menjadi

aliran data yang tepat bagi tingkatan aplikasi, protokol juga harus dapat membalik proses tersebut.

Pembuatan kerangka (*framing*). Arus data melalui jalur fisik dilakukan dengan membagi-bagi aliran data dan mentransmisikannya sebagai serangkaian kerangka. Sebagai tambahan bagi data yang dilekatkan pada kerangka (*field data*), mekanisme pembuatan kerangka protokol memberikan alamat sumber dan/atau tujuan, kode pengkoreksian kesalahan dan header serta trailer untuk menandai masing-masing awal dan akhir kerangka. Fungsi ini juga disebut sebagai **enkapsulasi**.

Jika sebuah kerangka mengandung data arbitrer, protokol harus memastikan bahwa beberapa pola bit pada *field data* tidak salah diambil bagi pengendalian pembuatan kerangka, jika tidak maka kerangka dapat berhenti secara tiba-tiba. Beberapa teknik telah diciptakan untuk mengatasi permasalahan ini, sebagai contoh, dengan memberi indikasi besarnya kerangka pada header dan pengisian bit (*bit stuffing*) (penyelipan bit tambahan ke dalam suatu aliran data untuk menghindari pemunculan rangkaian kendali yang tidak diinginkan).

Pengendalian kesalahan (*error control*). Saluran komunikasi yang berisik dapat menghilangkan bit-bit yang ditransmisikan, namun data dapat juga menyebabkan kesalahan atau hilang sewaktu ditransfer ke atau dari memori, menyebabkan sebuah komputer atau interface node jaringan, atau melalui node yang gagal. Semua keadaan ini akan menghasilkan kehilangan, kerusakan, duplikasi data dan informasi kendali dan dengan demikian protokol harus memberikan mekanisme pengkoreksian kesalahan. Hal ini biasanya dilakukan dengan menambahkan informasi kendali redundan pada data yang sedang ditransmisikan. Contoh mekanisme pengkoreksian kesalahan adalah skema ARQ (*automatic repeat request*) dan penggunaan kode siklis.

Pengendalian arus (*flow control*). Fungsi protokol ini dibutuhkan untuk menghadapmukakan tingkat data sumber informasi dengan tingkat data tujuan informasi sedemikian sehingga sumber informasi tidak membanjiri tempat tujuannya. Pengendalian arus juga diperlukan untuk berhubungan dengan arus logika bagi sumberdaya jaringan yang terbatas, dalam kasus dimana hal ini disebut sebagai **pengendalian penuh-sesak (*congestion control*)**.

Perangkaian (*sequencing*). Dalam beberapa kasus, protokol harus memungkinkan adanya transfer unit data ke tujuan dengan *urutan yang sama* sepa-

gaimana mereka dihasilkan oleh sumbernya. Mekanisme perangkaian biasanya didasarkan pada beberapa indikator nomor urut yang dilekatkan pada masing-masing pesan yang ditransmisikan.

Pemberian prioritas (*prioritization*). Data dan (terutama) pesan kendali mungkin perlu diatur prioritasnya sehingga mereka dapat mencapai entitas penerima dengan penundaan yang minimum. Pemberian prioritas mengurangi penundaan queuing dan penting bagi aplikasi interaktif dan untuk memastikan bahwa fungsi-fungsi jaringan yang kritis, seperti pengarahan pesan, tidak tertunda secara tidak pada tempatnya.

Pengalamatan (*addressing*). Entitas yang berkomunikasi harus dapat mengidentifikasi satu sama lain pada setiap tingkatan protokol. Alamat sumber memungkinkan penerima data mengidentifikasi aslinya dan alamat tujuan dibutuhkan untuk pengarahan (*routing*)/pengiriman (*delivery*) data yang tepat. Isu pengalamatan pada kenyataannya lebih kompleks daripada pengamatan pertama kita, seperti diilustrasikan oleh skenario berikut ini: (1) Mungkin terdapat lebih dari satu jaringan yang terlibat sehingga masing-masing harus memiliki identifier sendiri; (2) suatu sistem (misalnya komputer) dapat dilekatkan pada sebuah node melalui media transmisi yang terbagi (*shared*), dimana dalam kasus ini interface dan node harus memiliki identifier yang berbeda; (3) entitas komunikasi (misalnya program atau proses) mungkin berkedudukan pada sistem yang berbeda dari vendor yang berbeda pula, dimana masing-masing sistem memiliki konvensi identifikasinya sendiri; dan (4) proses dihubungkan dengan port I/O logikal yang harus dialamat.

Karena jelas tidak mungkin untuk memaksa aturan-aturan pengalamatan yang seragam, pendekatan umum bagi permasalahan ini adalah dengan menggunakan ruang alamat hierarkis. Dengan demikian pengalamatan mencakup sebuah **nama**, yang menyatakan sebuah pelayanan (misalnya komputer atau jaringan); **alamat**, yang menunjukkan dimana (misalnya node) dan **route**, yang menunjukkan cara untuk bisa ke sana. Ini penting membuat tujuan antara nama-nama, dimana berupa alamat logika, dan alamat, yang berupa lokasi fisik. Alamat logika dapat menunjukkan satu atau lebih alamat fisik. Dengan menggunakan alamat logika, pengirim tidak perlu tahu lokasi fisik dari penenerima, dan entitas hubungan dapat menempatkan kembali tanpa perubahan alamat.

Transmission services. Suatu protokol mungkin diperlukan untuk menyediakan beberapa fasilitas tambahan, seperti hubungan dan mekanisme keamanan.

Suatu hubungan akan berupa gabungan antara entitas komunikasi, dan mekanisme hubungan membuat setiap entitas komunikasi dimana yang lain muncul, dapat diakses, dan dapat mendukung data flow yang diinginkan (yang bisa berupa simpleks, half dupleks, atau full dupleks; lihat Bagian 12.4). Mekanisme keamanan mungkin diperlukan, misalnya, untuk membatasi akses dan penyediaan integritas data.

Klasifikasi Protokol Jaringan

Jaringan protokol membentuk kerja sama herarki dan dapat diklasifikasikan ke dalam sejumlah katagori (Falk, 1983) :

1. **Protokol antarjaringan** menunjukkan prosedur komunikasi antara konstituen yang ditempatkan pada jaringan yang berbeda.
2. **Protokol jaringan** transparan bagi pemakai dan diperuntukkan untuk meyakinkan transmisi data yang dapat dipercaya dan efisien. Protokol ini dapat lebih lanjut dibagi sebagai berikut:
 - (a). **Protokol manajemen jaringan** yang digabungkan dengan pembuatan fasilitas transmisi internodal, routing, dan berbagai pelajaran manajemen jaringan seperti *user billing*, penambahan atau pengurangan pemakai atau node, kegagalan node, dan peluncuran serta perbaikan perangkat lunak jaringan.
 - (b). **Protokol transfer data**, yang terlibat dengan dukungan *datagram* dan jasa *virtual circuit*. (Strategi pengambilan ini akan didiskusikan dalam Bagian 12.6.).
3. **Protokol akses jaringan** digunakan oleh para pelanggan untuk berkomunikasi dengan node jaringan dimana pelanggan tersebut dihubungkan (attached). Klasifikasi ketiga ini mendukung adanya standar interfacing internasional termasuk interface nonstandard.
4. **Protokol end-to-end** menentukan prosedur untuk membuat session end-to-end. Suatu variasi dari adanya protokol yang mendukung penanganan akhir, transfer file, pemasukan remote job, pengiriman elektronik, facsimile, grafik, packe voice, dan teleconferencing.

12.3 TOPOLOGI JARINGAN KOMUNIKASI KOMPUTER

Topologi jaringan merupakan struktur yang ditentukan dengan skema hubungan internodal. Strukturnya dapat dilihat sebagai grafik dimana titik di atasnya meru-

pakan node pengambilan dan cabangnya merupakan channel transmisi. Salah satu aspek yang paling sukar dalam design jaringan komunikasi adalah pemilihan topologi (Cerf, 1981), yang mencakup beberapa problem teori dan praktek. Ini dihubungkan dengan problem yang lain, yaitu penyusunan kapasitas channel komunikasi untuk sistem. Untuk mengilustrasikan problem yang ada, perhatikan sirkulasi argumen berikut ini: Susunan kapasitas chanel tergantung pada lalu lintas yang diharapkan dan metode routing; metode routing dipengaruhi oleh pemilihan topologi; pemilihan topologi tergantung pada traffic.

Problemnya adalah satu dari optimasi dan mungkin dikatakan sebagai berikut: Dengan diberikan traffic matrix (yang menunjukkan jumlah traffic yang diharapkan antarnode) dan harga matrix (yang menunjukkan harga berbagai variasi channel yang menghubungkan node dan sebaliknya tergantung pada panjang dan kemampuan rata-rata data channel), meminimalkan performace fungsi (yang dapat berupa kapasitas harga, dalam dollar per unit kapasitas, katakanlah demikian) subyek ke beberapa konstrain. "Konstrain" dapat berupa **delay**, yang merupakan waktu dari penerimaan pesan oleh suatu jaringan hingga deliverinya, dan **throughput**, yang merupakan rata-rata transfer data yang efektif dalam bit per detik.

Seperti yang dapat diapresiasi, bahwa hal ini merupakan problem yang sangat komplek. Problem ini biasanya dibagi kedalam sejumlah subproblem yang disusun dengan peningkatan pesan dari kerumitan tersebut. Hanya metode (heuristik) yang telah ditunjukkan untuk pemecahan problem ini (Inose dan Saito, 1978).

Topologi Jaringan

Beberapa kemungkinan topologi jaringan ditunjukkan dalam Gambar 12-2. Dalam topologi **star** [Gambar 12-2(a), semua node dalam suatu jaringan dihubungkan dengan pusat switching node. Operasi suatu jaringan secara total bergantung pada pengontrol pusat. Jika pusatnya gagal, maka proses komunikasi dalam seluruh jaringan harus digantungkan. Misalnya, berbagai variasi jaringan remote-access dan LAN disusun sebagai star dan pusatnya dikontrol oleh circuit-switched PBX (private branch exchange) atau yang lebih modern CBX (computerized branch exchange).

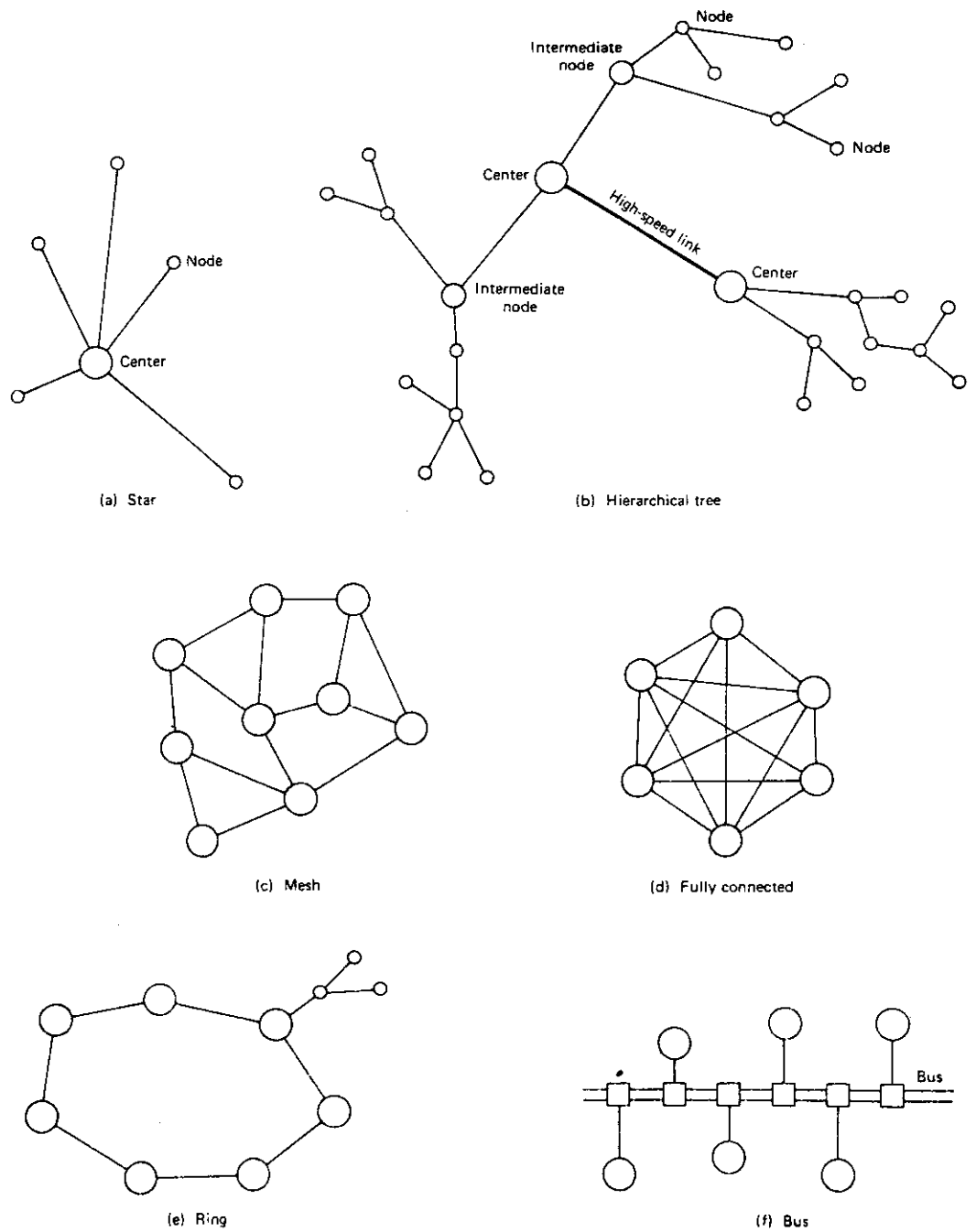
Tiga topologi yang ditunjukkan dalam Gambar 12-2 (b) mungkin atau mungkin juga tidak mempunyai node pusat. Jika mempunyai node pusat, maka akan ada lebih dari satu. Dengan menyediakan node lanjutan, topologi ini memungkin-

kan mengurangi total panjang cabang relatif pada topologi star. Apa yang sebenarnya ditunjukkan dalam Gambar 12-2 (b) adalah **hierachical** tiga jaringan dengan lebih dari satu pusat (pengontrol) dan dengan node lanjutan yang dapat beraksi sebagai konsentrator (multiplexer). Jika satu pusat gagal, maka suatu jaringan dapat melanjutkan beroperasi secara terpisah. Hampir semua jaringan remote-access merupakan topologi.

Topologi **mesh** yang ditunjukkan dalam Gambar 12-2 (c) merupakan yang paling umum dalam jaringan komunikasi data umum dan dalam jaringan komunikasi private modern. Meskipun total panjang cabang relatif lebih besar pada tiga topologi tersebut, topologi mesh merupakan suatu yang menguntungkan jika menyediakan suatu alternatif route (cabang) antarnode. Tiga cabang tentunya dapat keluar dari setiap atau beberapa node yang ditunjukkan. Topologi **hubungan penuh** diperoleh dalam Gambar 12-2 (d) yang ditunjukkan hanya untuk kelengkapan. Dengan node N , memerlukan $N(N - 1)$ cabang (full duplex) dan $(N - 1)$ dari bagian I/O untuk setiap node dan oleh karena itu tidak praktis.

Topologi **ring** yang ditunjukkan dalam Gambar 12-2 (e) adalah lebih umum ditemukan dalam LAN. Topologi dapat menggunakan kontrol pusat, dengan satu node yang disusun di tengah, atau kontrol desentralisasi, dengan semua node mempunyai status yang sama. Suatu jaringan mencakup serangkaian **repeaters** yang digabungkan dengan hubungan titik-ke-titik dalam loop yang tertutup. Setiap stasiun dilengkapi dengan jaringan repeater/pengulang. Paket data **dikirimkan** dan disirkulasikan secara sukses di sekitar ring. Setiap node mampu menerima masukan data, perpindahan data ke pelanggan yang dihubungkan dengannya (setiap node dapat mempunyai tiga cabang), atau pengiriman data ke node berikutnya. Tiga metode pemasukan ring pokok ini adalah **register insertion**, **slotted ring**, dan **token ring**.

Topologi terakhir yang diperhatikan, adalah **bus**, seperti ditunjukkan dalam Gambar 12-2. Secara dominan topologi ini digunakan dalam LAN, jaringan komunikasi adalah medium transmisi, yang ditangani oleh node yang dihadapi, dan tidak memerlukan pengambilan atau pengulangan. Paket data akan dikirimkan oleh satu stasiun yang dapat diterima oleh seluruh stasiun. Jika channel dihubungkan, maka hanya satu stasiun yang dapat mengirimkan pada waktu yang sama, oleh karena itu memerlukan beberapa bentuk kontrol akses. Kontrol ini dapat dipusatkan (didistribusikan) dalam bentuk protokol yang ditangani oleh seluruh node yang dihadapi, atau dapat dipusatkan. Dua metode kontrol pemasukan medium desentralisasi yang umum adalah **CSMA/CD** carrier-sense akses jamak dengan deteksi tubrukan (*collision detection*) dan **token bus**. Skema



Gambar 12-2 Topologi jaringan.

kontrol pemusatan ini digunakan dalam beberapa LAN yang disebut dengan **centralized reservation**.

12.4. TEKNIK TRANSMISI DATA

Topologi Link

Transfer data di antara dua stasiun, disebut dengan link **point-to-point**, yang bisa berupa tiga mode; **simplex**, **half duplex**, atau **full duplex**. Channel transmisi **simplex** dapat mentransfer informasi dalam hanya satu tujuan, seperti dalam penyiaran radio dan TV. Jika penerima tidak dapat berkomunikasi dengan pengirim, dan pengirim tidak dapat diinformasikan atas kesalahan pengiriman, maka channel **simplex** akan sukar digunakan dalam komunikasi data.

Channel transmisi **half-duplex** mampu dengan transfer dua tujuan, tetapi tidak simultan. Suatu informasi dapat dikirimkan dalam satu tujuan pada satu waktu, seperti dalam sistem komunikasi radio. Dua modem diperlukan untuk mengimplementasikan channel **half-duplex** ini. Pengiriman dalam satu tujuan, satu modem adalah pengirim dan modem yang lain adalah penerima. Setelah transfer lengkap, maka pengiriman dalam tujuan yang sebaliknya dipengaruhi oleh pengembalian aturannya; modem penerimaan akan menjadi pengirim dan modem pengiriman akan beraksi sebagai penerima. Waktu **putar** (**turnaround**) dalam mode **half-duplex** adalah waktu yang diperlukan untuk mengambil channel dari satu tujuan ke tujuan yang lain. Channel transmisi **full duplex** ada secara simultan, transfer dua tujuan, seperti dalam sistem telepon.

Jika ada lebih dari dua stasiun, maka topologi link disebut **multipoint**. Dalam hal ini stasiun yang menunjukkan perubahan informasi biasanya disebut dengan **stasiun primer** dan stasiun yang lainnya disebut dengan **stasiun sekunder**. Link **multipoint** dapat disusun dengan tiga cara sebagai berikut:

1. *Primary and secondary half-duplex*. Stasiun primer dapat mengirimkan ke satu stasiun sekunder dalam satu tujuan, setelah transfer lengkap, dapat menerima dari stasiun sekunder dalam tujuan lain.
2. *Primary full duplex, secondary half duplex*. Stasiun primer dapat mengirimkan ke satu stasiun sekunder dan secara simultan dapat menerima data dari yang lainnya.
3. *Primary and secondary full duplex*. Stasiun primer dan satu stasiun sekunder dapat mengirimkan ketika keduanya sedang menerima.

Mode Transmisi

Data digital dikirimkan secara **serial** dalam jaringan komunikasi komputer, atas channel komunikasi tunggal, dalam aliran bit yang dikelompokkan ke dalam **frames**. Untuk berkomunikasi dengan baik, beberapa pilihan sinkronisasi harus ada antara pengirim dan penerima. Penerima harus tahu (1) kapan frame memulai dan berakhir, dan (2) durasi setiap bit, sehingga dapat memberi contoh channel pada waktu yang semestinya untuk membacanya. Dengan jelas, bahwa meskipun waktunya sangat sedikit perbedaan antara pengirim dan penerima dapat menyebabkan adanya kesalahan, dengan penerima yang keluar dari sinkronisasi dengan pengirim.

Dua metode digunakan untuk mencapai sinkronisasi dua level. Salah satunya, disebut dengan *asynchronous transmission*, aliran bit dibagi ke dalam karakter dari panjang yang relatif kecil (secara khusus adalah 5 hingga 8 bit per karakter). Meskipun kesalahan waktu masih dapat terjadi, tetapi pembatasan panjang karakter dapat menunjukkan kurangnya kemungkinan kesalahan. Setiap karakter ditambahkan kedua bit, disebut dengan **start** dan **stop bit**, sebagai header dan trailer, secara berurutan. Meskipun interval waktu antara karakter yang sukses (kesuksesan bit start/stop) dapat berupa bersifat random, penerima dapat menyinkronkan kembali dirinya pada permulaan setiap karakter dan mengurangi efek waktu kesalahan kumulatif. Penambahan bit start/stop juga memungkinkan penerima untuk menentukan awal dan akhir setiap karakter. Transmisi yang tidak sinkron relatif sederhana untuk diimplementasikan tetapi menyebabkan **overhead** dari minimal 2 bit per karakter. Misalnya, jika karakter mempunyai panjang bit 8, maka overheadnya adalah $2/10 = 0.2$ atau 20 %. Overhead dapat dikurangi dengan penambahan panjang karakter, namun demikian akan berakibat dalam kesalahan waktu kumulatif yang lebih besar.

Skema sinkronisasi kedua, disebut dengan **synchronous transmission**, memungkinkan transmisi data dari block yang lebih besar. Untuk mengatasi perbedaan waktu antara pengirim dan penerima, pengirim akan mengirimkan informasi jam pada penerima pada baris yang terpisah atau memasangnya dalam data. Untuk memungkinkan penerima menunjukkan awal dan akhir blok data, maka setiap blok ditambahkan dengan informasi kontrol, yang disebut dengan **preamble**, dan **postable**, sebagai header dan trailer, secara respektif. Bagian dari informasi kontrol digunakan untuk tujuan kontrol link data (misalnya kesalahan dan flow control), tetapi pola bit khusus menunjukkan awal dan akhir setiap frame. Jika field data dari setiap frame berupa panjang yang tidak selalu sama (minimal dalam prinsip) maka transmisi sinkronisasi overhead dapat sangat kecil.

Multiplexing

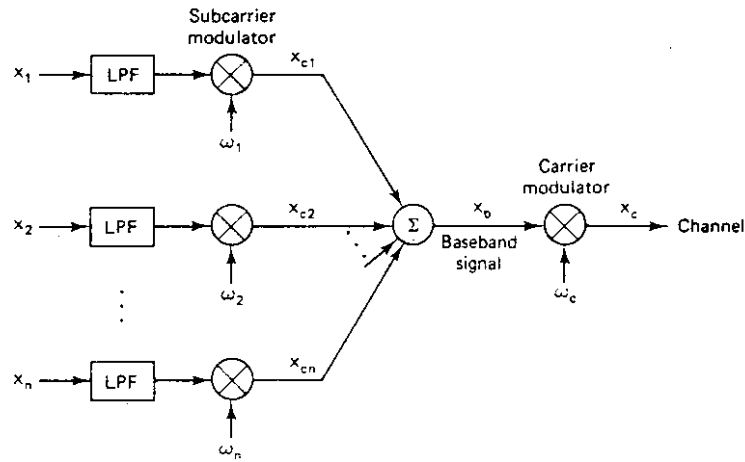
Fasilitas transmisi sering mahal, dan sistem komunikasi tidak selalu menggunakan kapasitas yang penuh dari channel data. Oleh karena itu, yang lebih ekonomis kita dapat melakukannya dengan memperlakukan fasilitas transmisi, suatu proses yang disebut dengan **multiplexing**. Kita akan melihat pada prinsip dari tiga skema multiplexing yang hampir pada umumnya dipakai dalam jaringan komunikasi komputer.

Frequency division multiplexing (FDM). FDM digunakan ketika bandwidth channel melebihi bandwidth total yang diperlukan oleh sinyal transmisi. Sinyal tersebut melewati multiplex untuk menangani bandwidth channel dan dikirimkan secara simultan. Setiap sinyal dimodulasikan untuk mencukupi frekuensi band dari channel yang tidak tercukupi oleh sinyal yang lain. Informasi dari berbagai variasi sinyal berisi dalam frekuensi ikatan channel yang tidak overlapping.

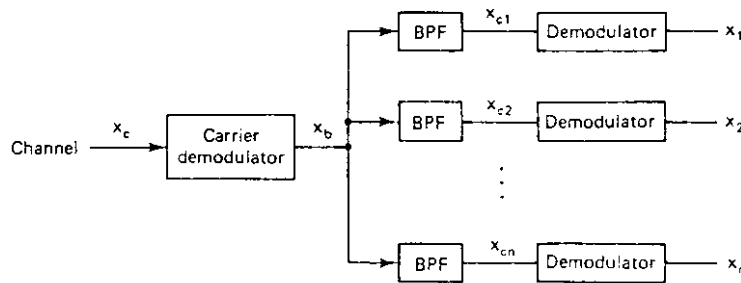
Proses FDM ditunjukkan dalam Gambar 12-3 (a). Anggap bahwa ada sinyal perbedaan n , x_1, x_2, \dots, x_n , masing-masing dimodulasikan dengan frekuensi sub pembawa $w_1, w_2, w_3, \dots, w_n$, secara berurutan. Frekuensi sub pembawa dipilih bagian yang cukup jauh sehingga setiap sinyal spektrum terpisah dari semua yang lainnya. Ini menunjukkan bahwa setiap frekuensi sub pembawa dipisahkan dari sub pembawa yang dekat dengan minimal $2w_m$. Oleh karena itu, jika frekuensi sub pembawa telah dipilih dengan semestinya, maka operasi multiplexing akan menyusun slot dalam frekuensi domain untuk setiap pesan individual dalam bentuk yang dimodulasi. Setiap pesan input akan memodulasi sub pembawa yang semestinya, setelah kemungkinan berjalan ke low-pass filter (LPF) untuk membalasi bandwidth. Sinyal modulasi yang dihasilkan, $x_{c1}, x_{c2}, \dots, x_{cn}$, dijumlahkan untuk memperoleh apa yang disebut dengan **sinyal Baseband** x_b . Sinyal baseband kemudian, mungkin, dikirimkan baik langsung ke channel atau mungkin digunakan untuk memodulasi pembawa w_c , dengan sinyal yang dihasilkan x_c dikirimkan.

Pemulihan pesan, atau **demultiplexing**, ditunjukkan dalam Gambar 12-3 (b), disusun dengan tiga langkah. Pertama, demodulator pembawa akan memproduksi kembali sinyal baseband. Kemudian sub pembawa yang dimodulasi dipisahkan dengan menggunakan bandpass filter (BPF). Akhirnya, setiap pesan secara individual didemodulasikan dan dipulihkan.

Time division multiplexing (TDM). TDM digunakan jika channel data melebihi rata-rata total data dari pesan yang dikirimkan. Proses TDM ditunjukkan dalam Gambar 12-4. Kita anggap bahwa ada sinyal n , setiap prefilter (atau biffer)



(a) Multiplexing process



(b) Demultiplexing process

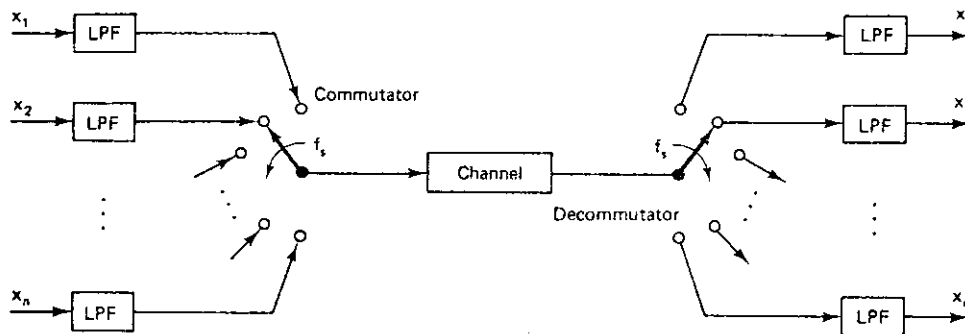
Gambar 12-3 Frequency division multiplexing (FDM).

dengan low-pass filter (LPF). Pada sisi pengirim, sinyal disederhanakan dengan rotasi switch, yang disebut dengan **commulator**, yang menyaring satu contoh dari setiap input per revolusi/perubahan. Contoh-contoh kemudian disisipkan dan dikirimkan ke channel pengiriman tunggal. Pada sisi penerima, rotasi switch yang sama, yang disebut dengan **decommutator**, memisahkan contoh dan mendistribusikannya melalui LPF (atau buffer) untuk penyusunan kembali pesan individual.

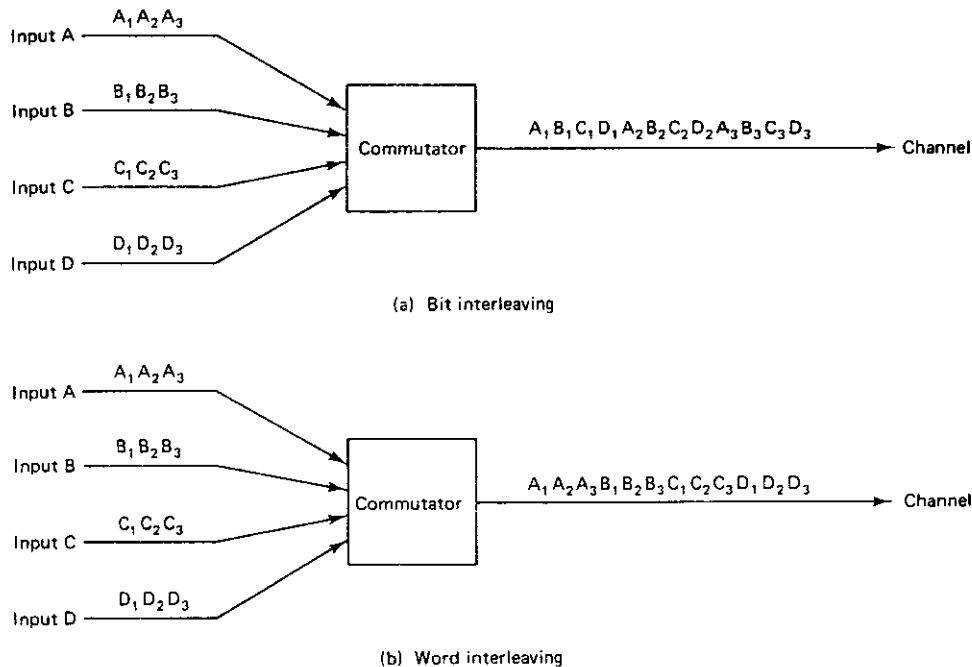
Jika semua input mempunyai pesan bandwidth W yang sama, maka commutator harus berotasi pada rata-rata $f_s \geq 2W$, sehingga contoh yang sukses dari suatu input diberi jarak dalam waktu dengan $T_s = 1/f_s \leq 1/2W$. Dengan sinyal input n , maka spasi pulse-to-pulse adalah $T_s/n = 1/nf_s$, dan rata-rata pulse dari sinyal TDM adalah $r = nf_s \geq 2nW$.

Hampir semua sistem TDM secara praktis menggunakan pengambilan elektronik untuk memmbuat sinyal multiplexed. Namun demikian, perhatikan bahwa TDM memerlukan sinkronisasi antara commutator dan decommutator. Ini adalah titik dalam TDM jika setiap contoh harus didistribusikan ke baris output yang benar pada waktu yang semestinya. Untuk alasan ini, TDM juga disebut dengan **synchronous TDM**.

Akhirnya mari kita perhatikan secara ringkas TDM dalam konteks sinyal digital. Jika suatu pesan input mempunyai rata-rata bit yang sama, maka multiplexing dapat dikerjakan pada basis bit-by-bit, yang disebut dengan **bit interleaving**, yang ditunjukkan dalam Gambar 12-5 (a), atau pada basis kata-per-kata (word-by-word) yang disebut dengan **word interleaving** dan ditunjukkan dalam Gambar 12-5 (b). Jika rata-rata bit dari pesan yang masuk tidak sama, maka input ditempatkan dengan slot waktu yang lebih dalam proporsi pada rata-rata bit, seperti diilustrasikan dalam Gambar 12-6. Dari empat input multiplexed yang ditunjukkan dalam Gambar 12-6, tiga mempunyai rata-rata bit yang sama (B, C, dan D), ketika rata-rata bit dari input A adalah tiga kali lebih tinggi. Seperti yang dapat dilihat, bahwa input A ditempatkan (secara proporsional) dengan slot waktu yang lebih dengan contoh commutator yang lebih sering.



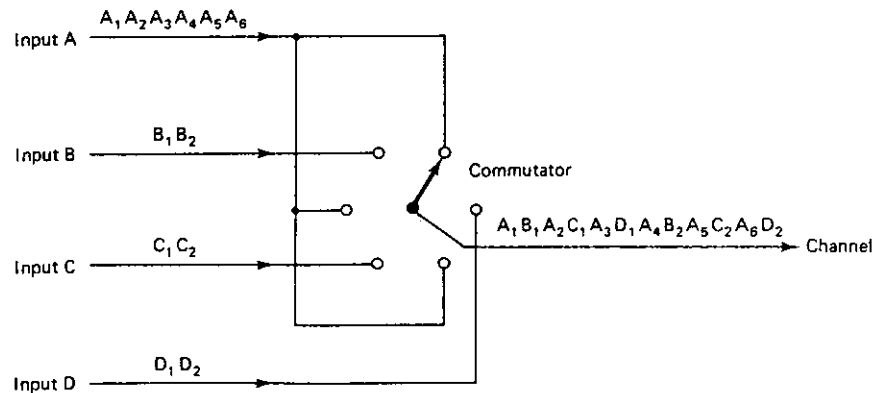
Gambar 12-4 Time Division Multiplexing (TDM)



Gambar 12-5 Time division multiplexing sinyal digital memiliki rata-rata yang sama.

Statistical time division multiplexing (STDM). Sinkronisasi TDM memberikan alokasi waktu slot pada basis yang telah ditentukan, Pada suatu waktu yang diberikan, beberapa input mungkin tidak mempunyai data untuk dikirimkan, dengan menghasilkan dalam slot waktu yang sia-sia. STDM menyediakan suatu alternatif dengan *secara dinamis* penyediaan alokasi waktu menurut kebutuhan input. Hal ini menjaga semua waktu yang dimasukkan dengan data sepanjang ada data yang menunggu untuk dikirimkan, dan akan mengubah alokasi slot, jika perlu, untuk menyusunnya. STDM menerima efisiensi ini dengan melewati input idle, dengan hasil bahwa rata-rata output data dari baris multiplexed kurang dari total rata-rata data input. Dengan demikian STDM dapat melayani sebanyak input TDM tetapi memerlukan baris rata-rata lower data, atau, dengan menggunakan baris yang sama, dia dapat menyajikan input yang lebih dibandingkan dengan TDM.

Setiap input pada STDM disusun buffer yang dapat berisi minimal beberapa karakter. Mekanisme kontrol buffer monitor buffer dan instruksi multiplexer



Gambar 12-6 Time division multiplexing sinyal digital memiliki rata-rata bertalian.

adalah contoh dengan pengisian yang sangat tinggi. Untuk tujuan identifikasi, setiap karakter atau blok data yang dikirimkan ditentukan dengan alamat yang dihadapi padanya dengan multiplexer. Pembagian waktu yang modern cenderung menjadi type STDMA. Beberapa telah ditambahkan ciri kontrol yang mampu beroperasi sebagai terminal inteligen yang tidak hanya data multiplex dan demultiplex tetapi juga menggunakan data.

12.5. MEDIA TRANSMISI

Transmission medium merupakan suatu fasilitas yang membawa data dari satu lokasi ke lokasi yang lainnya. Jaringan komunikasi komputer dapat menggunakan satu atau kombinasi media pengiriman, yang dapat disediakan dengan beberapa vendor. Medium transmisi dapat memandu secara fisik seperti kabel telepon dua pasang, kabel koaksial, atau kabel serat optik, atau transmisi dapat “tanpa kabel”, dengan menggunakan radio, mikrowave, atau channel satelit.

Pasangan ganda hanpir umum digunakan dalam sistem telepon. Kabel ini mudah untuk disusun tetapi relatif rendah suara imunitasnya. Kabel ini menawarkan rata-rata data hingga 9.6 kbps dalam hubungan loop lokal, tetapi dapat mencapai rata-rata data yang lebih tinggi ketika digunakan untuk aplikasi jarak jauh. Kabel twisted pair mempunyai bandwidth pada pesan 200 kHz dan memmer-

lukan pengulang setiap 2-6 km. Kabel koaksial menawarkan suara yang lebih baik imunitasnya dibandingkan dengan twisted pair tetapi lebih sukar disusunnya. Kabel ini dapat mengakomodasi rata-rata data yang tinggi hingga 500 Mbps dengan bandwidth pada pesan 300 MHz, dan memerlukan pengulang pada interval 1-10 Km. Kabel fiber-optik mempunyai imunitas suara yang tinggi, dapat mendukung rata-rata data hingga 1 Gbps dengan bandwidth 1 GHz, dan memerlukan pengulang setiap 10-100 Km.

Dengan menggunakan radio atau mikrowave seperti pengiriman medium akan menghilangkan keperluan hubungan fisik antara stasiun dan fasilitas komunikasi yang mempunyai jarak jauh. Media tanpa kabel dapat digunakan dalam penyiaran atau hubungan point-to-point, mempunyai kemampuan wide-band, yang dapat diimplementasikan secara cepat, dan mendukung perpindahan stasiun. Namun demikian, mereka merupakan subyek untuk interference dan propaganda (anomalies) dan memerlukan elaborasi yang lebih untuk keamanan. Komunikasi satelit merupakan teknik yang terbaru yang menawarkan beberapa keuntungan yang menarik. Hubungan dengan jaringan dapat sesederhana pengambilan antena pada satelit. Harga transmisi bebas dari jarak dalam area yang dicakup oleh satelit, kurang lebih seperempat permukaan bumi dengan satelit tunggal. Suatu komunikasi dapat berupa point-to-point atau penyiaran, dengan kualitas transmisi yang tinggi dan bandwidth yang sangat besar.

12.6. TEKNIK SWITCHING/PENGAMBILAN JARINGAN

Ciri pokok dari jaringan komunikasi komputer adalah teknik pengambilan yang diterapkan. Jaringan komunikasi komputer dapat diklasifikasikan sebagai jaringan switched maupun jaringan broadcast. "Switched network" disusun atas koleksi node pengambilan melalui pesan yang melalui rute dari sumber ke tujuan. Sebaliknya, **broadcast network** tidak mempunyai node, dan pemakainya menangani medium transmisi umum ke dalam pesan yang disiarkan oleh seseorang dan diterima oleh orang lain. Kita hanya ingin mengamati jaringan switched dan akan mendiskusikan prinsip tiga teknik pengambilan (switching): sirkuit switching, message switching, dan packet switching. Namun demikian, sebelum membicarakannya, mari kita uji dulu prinsip pokok dari teknik switching.

Tujuan dari strategi pengambilan adalah mengijinkan transfer pesan "*end-to-end*". Suatu stasiun memerlukan service semacam ini yang memberi inisial

dengan menentukan node switching ke arah adanya switching code. Proses pemberian inisial ini harus minimal mencakup alamat tujuan, yang mungkin ditambahkan pada atau dipancarkan dalam suatu pesan. Berdasarkan pada alamat tujuan, node switching akan menyusun **route** untuk suatu pesan. Route mungkin berupa end-to-end atau hanya per bagian saja, dimana switching node memutuskan/menentukan pada node berikutnya yang akan menangani kelanjutan pengaturan panggilan (call). Semua ketentuan tersebut berdasarkan pada efek **scheme routing**.

Jika ada alternatif path untuk mentransfer suatu jaringan dari suatu sumber ke suatu tujuan, proses routing bukan merupakan tugas yang tidak penting. Ketentuan routing mungkin berdasarkan pada variasi obyektif, yang mungkin di antaranya ada yang konflik. Untuk menyebutkan sedikit kemungkinan, suatu obyektif mungkin di-routekan oleh path yang paling pendek (overall), delay pesan minimum, harga komunikasi yang minimum, maksimum throughput, atau kesamaan dalam menggunakan sumber transmisi. Dengan demikian, misalnya, untuk meminimalkan pesan delay di bawah kondisi perubahan load, beberapa pesan kontrol (overhead) harus diubah antara berbagai variasi node dengan cara, yang dihasilkan dalam jaringan throughput yang berkurang. Sebaliknya, dengan memaksimalkan throughput mungkin akan mempengaruhi delay yang telah dinaikkan. Demikian juga, dalam traffic builds akan berpengaruh pada kepadatan beberapa link, route oleh path yang paling pendek mungkin dapat memberikan hasil yang tidak diinginkan.

Setiap node "switching" berisi **routing table**, yang memberikan daftar semua kemungkinan route yang ada di antaranya dan antara node yang lain. Jika suatu tabel telah ditentukan dan tidak bervariasi dengan pola traffic, maka routing adalah **static**. **Dynamic routing**, yang juga disebut dengan **adaptive routing**, menggunakan tabel yang mengubah sebagai suatu fungsi dari aliran traffic dan usaha untuk memberikan route terbaik pada waktu yang berbeda. Tabel pemberian route ini dapat diupdate dari fasilitas jaringan pusat, dimana skema route disebut dengan **centralized**, atau dapat dimutakhirkan oleh setiap node, yang berpengaruh terhadap skema route **decentralized**. Sekarang mari kita perhatikan pada tiga strategi switching.

Sirkuit Switching

Sirkuit switching telah digunakan dalam waktu yang lama dalam perubahan telepon dan service teleks. Suatu jaringan sirkuit-switched akan membuat

dedicated circuit end-to-end (path) antara entiti komunikasi. Path tidak selalu berupa kabel fisik yang menghubungkan dua stasiun komunikasi. Jaringan sirkuit-switched yang modern menggunakan bagian waktu multiplexing (di mana stream bit dari sumber yang berbeda di-interleaved; lihat bagian 12.4).

Proses komunikasi yang tercakup adalah meliputi tiga tahap. Dalam tahap yang pertama, sirkuit end-to-end akan diatur. Kemudian suatu sirkuit dijaga untuk durasi transfer pesan. Akhirnya, setelah proses transfer selesai, sirkuit tidak dihubungkan lagi. Untuk mengatur suatu hubungan, kedua stasiun tersebut, termasuk sumber transmisi yang diperlukan, harus ada pada waktu yang sama *sebelum* perubahan dapat memulai. Prosedur pengaturan memperkenalkan delay ke dalam proses komunikasi keseluruhan. Jika suatu pernyataan diperlukan, maka delay ini akan berupa kumulatif dan didatangkan *sebelum* suatu pesan dapat dikirimkan. Semua sumber yang diperlukan harus diperoleh dari suatu durasi, meskipun tidak ada informasi yang diubah. Hal ini mungkin sangat cocok dalam aplikasi yang sesungguhnya seperti suara komunikasi atau jika lanjutan flow data yang ada di dalamnya, tetapi tidak efisien dan sia-sia dari kapasitas channel. Akhirnya tujuannya juga memperkenalkan suatu delay.

Untuk mengatasi hal yang tidak efisien ini, beberapa jaringan komunikasi data sirkuit-switched yang menggunakan pengaturan yang sangat cepat dan mekanisme yang tidak dihubungkan bahwa sering memungkinkan hubungan switched untuk digunakan hanya dari durasi satu pesan atau ke satu pesan dan respon interaktif. Setelah itu, sirkuit dihubungkan, dan suatu sumber digunakan untuk menangani pesan yang lain, untuk dihubungkan lagi jika pesan yang kedua telah ada. Kemungkinan pendekatan lain, dalam konteks perubahan data digital, adalah message switching, yang akan kita bicarakan nanti.

Jaringan sirkuit-switched dapat berupa **bloking** maupun **nonbloking**. Jika suatu jaringan tidak dapat menghubungkan dua stasiun karena semua path di antaranya sedang digunakan, maka akan terjadi bloking. Jaringan nonbloking memungkinkan setiap pasang stasiun untuk dihubungkan sekali. Dimana bloking dapat menerima dalam jaringan suara telepon, dia tidak dapat diterima jika transfer data tercakup di dalamnya; maka di sini diperlukan susunan jaringan nonbloking.

Message Switching

Transmisi data interaktif sering mendadak rusak dan memiliki karakter **peak-to-average ratios (PAR)** tinggi. PAR menilai ratio antara *peak transmission rate*, yang

ditentukan oleh respon waktu yang diperlukan dari stasiun, dan *average transmission rate* selama satu session. Misalnya, stasiun grafik kompleks mungkin memerlukan PAR dari 1000, katakanlah, sebagai kebalikan dari 20 untuk terminal dumb.

Untuk data dengan PAR tinggi, sirkuit switching agak tidak efisien, dan message switching mungkin lebih tepat. Dengan message switching, stasiun terakhir tidak mempunyai path yang dipersembahkan di antaranya. Pesan data dikirimkan ke switching node, dimana pesan data tersebut disimpan dalam suatu antrian (queue). Ketika mencapai ujung antrian, kemudian dikirimkan ke node berikutnya jika suatu link ada. Proses ini berlanjut hingga suatu jaringan ditransfer dari sumber ke suatu tujuan. Jika suatu pesan disimpan dalam node switching dan kemudian berlanjut ke node berikutnya pada route, maka message switching juga disebut dengan **store-and-forward switching**. Suatu pesan diperlakukan sebagai entiti melalui jaringan dan dikirimkan melalui route yang sama.

Setiap pesan berisi header yang mencakup alamat tujuan, dan satu pesan dapat dikirimkan ke beberapa tujuan. Prioritas dapat juga dibuat dalam header, dengan memungkinkan pengambilan node untuk memproses pesan dengan prioritas tinggi sebelum adanya prioritas yang rendah. Jika suatu pesan disimpan secara temporer pada setiap node-nya, maka periode traffic yang paling sibuk dapat ditunjukkan. Misalnya, pesan dengan prioritas rendah dapat lebih lama di antrian. Pengambilan node juga menyediakan pengontrolan kesalahan dan service perbaikannya.

Packet Switching

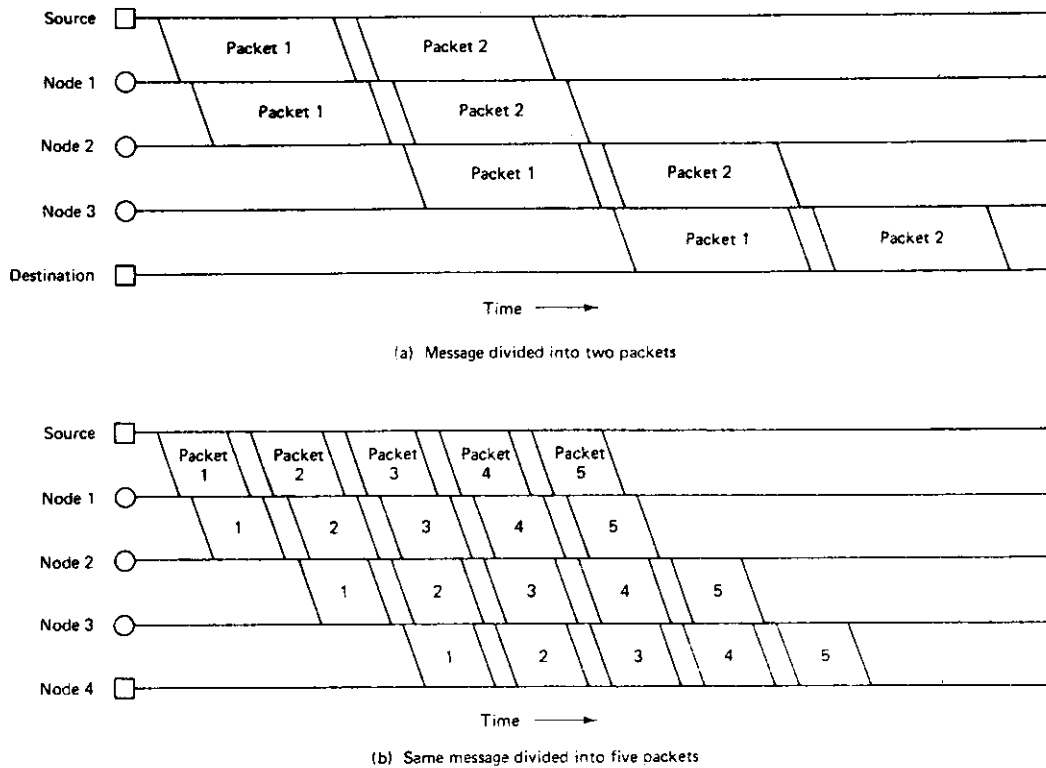
Pengambilan paket akan beroperasi dengan prinsip yang sama seperti penyimpanan dan pengiriman data; namun demikian ini berbeda dengan message switching oleh satu ukuran pesan. Seperti yang akan kita lihat, dia dapat dioperasikan dengan hubungan kecepatan tinggi antara pengambilan node dan cenderung meminimalkan delay dengan efisien.

Suatu jaringan packet-switched membagi data traffic ke dalam blok, yang disebut dengan **packet**, dari beberapa panjang maksimum yang diberikan. Panjang maksimum secara khusus adalah beberapa ribu bit, merupakan kebalikan dengan message switching, dimana pesan yang lebih besar akan mungkin. Setiap paket secara individual dilindungi dengan mekanisme deteksi kesalahan dan berisi didalam header sumber dan alamat tujuan.

Ketika keluar, panjang paket yang terbatas mempunyai akibat yang dapat ditandai pada delivery delay suatu pesan. Untuk mengilustrasikan hal ini, perhatikan Gambar 12-7, yang menunjukkan diagram waktu pengiriman pesan dari sumber ke tujuan melalui tiga node. Stasiun sumber didapatkan untuk node 1 dan stasiun tujuan dengan node 3. Seperti yang dapat kita lihat dari gambar tersebut, bahwa pembagian pesan ke dalam paket (bagian (a)) akan menghasilkan dalam delivery waktu yang lebih lama dibandingkan dengan jika dibagi ke dalam lima paket [bagian (b)]. pengambilan paket dapat menggunakan link dengan kecepatan tinggi. Ukuran paket yang terbatas juga menyebabkan persyaratan penyimpanan dalam setiap pengambilan node. Tidak seperti message switching, yang memerlukan media penyimpanan dan kemampuan pemrosesan, suatu packet switching secara khusus memerlukan kapasitas penyimpanan yang lebih kecil, yang lebih mudah diatur.

Setiap pesan ditangani sebagai entiti tunggal melalui suatu jaringan, dan deliverynya ditangani sebagai message switching. Namun demikian kebalikan dari message switching, dimana seluruh pesan diperoleh melalui route yang sama, membaginya ke dalam paket dengan ukuran yang telah ditentukan yang menunjukkan dua mekanisme (protokol) yang berbeda untuk packet switching, suatu diagram dan sirkuit virtual. Di dalam strategi **datagram**, setiap paket yang sekarang disebut dengan datagram, merupakan pengajuan yang bebas pada suatu jaringan dan membawa semua informasi yang diperlukan untuk delivery-nya pada suatu tujuan. Jika tidak ada gabungan priori di antara datagram, maka masing-masing dapat menggunakan route yang berbeda. Dengan demikian order dimana pengirim memasukkan datagram ke dalam suatu jaringan mungkin berbeda dengan pesan dengan datagram yang sama yang dipindahkan dari suatu jaringan oleh penerima. Oleh karena itu, paket mungkin tiba pada suatu tujuan yang salah, yang memerlukan penyusunan kembali untuk menyimpan kembali pesan aslinya. Dengan demikian setiap datagram harus berisi angka yang sesuai yang menunjukkan posisinya di dalam pesan tersebut.

Dalam strategi **virtual sirkuit**, hubungan logika (virtual) harus diatur antara pasangan stasiun yang ingin berkomunikasi sebelum dapat dibuat pengiriman data. Hal ini menunjukkan suatu prosedur pengaturan, tahap transfer data, dan prosedur pemutusan. Namun demikian tidak seperti pengambilan sirkuit, dimana suatu hubungan diperoleh, hubungan logika berarti bahwa setiap stasiun dapat mempunyai lebih dari satu stasiun. Jika hubungan logika dibuat dengan lebih baik, maka pengambilan node tidak diperlukan untuk membuat keputusan routing, dan order delivery paket akan menggabungkan order pemasukan paket ke dalam jaringan.



Gambar 12-7 Efek ukuran paket dalam delivery delay.

Suatu paket harus disusun kembali dan jika tidak dalam suatu rangkaian, juga harus disusun kembali. Proses ini diimplementasikan dalam perangkat lunak pada node yang memerlukan untuk menangani beberapa pesan secara simultan, yang menghasilkan kemungkinan deadlocks yang harus dipecahkan. Juga, ada overhead yang digabungkan dengan pemaketan. Paket yang lebih kecil dapat dikirimkan dengan cepat, tetapi overheadnya akan lebih tinggi. Ini berarti bahwa ada beberapa ukuran paket maksimum yang menyeimbangkan delay dan efisiensi.

12.7. RINGKASAN

Kita telah membicarakan tambahan cukup banyak dari perkembangan secara revolusi— jaringan komunikasi komputer. Bibliografi pada akhir buku ini mencakup beberapa referensi yang dapat mendukung pengertian yang lebih mendalam atas aspek teknik dari subyek ini.

Beberapa bagian pembuatan mencakup pengkodean data dan deteksi kesalahan dan mekanisme pembetulan; rancangan protokol, skema route, switching, dan strategi broadcasting, pengaturan waktu, pembuatan frame, sinkronisasi, kesalahan, dan flow control dan internetworking—kemampuan untuk menghubungkan berbagai variasi jaringan yang berbeda sehingga dua stasiun dengan jaringan yang konstituen dapat berkomunikasi.