

Concepte ale rețelelor și stiva de protocoale TCP/IP

Obiective:

Se va prezenta conceptul de schimb de date și modul în care datele sunt transmise într-o rețea

Se va defini conceptul de rețea, criteriile și topologiile acestora

Se vor discuta structura rețelelor de tip LAN și WAN, precum și conceptul de internetwork

Se va analiza conceptul de comutare (switching) pentru a arăta cum rețelele mici pot fi combinate în unele mai mari

Se vor prezenta tipurile de standarde care se referă la rețelele de calculatoare și organizațiile care se ocupă de elaborarea acestor standarde

Se va prezenta conceptul de arhitectură de rețea pe nivele de protocoale, făcând analogie cu două scenarii din viața reală

Se vor prezenta două principii pe care se bazează arhitectura pe nivele de protocoale, precum și conceptul de conexiune logică între nivelele corespunzătoare ale sursei și destinației

Se va prezenta suita de protocoale TCP/IP pe cinci nivele, denumirea pachetelor de date la fiecare nivel, mecanismul utilizat pentru adresare la fiecare nivel, funcțiile fiecărui nivel și relațiile dintre nivele

Se va realiza o comparație a modelului de referință OSI versus suita de protocoale TCP/IP

1.1. Schimbul de date (sisteme de comunicație)

Comunicând schimbăm între noi informația deținută => partajăm informație

Partajarea informației se poate face local (prin comunicare *face to face*) sau de la distanță

Termenul tehnic utilizat *telecomunicații* (include telefonica, telegraful și televiziunea) prevede comunicarea de la distanță („tele„ în greacă veche înseamnă „departe”)

Date => informația partajată, care este prezentată în careva formă

Comunicarea (schimbul) de date => schimbul de date între două sau mai multe dispozitive, folosind un mediu de transmisie a datelor (cablul sau aerul)

Sistem de comunicație => hard-ul (echipamentul fizic) + soft-ul (aplicațiile utilizate) necesar pentru a realiza legătura între dispozitive și transmiterea de date

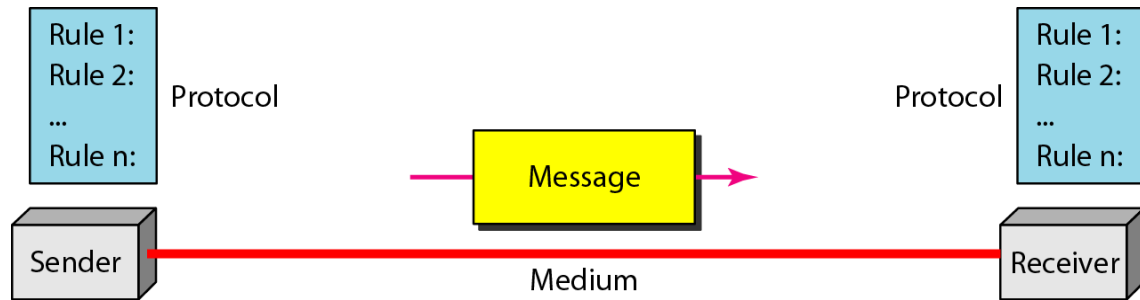


Eficiența sistemului de comunicație poate fi descrisă prin următoarele caracteristici:

- *Livrarea datelor.* Sistemul trebuie să furnizeze date la destinație. Datele trebuie să fie recepționate de dispozitivul sau utilizatorul destinatar și numai de respectivul dispozitiv sau utilizator.
- *Acuratețea livrării.* Sistemul trebuie să furnizeze cu acuratețe înaltă datele către destinatar. Nu pot fi utilizate datele care au fost modificate pe parcursul transmisiei și nu au fost corectate erorile din acestea.
- *Timpul de livrare.* Sistemul trebuie să furnizeze date în timp util. Datele livrate cu întârziere sunt inutile. În cazul transmisiei video sau audio, livrarea la timp înseamnă livrarea de date pe măsură ce sunt generate, în aceeași ordine în care sunt generate și fără întârzieri semnificative. Acest tip de livrare se numește transmisie în timp real.
- *Bruiajul.* Acesta se referă la variația timpului de sosire a pachetelor. Pachetele audio sau video pot fi livrate cu întârziere. De exemplu, pachetele video sunt trimise la fiecare 30 ms. Dacă unele dintre pachete ajung cu 30 de ms întârziere, iar altele cu 40 ms întârziere, apare bruiatul (calitatea proastă a videoclipului).

Un **sistem de comunicație** a datelor poate fi descris prin 5 componente:

- **Mesajul** – datele ce urmează a fi comunicate (text, valori numerice, imagini, audio și video)
- **Expeditorul** – dispozitivul (calculator, telefon, video cameră ș.a.) ce transmite mesajul
- **Destinatarul** – dispozitivul ce recepționează mesajul (calculator, telefon, televiziune ș.a.)
- **Mediul de transmisie** – calea fizică (cablu cu perechi răsucite, cablu coaxial, cablu cu fibră optică, unde radio ș.a.) prin care se transmite mesajul de la sursă la destinație
- **Protocolul** - un set de reguli care guvernează comunicațiile de date. Reprezintă un acord între dispozitivele care comunică. Fără un protocol, două dispozitive pot fi conectate, dar nu comunică, la fel cum o persoană care vorbește franceza nu poate fi înțeleasă de o persoană care vorbește doar japoneza.



Reprezentarea datelor

Numere – sunt reprezentate ca șiruri de biți. Numărul este convertit direct într-un șir binar

Text – simbolurile de text sunt reprezentate ca șiruri de biți (prin codificare)

Codificarea Unicode – folosește 32 de biți pentru a reprezenta un simbol sau un caracter folosit în careva limbă din lume

Codificarea ASCII (American Standard Code for Information Interchange) constituie primele 127 de caractere din Unicode (se mai numesc **Basic Latin**). Simbolului    este asociat un num  r   ntreg pozitiv, iar acesta este convertit   ntr-un șir de biți

Imagini – sunt reprezentate ca șiruri de biți. Imaginea 2D reprezint   un tablou bidimensional de pixeli (un punct mic ce reprezint   elementul de imagine). Rezoluția imaginii este caracterizată prin num  rul de pixeli ce o formeaz  . O imagine cu o rezoluție mai mare (cu un num  r mai mare de pixeli) este mai calitativ  , dar necesit   mai mult   memorie pentru stocare

Fiec  rui pixel al imaginii i se asociaz   un șir de biți, de lungime care depinde de scala de culori utilizat  . Pentru imaginile color se folosesc metode de reprezentare binar  , precum RGB (red, green, blue) sau YCM (yellow, cyan, magenta)

Audio –   nregistrarea sau difuzarea sunetelor. Spre deosebire de texte, numere și imagini, audio este de natur   continu   și nu discret  .

Video - –   nregistrarea animat   a imaginilor și a filmelor. Poate fi o combinație de imagini (de natur   discret  ) aranjate astfel   nc  t s   simuleze deplasarea, sau poate fi de natur   continu   (  nregistr  ri realizate de o camer   TV)

Fluxul de date

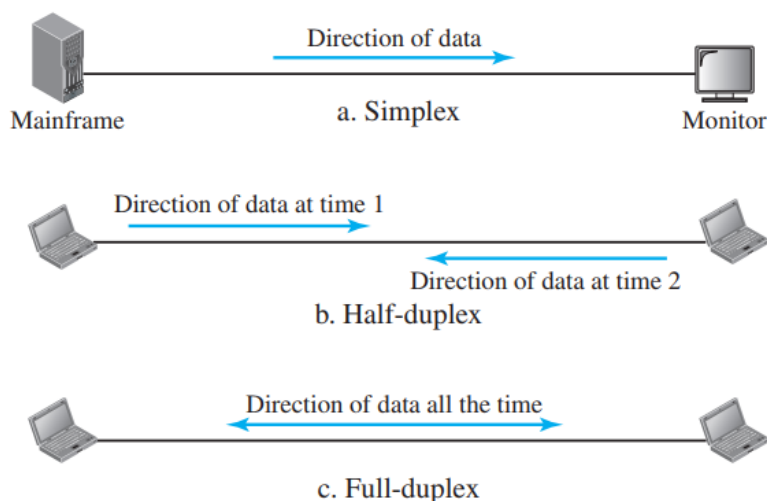
Modul simplex – comunicare unidirecțională. Doar unul din dispozitivele ce asigură legătura poate să transmită date; celălalt poate doar să recepționeze date. Astfel este utilizată întreaga capacitate a canalului de transmisie. Dispozitive simplex – tastatura

Modul half-duplex – fiecare dispozitiv poate atât să transmită, cât și să recepționeze, dar nu în același timp. Când un dispozitiv transmite date, celălalt doar recepționează. Modul semi-duplex este ca un drum cu o singură bandă cu trafic permis în ambele sensuri. Când un grup de mașini se deplasează într-o direcție, mașinile care intenționează să meargă în direcție opusă trebuie să aștepte. Într-o transmisie half-duplex, întreaga capacitate a unui canal este preluată de oricare dintre cele două dispozitive, de la care se transmit date la momentul respectiv.

Modul full-duplex – ambele dispozitive pot să transmită și să recepționeze date simultan. Modul full-duplex este ca o drum cu două sensuri, cu trafic care curge în ambele direcții în același timp. În modul full-duplex, semnalele care merg într-o direcție împart capacitatea link-ului (conexiunii) cu semnalele care merg în direcție opusă. Această partajare poate avea loc în două moduri: sau linkul trebuie să conțină două căi de transmisie separate fizic, una pentru trimitere și cealaltă pentru recepționare; sau mediul de transmisie este împărțit între semnalele care circulă în ambele direcții.

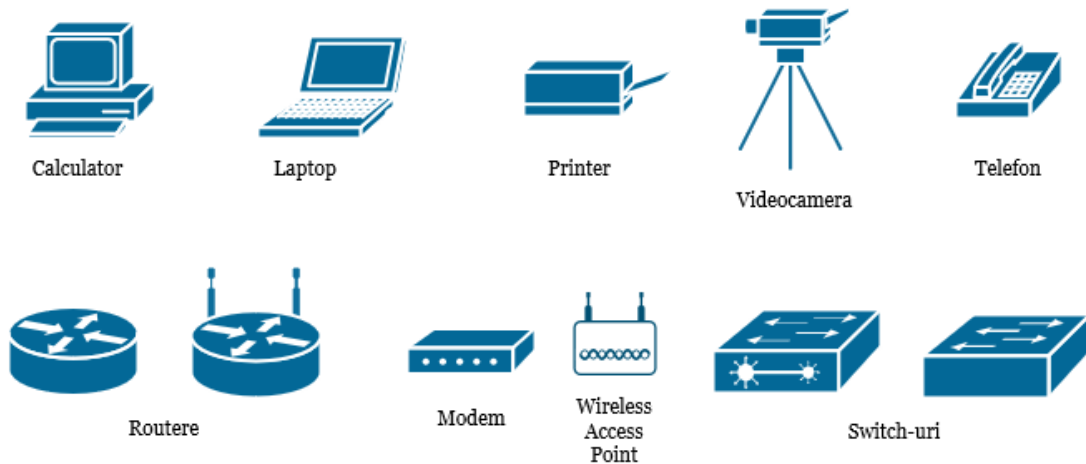
Un exemplu clasic de comunicare full-duplex este rețeaua de telefonie. Când două persoane comunică prin intermediul unei linii telefonice, ambele pot vorbi și asculta în același timp.

Figure 1.2 Data flow (simplex, half-duplex, and full-duplex)



1.2. Rețele de comunicație

O rețea de comunicație reprezintă un număr finit de dispozitive de comunicație (numite noduri), care sunt conectate prin intermediul unor tehnologii de comunicație (numite link-uri). Nodurile pot să fie host-uri (calculatoare, notebook-uri, imprimante, dispozitive mobile, sisteme de securitate ș.a.) și dispozitive de rețea (routere, switch-uri, modem-uri, repetoare ș.a.). Acestea sunt capabile să transmită și/sau să recepționeze date generate de alte noduri ale rețelei



Router-ul conectează o rețea la o altă rețea

Switch-ul conectează dispozitivele din aceeași rețea

Modem-ul (modulator-demodulator) modifică forma datelor transmise (din format analogic în digital și invers)

În cadrul rețelei dispozitivele sunt conectate, folosind medii de transmisie cu fir (cabluri) sau fără fir (aerul).

Când conectăm două calculatoare acasă, folosind un router plug-and-play, realizăm o rețea, deși foarte mică.

Criteriile pe care trebuie să le satisfacă o rețea

Performanța rețelei poate fi măsurată în mai multe moduri, inclusiv prin timpul de transfer și timpul de răspuns.

Timpul de transfer este perioada necesară pentru ca un mesaj să ajungă de la un dispozitiv la altul.

Timpul de răspuns este timpul scurs de la transmiterea unei cereri până la obținerea unui răspuns.

Performanța unei rețele depinde de numărul de utilizatori, tipul de mediu de transmisie, capacitățile hardware-ului conectat și eficiența software-ului utilizat.

Performanța este adesea evaluată prin două metrice de rețea:

- rata de transfer (throughput)
- întârziere (delay).

De multe ori avem nevoie de o rată de transfer mai mare și mai puțină întârziere. Totuși, aceste două criterii pot fi contradictorii. Dacă încercăm să trimitem mai multe date în rețea, este posibil să creștem rata de transfer, dar creștem și întârzierea din cauza congestionării (aglomerării) traficului în rețea.

Fiabilitate

Pe lângă acuratețea livrării, fiabilitatea rețelei este măsurată prin frecvența defecțiunilor și prin timpul necesar pentru a recupera o legătură în urma unei defecțiuni.

Securitate

Problemele de securitate a rețelei urmăresc protejarea datelor împotriva accesului neautorizat, împotriva modificării acestora

Este necesară implementarea unor proceduri de recuperare după comiterea unor fraude sau pierderea datelor.

Tipul de conexiune

Rețeaua – două sau mai multe dispozitive conectate prin intermediul link-urilor

Link-ul este o structură de comunicații care transferă date de la un dispozitiv la altul

Geometric ne vom imagina un link ca o linie trasată între două puncte (noduri)

Dacă două dispozitive sunt conectate la un link, acestea pot să comunice

Există două tipuri posibile de conexiuni: punct la punct (point-to-point) și multipunct (multipoint).

Punct la punct

O conexiune punct la punct furnizează o link dedicat între două dispozitive. Întreaga capacitate a conexiunii este rezervată pentru transmiterea datelor între cele două dispozitive.

Când schimbăm canalele de televiziune prin telecomandă cu infraroșu, stabilim o conexiune punct la punct între telecomandă și sistemul de control al televizorului.

Multipoint

O conexiune multipoint este una în care mai mult de două dispozitive partajează resursele unui singur link (vezi Figura 1.3b).

Figure 1.3 *Types of connections: point-to-point and multipoint*

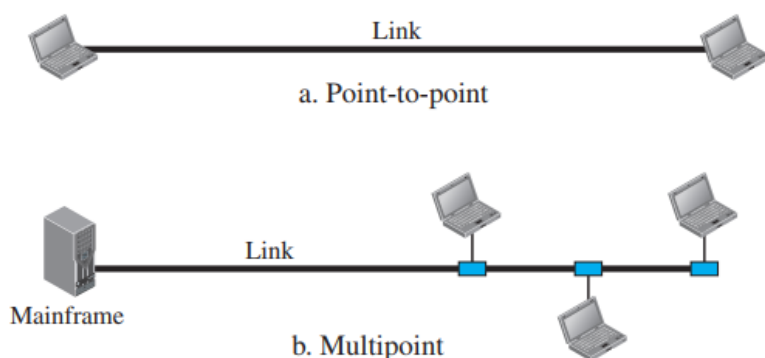
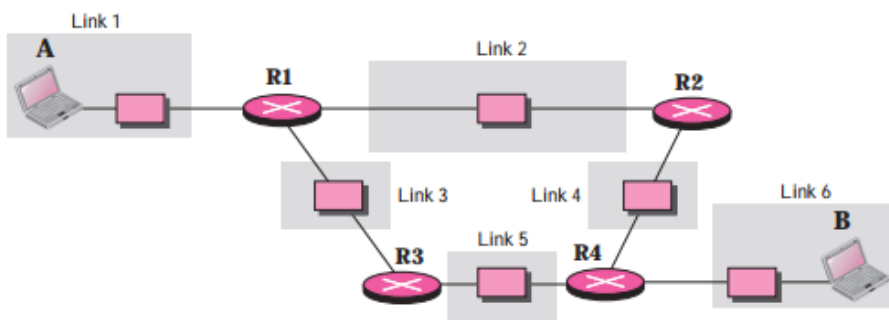


Figure 2.9 *A private internet*



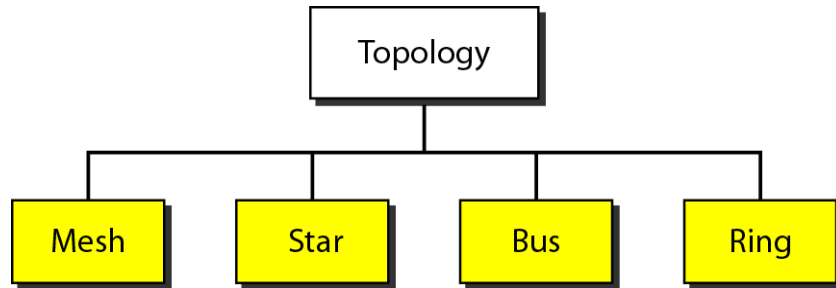
Topologia fizică

Termenul de topologie fizică se referă la structura geometrică pe care o constituie componentele rețelei.

Două sau mai multe dispozitive se conectează la un link

Două sau mai multe link-uri formează o topologie

Topologia unei rețele este descrisă de relația geometrică dintre link-uri și dispozitivele de legătură (numite noduri). Există patru topologii de bază: plasă (mesh), stea (star), magistrală (bus) și inel (ring).



Topologia plasă (mesh)

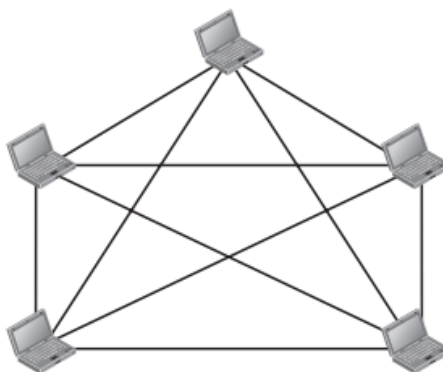
Într-o topologie de tip plasă, fiecare dispozitiv are o conexiune punct la punct dedicată cu orice alt nod al rețelei. Termenul dedicat înseamnă că link-ul transportă trafic doar între cele două dispozitive pe care le conectează.

Pentru a găsi numărul de conexiuni fizice într-o rețea de tip mesh completă (aceasta înseamnă că între oricare două noduri este asigurată o conexiune full duplex), ce conectează n noduri, se va ține cont că fiecare nod este conectat la fiecare din celelalte noduri. Nodul 1 este conectat la $n - 1$ noduri, nodul 2 la fel la $n - 1$ noduri, ș.a.m.d. nodul n este conectat la $n - 1$ noduri. Astfel, sunt necesare $n(n - 1)$ conexiuni fizice. Totuși, se va ține cont că fiecare conexiune fizică permite comunicarea în ambele direcții (modul full duplex), și, prin urmare, putem înjumătăți numărul de conexiuni. Cu alte cuvinte, putem spune că într-o topologie de tip plasă completă avem nevoie de $n(n - 1) / 2$ link-uri prin care are loc transmisie în modul full duplex.

Pentru a acoperi atât de multe link-uri, fiecare dispozitiv din rețea trebuie să aibă $n-1$ porturi de intrare/ieșire (I/O) (a se vedea Figura 1.4), care să fie conectate la celelalte $n-1$ stații.

Figure 1.4 A fully connected mesh topology (five devices)

$n = 5$
10 links.



Un exemplu practic de topologie mesh este conectarea birourilor regionale de telefonie, în care fiecare birou regional trebuie să fie conectat la orice alt birou regional.

Topologia mesh oferă unele **avantaje** față de alte topologii de rețea.

- utilizarea de link-uri dedicate garantează că fiecare conexiune își poate transporta propria sarcină de date, eliminând astfel problemele de trafic care pot apărea atunci când link-urile trebuie să fie partajate de mai multe dispozitive.
- din puncte de vedere a fiabilității, topologia mesh este suficient de robustă. Dacă o legătură devine inutilizabilă, aceasta nu scoate din funcțiune întregul sistem.
- este asigurat un anumit nivel de securitate. Când fiecare mesaj este transmis de-a lungul unei linii dedicate, doar destinatarul real îl vede. Restricțiile fizice împiedică alți utilizatori să acceseze mesajele transmise.
- link-urile punct la punct fac ușoară identificarea și izolarea defectelor. Traficul poate fi dirijat pentru a evita link-urile cu anumite probleme. Această facilitate permite managerului de rețea să descopere locația precisă a defectului și ajută la găsirea cauzei și a soluției corespunzătoare.

Principalele **dezavantaje** ale unei *rețele de tip plasă* țin de cantitatea necesară de cablu și numărul de porturi de I/O necesare.

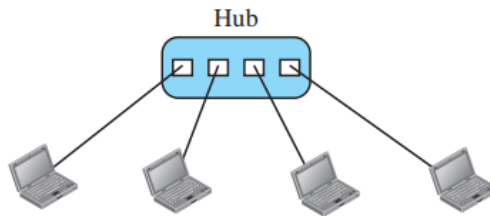
- Deoarece fiecare dispozitiv trebuie conectat cu toate celelalte, instalarea și reconectarea este anevoioasă.
- Cea mai mare parte a cablurilor poate fi mai mare decât spațiul disponibil (în pereți, tavane sau podele).
- Hardware-ul necesar pentru conectarea fiecărui link (porturi I/O și cablu) poate fi costisitor.

Din aceste considerente, de obicei o topologie plasă este rar utilizată, de exemplu, ca o coloană centrală (se zice backbone) care conectează calculatoarele principale ale unei rețele hibride, care poate include câteva alte topologii.

Topologia stea (star)

Într-o topologie stea, fiecare dispozitiv (nod) are o conexiune punct-la-punct dedicată la un controller central, de tip hub sau switch. Dispozitivele nu sunt legate direct între ele. Spre deosebire de o topologie plasă, o topologie stea nu permite traficul direct între dispozitive. Controller-ul acționează ca un dispozitiv de dirijare a traficului: datele transmise de la un dispozitiv la altul, ajung mai întâi la controller, care le transmite dispozitivului destinatar (a se vedea Figura 1.5).

Figure 1.5 *A star topology connecting four stations*



Topologia de tip stea este folosită în rețelele locale (LAN).

Avantaje ale topologiei stea:

- Din punct de vedere a costului, o topologie stea este mai ieftină decât o topologie plasă. Într-o topologie stea, fiecare dispozitiv are nevoie de o singură conexiune și un singur port I/O pentru a-l conecta la celelalte dispozitive prin intermediul controller-ului. Acest factor contribuie la instalarea și reconfigurarea simplă de realizat a rețelei. Se va lucra cu mai puțin cablu, iar adausurile, mutările și eliminările implică o singură conexiune: între acel dispozitiv și controller.
- Robustețea - dacă o conexiune este întreruptă, numai acea conexiune este afectată. Toate celelalte conexiuni rămân active. Acest factor contribuie la identificarea ușoară a defectelor. Atâta timp cât controller-ul funcționează, acesta poate fi utilizat pentru a monitoriza problemele de conexiune și a ocoli link-urile defecte.

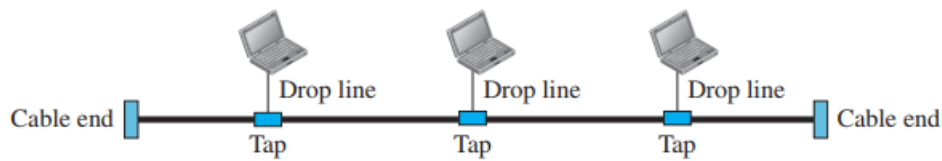
Dezavantaje:

- dependența întregii topologii de un singur punct - controller-ul. Dacă acesta se defectează, atunci este considerat defect întregul sistem de comunicare. Deși o topologie stea necesită mai puțin cablu decât o topologie de tip plasă, fiecare nod trebuie conectat la un controller. Din acest motiv, de multe ori este nevoie de mai mult cablu într-o topologie stea decât în topologiile de tip inel sau magistrală.

Topologia magistrală (bus)

Topologiile plasă și stea constau din conexiuni punct la punct. Spre deosebire de acestea, topologia magistrală formează conexiuni multipoint. Un cablu suficient de lung acționează ca o coloană centrală (backbone) ce conectează la rețea toate dispozitivele (a se vedea Figura 1.6).

Figure 1.6 A bus topology connecting three stations



Nodurile sunt conectate la cablul central prin linii de cădere și robinete. Linia de cădere conectează un dispozitiv la cablul central. Un robinet este un conector la cablul central, care realizează un contact cu miezul metalic al cablului.

Pe măsură ce un semnal circulă de-a lungul cablului central (backbone), o parte din energia sa este transformată în căldură. Prin urmare, semnalul devine din ce în ce mai slab, pe măsură ce se deplasează. Din acest motiv, există o limită impusă față de numărul de robinete pe care le poate suporta o topologie magistrală și față de distanța dintre aceste robinete.

Topologia de tip magistrală a fost una dintre primele topologii utilizate în proiectarea rețelelor locale.

Avantajele unei topologii magistrală:

- Ușurința instalării.
- Cablul central poate fi amplasat pe calea cea mai eficientă, apoi este conectat la noduri prin linii de cădere de diferite lungimi. În acest fel, o rețea cu o topologie magistrală folosește mai puțin cablu decât topologiile plasă sau stea. Într-o topologie stea, de exemplu, patru dispozitive de rețea din aceeași cameră necesită patru lungimi de cablu care ajung până la switch. Într-o topologie magistrală, această redundanță este eliminată. Doar cablul central se întinde prin întreaga instalație. Fiecare linie de cădere trebuie să ajungă doar până la cel mai apropiat punct al coloanei centrale.

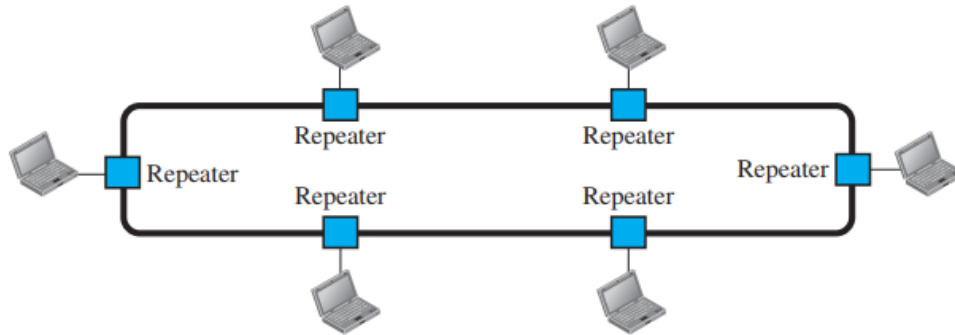
Dezavantajele unei topologii magistrală:

- Reconectarea dificilă și izolarea defectelor. O rețea cu topologie magistrală este de obicei proiectată pentru a fi optimal de eficientă la instalare. Prin urmare, poate fi dificil să adăugați dispozitive noi. Reflectarea semnalului la robinete poate provoca degradarea calității. Această degradare poate fi controlată prin limitarea numărului robinetelor și distanțării dispozitivelor conectate la o anumită lungime de cablu. Prin urmare, adăugarea de dispozitive noi poate necesita modificarea sau înlocuirea coloanei centrale.
- O defecțiune sau o întrerupere a cablului rețelei oprește orice transmisie, chiar și între dispozitivele din aceeași parte a problemei. Zona deteriorată reflectă semnalele înapoi în direcția de origine, creând zgomot în ambele direcții.

Topologia inel (ring)

Într-o topologie inel, fiecare dispozitiv are o conexiune punct la punct dedicată cu cele două dispozitive de o parte și de alta a acestuia. Un semnal este trecut de-a lungul inelului într-o direcție, de la dispozitiv la dispozitiv, până când ajunge la destinație. Fiecare dispozitiv al inelului include un *repetor*. Când un dispozitiv primește un semnal destinat unui alt dispozitiv, repetorul său regenerează semnalul și îl transmite mai departe (a se vedea Figura 1.7).

Figure 1.7 A ring topology connecting six stations



Topologia inel a predominat atunci când IBM a propus rețeaua sa locală, Token Ring. Astăzi, nevoia de LAN-uri cu viteză mai mare a făcut ca această topologie practic să dispară.

Avantajele topologiei inel:

- Un inel este relativ ușor de instalat și reconfigurat. Fiecare dispozitiv este legat doar de vecinii săi nemijlociți (atât fizic cât și logic). Pentru a adăuga sau a elimina un dispozitiv, trebuie de schimbat doar două conexiuni. Singurele constrângeri sunt considerațiile de trafic (lungimea maximă a inelului și numărul de dispozitive).
- Izolarea defectelor este simplificată. În general, într-un inel, un semnal circulă în permanență. Dacă un dispozitiv nu primește un semnal într-o perioadă specificată, acesta poate emite o alarmă. Alarma avertizează operatorul de rețea despre problemă și locația sa.

Dezavantajele topologiei inel:

- Traficul unidirecțional poate fi un dezavantaj. Defectarea unui dispozitiv din rețeaua cu topologie inel dezactivează întreaga rețea.

Tipuri de rețele

Criterii pentru a delimita tipurile de rețele: dimensiunea, acoperirea geografică, proprietarul.

Tipuri de rețele: LAN și WAN (uneori se mai examinează separat și MAN)

Comutarea – procedeu utilizat la conectarea rețelelor pentru a forma un internetwork (rețea de rețele)

LAN – Local Area Network

O rețea (deseori privată) de dimensiuni reduse ce conectează dispozitivele unui office, unei clădiri sau ale unui campus

Poate să fie formată acasă din două calculatoare și un printer, dar poate să fie de proporțiile unei companii și să includă audio și video device-uri

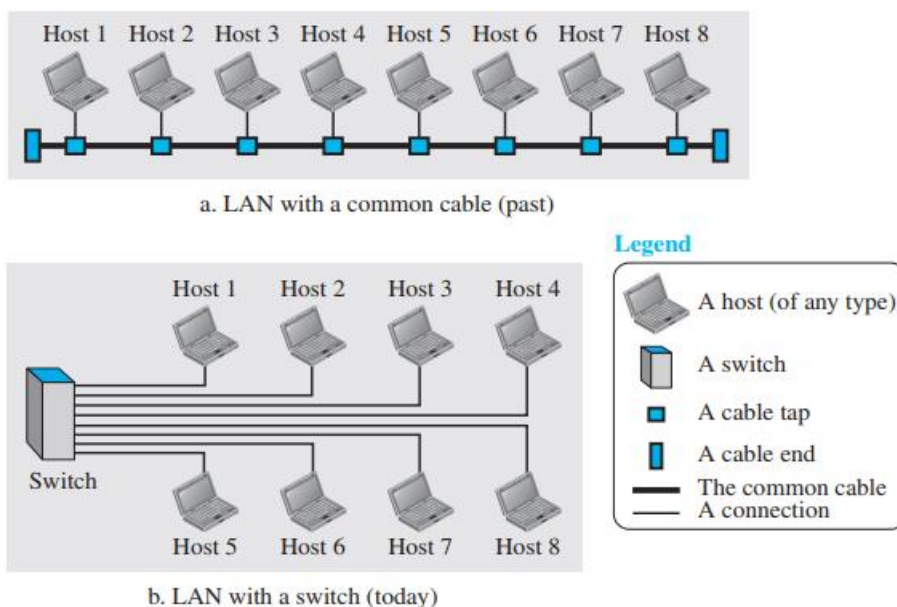
Fiecare host din cadrul LAN-ului are un identificator, o adresă care definește în mod unic host-ul în acest LAN. Un pachet de date ce este transmis între două host-uri, include adresele sursei și a destinației

În trecut toate host-urile rețelei erau conectate printr-un cablu comun (magistrală), ceea ce însemna că pachetul de date transmis de la un host la altul, de fapt, ajungea la toate host-urile rețelei. Destinatarul real prelua pachetul; celelalte dispozitive eliminau pachetul.

Astăzi, majoritatea rețelelor LAN folosesc un comutator de conectare inteligent (numit switch), care este capabil să recunoască adresa de destinație a pachetului și să ghideze pachetul către destinație, fără a-l trimite la toate celelalte host-uri. Comutatorul reduce traficul inutil în cadrul LAN-ului și permite mai multor perechi să comunice între ele în același timp, dacă nu există o sursă și o destinație comună între ele.

Figura 1.8 arată o rețea LAN cu topologie magistrală și încă una bazată pe utilizarea unui comutator (switch).

Figure 1.8 *An isolated LAN in the past and today*



Într-o rețea LAN, de regulă, nu se definește numărul minim sau maxim de host-uri.

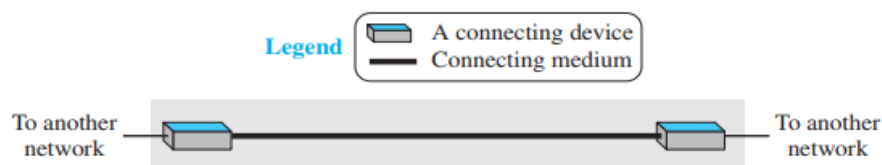
WAN – Wide Area Network

Spre deosebire de LAN care conectează host-uri, WAN-ul reprezintă o rețea ce conectează dispozitivele de rețea, precum switch-urile, routerele și modemele, localizate la o scară geografică mai mare, cum ar fi, un oraș, o țară sau chiar continente.

Dacă LAN-urile aparțin unor companii private, WAN-urile aparțin companiilor de comunicații. LAN-urile sunt interconectate între ele, dar și la rețelele WAN, oferind posibilități de comunicare la nivel mai înalt.

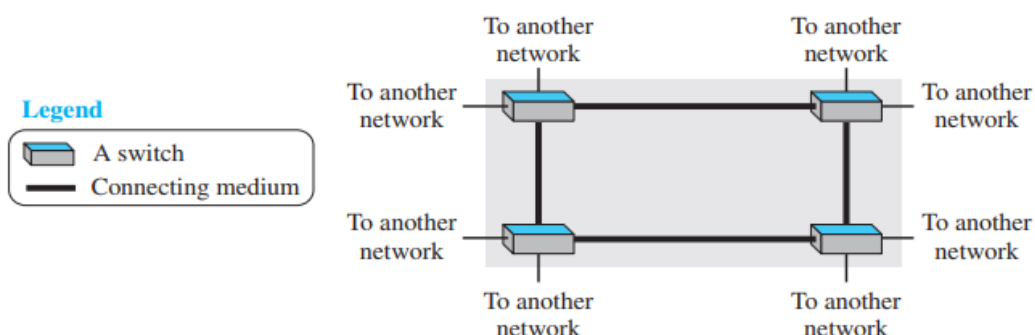
Actualmente, există două tipuri de rețele WAN: punct la punct (point-to-point) și comutate (switched). Rețelele WAN punct la punct conectează prin mediul de transmisie (cablu sau aer) două dispozitive de comunicație din diferite rețele. În Figura 1.9 este ilustrat un exemplu de rețea WAN punct la punct.

Figure 1.9 *A point-to-point WAN*



Putem spune că o rețea WAN comutată este o combinație a mai multor WAN-uri punct la punct care sunt conectate prin comutatoare. O rețea WAN comutată este folosită astăzi în coloana centrală (backbone) a comunicării globale. Figura 1.10 prezintă un exemplu de WAN comutat

Figure 1.10 *A switched WAN*



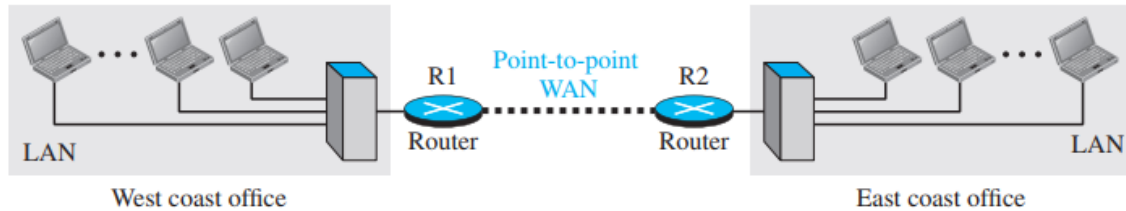
Rețea de rețele Internetwork

Internetwork – conexiune a două sau mai multe rețele astfel încât acestea să poată comunica între ele

Exemplu de internetwork:

Două office-uri ale unei organizații sunt amplasate în regiuni diferite. Fiecare office are LAN-ul său, dar acestea pot fi conectate, apelând la serviciile unui Service Provider (de exemplu, o companie de telefonie), printr-un WAN punct la punct (a se vedea Figura 1.11).

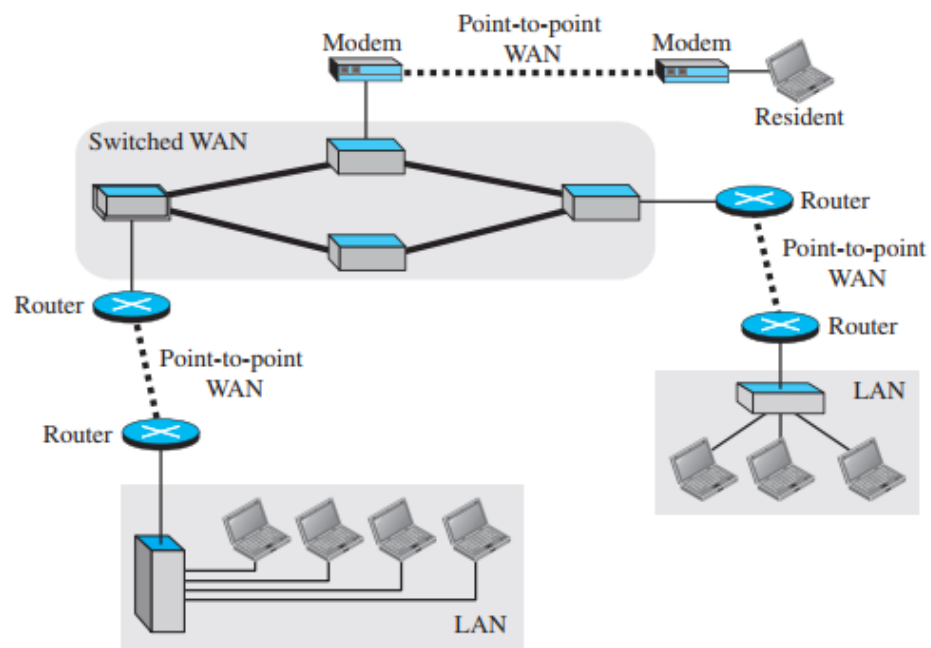
Figure 1.11 *An internetwork made of two LANs and one point-to-point WAN*



Când un host care se află în office-ul din west transmite un mesaj către un host din același office, routerul R1 blochează trecerea mesajului, iar switch-ul direcționează mesajul la destinație. Atunci când host-ul din west transmite un mesaj unui host din east, routerul R1 transmite pachetul către routerul R2, iar switch-ul din acea rețea îl transmite la destinație.

În Figura 1.12 este ilustrat un alt internetwork cu mai multe LAN-uri și WAN-uri conectate. Unul din aceste WAN-uri este un WAN comutat (switched) cu 4 switch-uri.

Figure 1.12 *A heterogeneous network made of four WANs and three LANs*



Comutarea (switching)

Switch-ul de nivelul trei sau routerul trebuie să transmită date dintr-o rețea către o altă rețea.

Două tipuri de rețele comutate: rețele cu **comutare de circuit** (circuit-switched – a se vedea Figura 1.13) și rețele cu **comutare de pachete** (packet-switched).

Într-o rețea de calculatoare, comunicarea dintre două host-uri se face prin blocuri de date numite pachete. Cu alte cuvinte, în loc de comunicarea continuă cum este între două telefoane, între două calculatoare are loc schimbul de pachete de date. Comutatoarele pot să stocheze pachetele de date și să le transmită ulterior. Figura 1.14 prezintă o mică rețea cu comutație de pachete, care conectează patru calculatoare ale unei rețele la patru calculatoare ale altei rețele.

Figure 1.13 A circuit-switched network

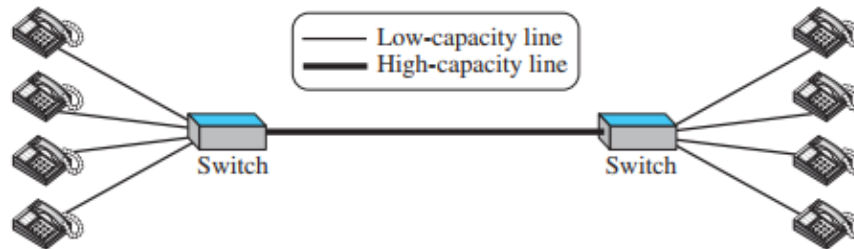
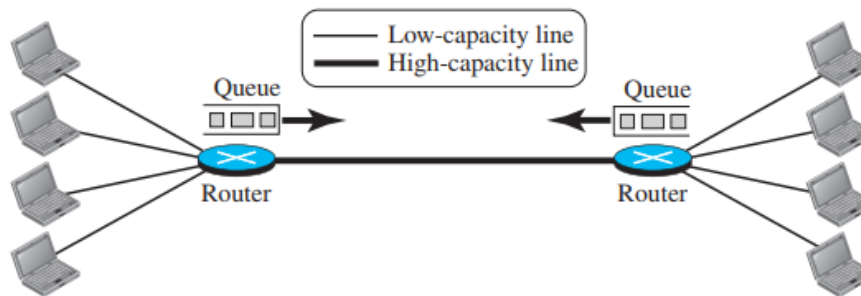


Figure 1.14 A packet-switched network



Routerul din rețeaua cu comutație de pachete poate stoca și returna pachete.

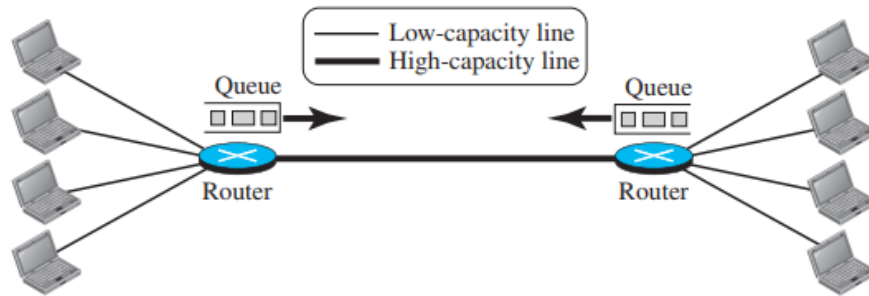
Argumentare:

Fie capacitatea canalului principal al rețelei din Figura 1.14 de două ori mai mare decât cea a fiecărui canal ce conectează calculatoarele la router. Dacă doar două calculatoare din cele patru (câte unul din fiecare rețea) comunică între ele, nu există nicio întârziere în recepționarea pachetelor.

Totuși, dacă pachetele ajung la router atunci când canalul principal funcționează deja la întreaga capacitate, aceste pachete sunt stocate și trimise când canalul are suficient loc liber, în ordinea în care au ajuns la router.

În general, rețeaua cu comutație de pachete este mai eficientă decât rețeaua cu comutare de circuit, dar transmiterea pachetelor de la router la destinație ar putea să fie realizată cu întârziere.

Figure 1.14 *A packet-switched network*



Internet-ul

Internet-ul este compus din mii de rețele interconectate

În Figura 1.15 este dată o reprezentare conceptuală (nu geografică) a Internetului.

Figure 1.15 *The Internet today*

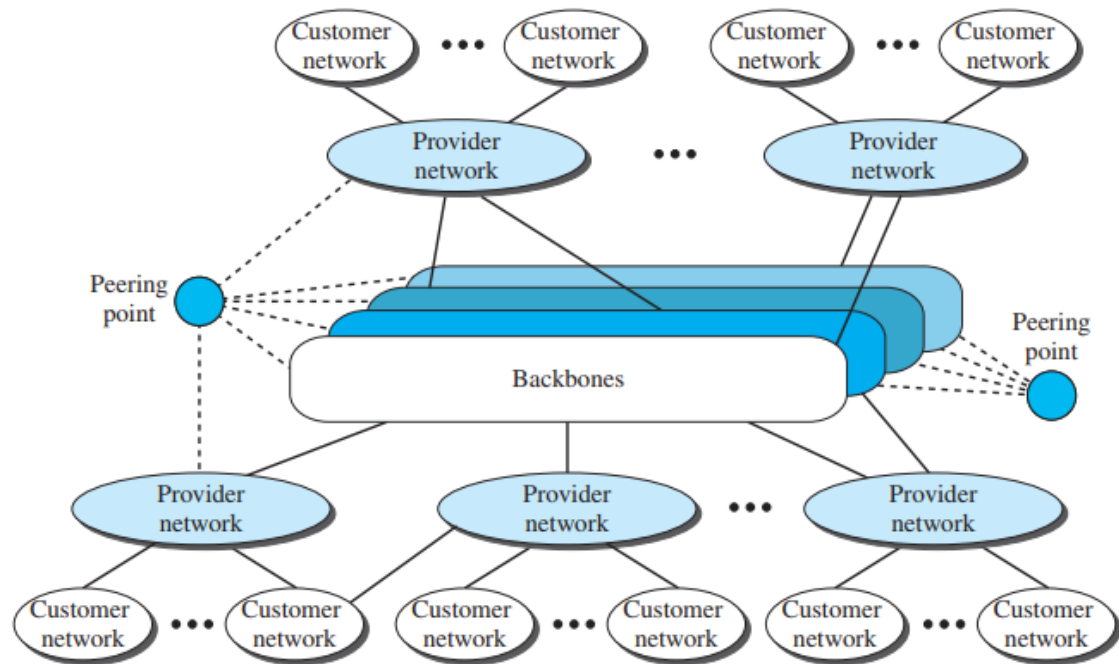


Figura prezintă Internetul ca fiind o combinație de coloane centrale (backbones), rețele de furnizori (provider networks) și rețele de clienți (customer networks).

La nivel superior, coloanele centrale sunt rețele vaste deținute de unele companii de comunicație precum Sprint, Verizon (MCI), AT&T și NTT (companii din SUA). Coloanele centrale sunt conectate prin intermediul unor sisteme de comutare complexe, numite *puncte de peering*. La cel de-al doilea nivel, există rețele mai mici, numite rețele de furnizori, care folosesc serviciile coloanelor centrale contra cost. Rețelele furnizorilor sunt conectate la coloanele centrale și uneori la alte rețele ale furnizorilor. Rețelele de clienți sunt rețele de la frontiera Internetului, care folosesc efectiv serviciile furnizate de Internet. Acestea achită taxe către rețelele de furnizori pentru serviciile recepționate. Coloanele centrale și rețelele de furnizori sunt, de asemenea, numite *Furnizori de Servicii Internet* (ISP). Coloanele centrale sunt adesea denumite ISP-uri internaționale, iar rețelele de furnizori - ISP naționale sau regionale.

Accesarea Internet-ului

Internet-ul reprezintă un internetwork de servicii la care se poate folosi oricine. Pentru aceasta, este necesar ca utilizatorul să fie conectat fizic la un ISP, prin intermediul unui WAN punct la punct.

Ceea ce a făcut ca Internet-ul să fie popular sunt aplicațiile sale, precum www (world wide web) și aplicațiile multimedia (voice over IP (telefonie), video over IP (Skype), partajarea vizualizării (Youtube), television over IP).

a) Conectarea utilizatorului la un ISP se poate face *prin rețelele de telefonie*:

Întrucât majoritatea rețelelor telefonice s-au conectat deja la Internet, o opțiune pentru reședințele și întreprinderile mici să se conecteze la Internet este schimbarea liniei vocale între reședință și centrul telefonic într-un WAN punct la punct. Acest lucru se poate face în două moduri.

- *Serviciu de apelare telefonică (Dial-up service)*. Prima soluție este să adăugați la linia telefonică un modem care transformă datele în voce. Software-ul instalat pe computer apelează ISP-ul și imită realizarea unei conexiuni telefonice. Din păcate, serviciul de apelare este foarte lent, iar atunci când linia este folosită pentru conexiune la Internet, nu poate fi utilizată pentru conexiune telefonică (vocală). Este util doar pentru reședințe mici.
- *Serviciul DSL*. Încă de la apariția Internetului, unele companii de telefonie și-au modernizat liniile telefonice pentru a oferi servicii de internet de mare viteză reședințelor sau întreprinderilor mici. Serviciul DSL permite, de asemenea, utilizarea liniei simultan pentru comunicații de voce și date.

b) Conectarea utilizatorului la un ISP se poate face *prin utilizarea rețelelor de cablu TV*:

În ultimele două decenii mulți utilizatori au început să folosească servicii de cablu TV în loc de antene pentru a primi difuzare TV. Companiile de telecomunicații și-au modernizat rețelele de cablu și s-au conectat la Internet. O reședință sau o întreprindere mică poate fi conectată la Internet prin utilizarea acestui serviciu. Oferă o viteză mai mare, dar aceasta variază în funcție de numărul de vecini care folosesc același cablu.

c) Conectarea utilizatorului la un ISP se poate face *prin utilizarea rețelelor wireless*:

Conectarea prin wireless a devenit recent extrem de populară. O gospodărie sau o întreprindere mică pot utiliza o combinație de conexiuni fără fir și cu fir pentru a accesa Internetul. Odată cu accesul wireless WAN în creștere, o gospodărie sau o întreprindere mică poate fi conectată la Internet printr-un WAN wireless.

d) Conectarea utilizatorului la un ISP se poate face *prin conexiune directă la Internet*

O organizație mare sau o corporație mare poate deveni ea însăși un ISP local și să fie conectată la Internet. Acest lucru se poate face dacă organizația închiriază un WAN de mare viteză și se conectează la un ISP regional. De exemplu, o universitate mare, cu mai multe campusuri, poate crea un internetwork și apoi poate conecta internetwork-ul la Internet.

Standarde și administrare

În discuția protocoalelor de Internet deseori se face referință la un standard sau la o entitate ce administrează protocolul respectiv.

Unele tipuri de documente:

Internet draft – document fără statut oficial și timp de viață de 6 luni

Înainte ca un standard să fie recomandat de către autoritățile Internet este necesar ca draft-ul să fie publicat ca un *Request for Comment* (RFC).

Standard propus – o specificare bine cunoscută și de interes pentru comunitatea Internet, care este testată și implementată de diferite grupuri

Figure 1.16 *Maturity levels of an RFC*

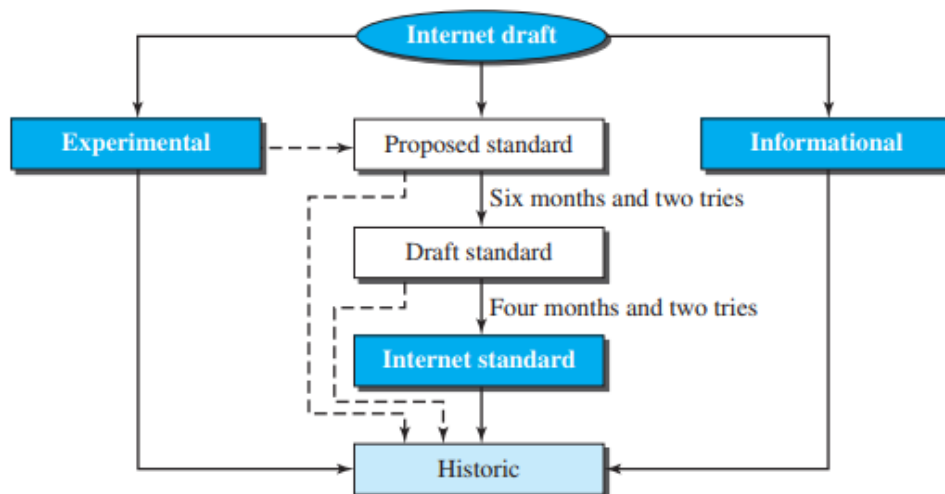
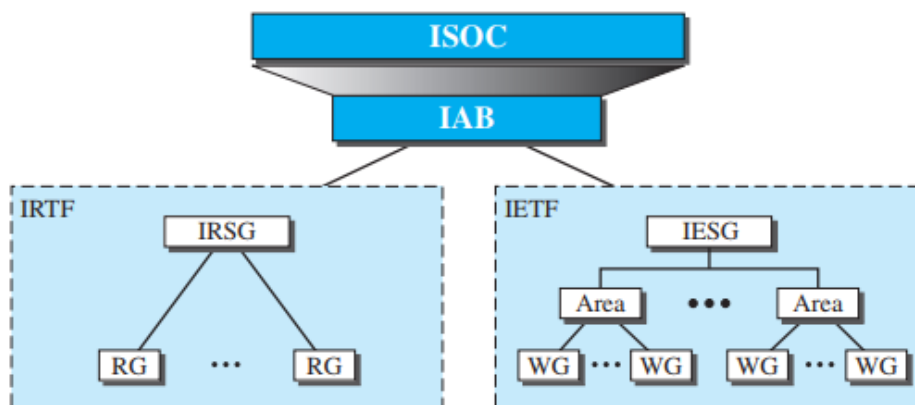


Figure 1.17 *Internet administration*



2. Modele de arhitectură de rețea

- Se introduce conceptul de arhitectură a rețelei pe nivele de protocoale, folosind două scenarii
- Se discută cele două principii pe care se bazează stiva de protocoale
- Se discută conexiunea logică între două nivele identice ale stivei de protocoale (de la sursă și destinație respectiv)
- Se face o comparație între conexiunile logice și cele fizice
- Se discută cele cinci nivele ale stivei de protocoale TCP/IP
- Se prezintă denumirile pachetelor de date transmise la fiecare nivel și se descrie mecanismul de adresare de la fiecare nivel
- Se discută modelul de referință OSI și se compară acesta cu modelul TCP/IP

Stiva *protocoalelor de rețea*

În comunicațiile prin rețelele de calculatoare protocolul definește regulile pe care urmează să le respecte expeditorul și destinatarul mesajelor (dar și dispozitivele intermediare prin care se transmit mesajele) pentru o comunicare eficientă.

Protocoalele controlează toate aspectele comunicațiilor de date, incluzând:

- Care sunt echipamentele utilizate pentru a construi rețeaua?
- Cum sunt conectate între ele dispozitivele din rețea?
- Cum sunt formate datele pentru transmitere?
- Cum sunt transmise datele?
- Ce se întâmplă când apar erori și cum se pot corecta erorile?

Când comunicarea este una simplă, este necesar doar un protocol de comunicații simplu, însă dacă comunicarea este una complexă, sunt necesare mai multe protocoale relativ simple, care interacționează și care în totalitatea lor permit o comunicare eficientă. Problema de asigurare a unei comunicări eficiente este divizată în subprobleme mai simple, interdependente. La fiecare nivel este soluționat un set de subprobleme în conformitate cu unele protocoale. Astfel, comunicarea în rețelele moderne este bazată pe o structură ierarhică (de tip stivă) de protocoale de rețea. Ierarhizarea protocoalelor are ca obiectiv simplificarea proiectării rețelelor de calculatoare.

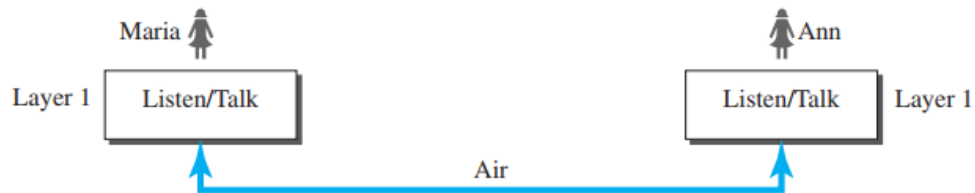
Protocoalele de comunicații sunt clasificate după nivelul la care operează, cele mai cunoscute ierarhii de protocoale fiind *modelul OSI* și *modelul TCP/IP*. Pentru ca o comunicație să transmită un mesaj, este necesară folosirea de protocoale de la toate nivelele modelului TCP/IP.

Scenarii ce argumentează necesitatea ierarhizării protocoalelor de rețea

Scenariul I

Comunicarea este atât de simplă, încât poate fi reprezentată într-un singur nivel. Presupunem că Maria și Ana sunt două prietene vecine. Acestea deseori comunică între ele. Comunicarea dintre Maria și Ana poate fi descrisă, folosind un singur nivel de complexitate, adică discuțiile se poartă față în față în aceeași limbă (a se vedea ilustrarea din Figura 2.1).

Figure 2.1 *A single-layer protocol*



Chiar și în acest scenariu simplu de comunicare trebuie respectate o serie de reguli:

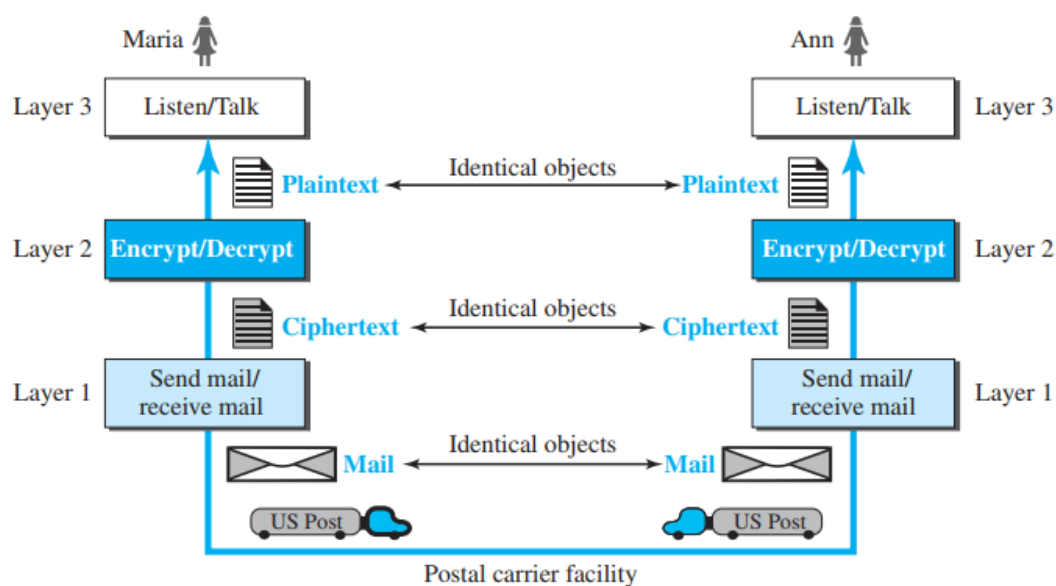
- Maria și Ana știu că trebuie să se salute reciproc atunci când se întâlnesc.
- Maria și Ana știu că trebuie să discute lucruri de care ambele sunt interesate.
- Maria și Ana știu că trebuie să asculte atunci când prietena vorbește.
- Maria și Ana știu că conversația trebuie să fie un dialog, nu un monolog; adică ambele prietene trebuie să aibă posibilitate să vorbească referitor la chestiunea discutată.
- Maria și Ana știu că trebuie să facă schimb de expresii de politețe atunci când se despart.

Putem vedea că protocolul folosit de Maria și Ana este diferit de comunicarea dintre un profesor și studenții săi dintr-o sală de curs. Comunicarea în cel de-al doilea caz este în mare parte un monolog; profesorul vorbește de cele mai multe ori, cu excepția cazului în care un student are o întrebare, situație în care protocolul prevede că studentul ar trebui să ridice mâna și să aștepte permisiunea de a vorbi. În acest caz, comunicarea este una formală și limitată la materia predată.

Scenariul II

În cel de-al doilea scenariu, presupunem că Anei i se oferă o poziție de nivel superior în compania ei, dar trebuie să se mute la o altă sucursală, situată într-un oraș îndepărtat de cel al Mariei. Cele două prietene continuă comunicarea, transmițând scrisori prin poștă. Totuși, Maria și Ana nu doresc ca cele relatate între ele să fie văzut și înțeles de alte persoane, dacă scrisorile sunt interceptate. Ele s-au înțeles să creeze/decreteze mesajele transmise. Expeditorul creezează conținutul scrisorii pentru a ascunde înțelesul acesteia în fața unui intrus; destinatarul scrisorii o decretează pentru a obține scrisoarea inițială. Vom admite că Maria și Ana folosesc același algoritm de criptare, care asigură imposibilitatea decriptării conținutului scrisorii dacă nu se știe cheia secretă pentru criptare. Acum putem spune că comunicarea dintre Maria și Ana are loc pe trei nivele, așa cum se arată în Figura 2.2.

Figure 2.2 A three-layer protocol



Presupunem că Ana și Maria au fiecare câte trei dispozitive (sau roboți) care pot îndeplini sarcina de la fiecare nivel.

Atunci când Maria vrea să îi trimită Anei o scrisoare, aceasta dictează textul scrisorii dispozitivului de nivelul trei. Dispozitivul de nivelul trei înregistrează textul Mariei (text în engleză pe care îl vom numi text clar), care este transmis dispozitivului de nivelul doi, care la rândul lui criptează textul clar, generând astfel textul criptat, pe care îl transmite dispozitivului de nivelul I. Dispozitivul de nivelul I, preia textul criptat, îl introduce într-un plic, adaugă adresele expeditorului și ale destinatarului și trimite plicul prin poștă. Dispozitivul de nivelul I ce aparține Anei preia scrisoarea din căsuța poștală a lui Ana și verifică adresa expeditorului, astfel aflând că scrisoarea este de la Maria. Dispozitivul extrage din plic textul criptat și îl livrează dispozitivului de nivelul doi, care, la rândul lui, decriptează mesajul, obținând textul clar, pe care îl transmite dispozitivului de nivelul trei. Dispozitivul de nivelul trei citește Anei textul clar recepționat de la Maria.

La problema formulată a fost aplicată metoda de decompoziție, adică aceasta a fost reprezentată ca trei subprobleme mai simple, cu protocoale mai simple asociate fiecărei subprobleme. Desigur, există posibilitatea de soluționat problema formulată în baza unui singur dispozitiv. Totuși, trebuie să se țină cont că dacă Maria și Ana constată că criptarea / decriptarea cu algoritmul definit de dispozitivul de la nivelul doi nu le mai asigură confidențialitatea datelor transmise, acestea ar trebui să schimbe dispozitivul respectiv (care va aplica un alt algoritm de criptare). Dacă problema este descompusă în trei subprobleme (nivele) la fiecare din care acționează respectivul dispozitiv, atunci se va înlocui doar dispozitivul de nivelul doi, celelalte două dispozitive rămânând intacte.

În caz contrar, când se folosește un singur dispozitiv pentru soluționarea problemei inițiale, acesta va trebui să fie înlocuit, ceea ce ar putea să implice costuri esențial mai mari și, eventual, alte dificultăți.

Ana și Maria pot procura dispozitivul de nivelul doi de la producători diferiți, cu condiția ca acesta să funcționeze la fel (adică să folosească același algoritm de criptare). Atât timp cât cele două dispozitive creează din textul clar același text criptat și invers, acestea sunt acceptabile.

Astfel poate fi argumentată necesitatea ierarhizării protocoalelor de rețea, menite să soluționeze anumite probleme ale rețelei construite.

Unul dintre avantajele ierarhizării protocoalelor este că astfel se pot separa serviciile prestate de implementare. Un nivel trebuie să poată recepționa anumite servicii de la nivelul inferior și să furnizeze servicii nivelului superior; nu ne interesează modul în care este implementat nivelul. De exemplu, Maria poate decide să nu procure dispozitivul de la nivelul I, deoarece ea poate realiza singură operațiile necesare. Atâta timp cât Maria poate realiza sarcinile primului nivel, în ambele direcții, sistemul de comunicații funcționează.

Un alt avantaj al ierarhizării protocoalelor, care nu poate fi văzut în exemplele prezentate, este că comunicarea nu întotdeauna are loc direct între două host-uri; există dispozitive intermediare prin intermediul cărora are loc comunicarea, care necesită doar unele nivele, dar nu toate. Dacă nu am folosi stiva de protocoale, ar trebui să facem fiecare dispozitiv intermediar la fel de complex ca și host-urile, ceea ce face ca întregul sistem să coste mai mult.

Există vreun dezavantaj în ceea ce privește utilizarea stivei de protocoale? Desigur când avem un singur nivel proiectarea rețelei este mai simplă. Nu este nevoie ca fiecare nivel să furnizeze servicii nivelului superior și să beneficieze de servicii de la nivelul inferior. De exemplu, Ana și Maria ar putea procura un dispozitiv care ar putea realiza sarcinile de la cele trei nivele. Totuși, după cum am menționat mai sus, dacă într-o zi au descoperit că codul lor a fost spart, fiecare ar trebui să înlocuiască dispozitivul cu unul mai performant, iar aceasta ar putea să implice costuri mari.

Principiile ce stau la baza stivei de protocoale

Principiile ce stau la baza proiectării stivei de protocoale

Principiul I:

Dacă dorim o comunicare bidirecțională, trebuie să realizăm fiecare nivel, astfel încât acesta să poată îndeplini două sarcini opuse, câte una în fiecare direcție de parcurgere a datelor.

În exemplul considerat mai sus, avem:

- sarcina celui de-al treilea nivel este de a asculta (într-o direcție) și a vorbi (în direcție opusă)
- sarcina celui de-al doilea nivel este de a cripta și, respectiv, a decripta
- sarcina primului nivel este să trimită și să recepționeze scrisorile prin poștă

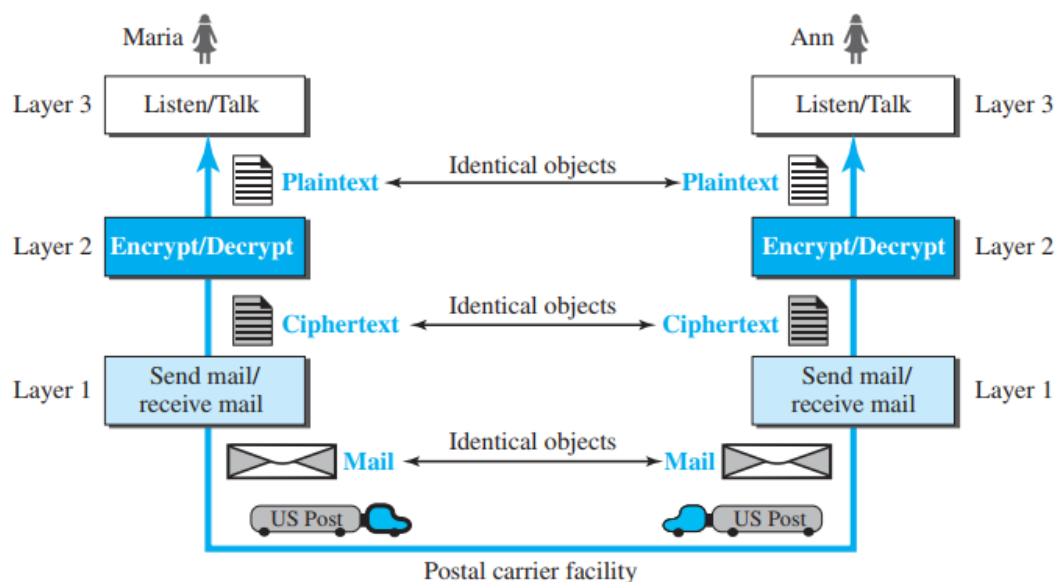
Principiul II:

La fiecare nivel obiectul de la sursă trebuie să fie identic cu obiectul corespunzător de la destinație.

Astfel, în exemplul considerat mai sus, la ambii utilizatori avem:

- obiectul de la nivelul 3 este text clar
- obiectul de la nivelul 2 este un text criptat
- obiectul de la nivelul 1 este un mesaj transmis prin poștă

Figure 2.2 A three-layer protocol

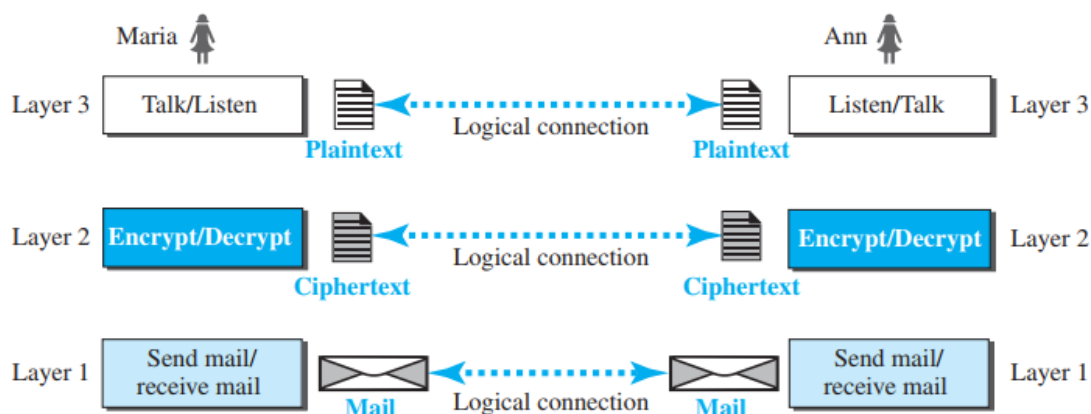


Conexiuni logice

Odată ce sunt respectate principiile de proiectare a stivei de protocoale, ne putem gândi la conexiunea logică între nivelele corespunzătoare ale sursei și destinației, așa cum se arată în Figura 2.3. Aceasta înseamnă că avem o comunicare virtuală între nivele. Conceptul de conexiune logică permite să înțelegem mai bine structurarea pe nivele în comunicarea de date și în rețele

Maria și Ana pot crede că există o conexiune logică (imaginară) între nivelele corespunzătoare, prin care pot trimite obiectul creat în cadrul aceluși nivel.

Figure 2.3 Logical connection between peer layers



Suita de protocoale TCP/IP

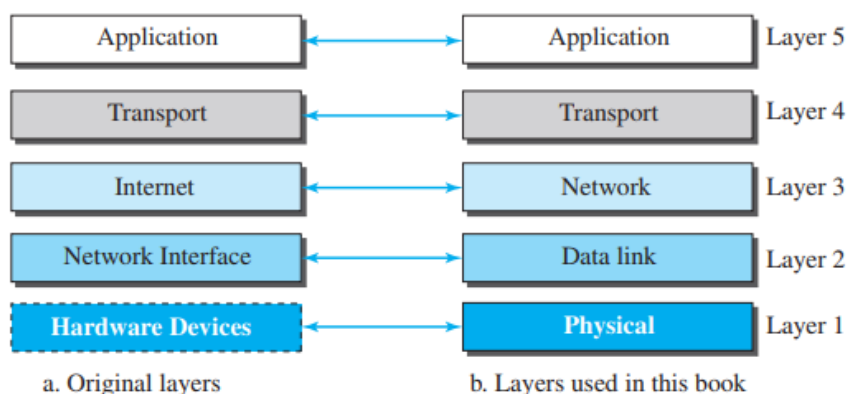
Pentru realizarea comunicării între calculatoarele interconectate în cadrul unei rețele, se utilizează modele de arhitectură de protocoale structurate pe nivele (layers).

Ținând cont că știm despre conceptul de ierarhie a protocoalelor de rețea și despre comunicarea logică între nivele (a se vedea exemplul adus mai sus), putem descrie modelul architectural TCP/IP (Transmission Control Protocol / Internet Protocol).

TCP/IP reprezintă o suită de protocoale (un set de protocoale organizate pe diferite nivele, cu structură de stivă) utilizate actualmente în Internet. Nivelele stivei pot fi considerate ca module interactive, fiecare oferind o funcționalitate specifică. Fiecare protocol de la un anumit nivel beneficiază de serviciile furnizate de unul sau mai multe protocoale de la nivelul inferior.

Suita de protocoale TCP/IP inițial definită, era pe patru nivele plus dispozitivele hardware. Astăzi, însă, suita TCP/IP este considerată ca un model pe cinci nivele. Figura 2.4 prezintă ambele configurații.

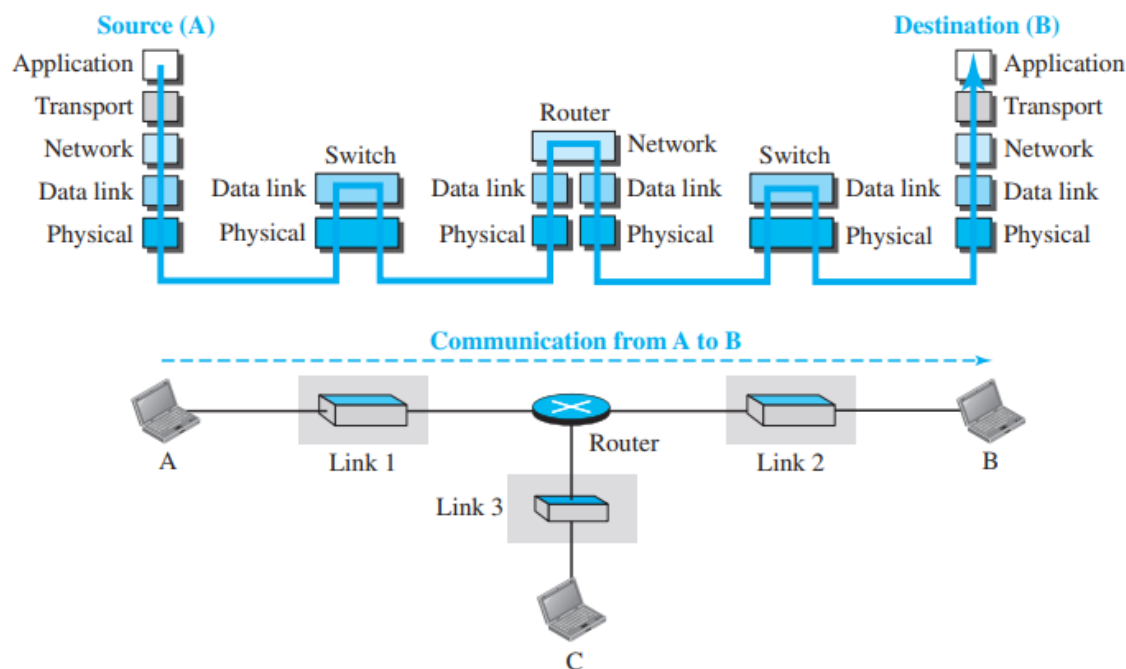
Figure 2.4 *Layers in the TCP/IP protocol suite*



Arhitectura pe nivele

Pentru a arăta modul în care sunt implicate nivelele din suita de protocoale TCP/IP în comunicarea dintre două host-uri, presupunem că avem un internetwork mic format din trei LAN-uri (link-uri), fiecare cu un comutator (switch). De asemenea, presupunem că link-urile sunt conectate la un router, așa cum se arată în Figura 2.5.

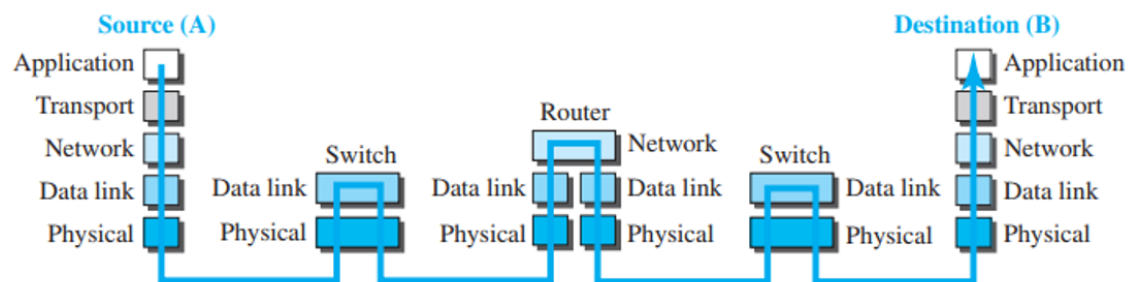
Figure 2.5 *Communication through an internet*



Să admitem că calculatorul A transmite un mesaj către calculatorul B. După cum arată figura, avem cinci dispozitive de comunicare: host-ul sursă (calculatorul A), comutatorul (switch-ul) de la link-ul 1, routerul, comutatorul (switch-ul) de la link-ul 2 și host-ul destinație (calculatorul B).

Fiecare dispozitiv este implicat cu un set de nivele în funcție de rolul dispozitivului în rețeaua examinată. Cele două host-uri sunt implicate în toate cele cinci nivele; host-ul sursă trebuie să genereze un mesaj la nivelul aplicație și să îl trimită spre nivelele de jos, astfel încât acesta să fie trimis fizic host-ului destinație. Host-ul destinație trebuie să primească mesajul la nivelul fizic și apoi să îl livreze prin celelalte nivele către nivelul aplicație.

Figure 2.5 *Communication through an internet*

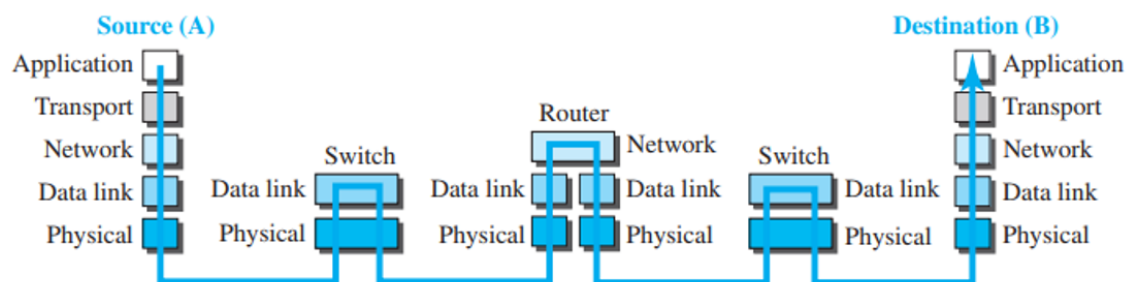


Routerul este implicat doar în trei nivele; nu există nivelul de transport sau de aplicație într-un router, atât timp cât routerul este utilizat numai pentru rutare.

Deși un router este întotdeauna implicat la un nivel rețea, acesta este implicat în n combinații ale nivelelor legătură de date și fizic, unde n este numărul de link-uri la care este conectat routerul. Motivul este că fiecare link poate utiliza propriul protocol de nivel legătură de date sau de nivel fizic.

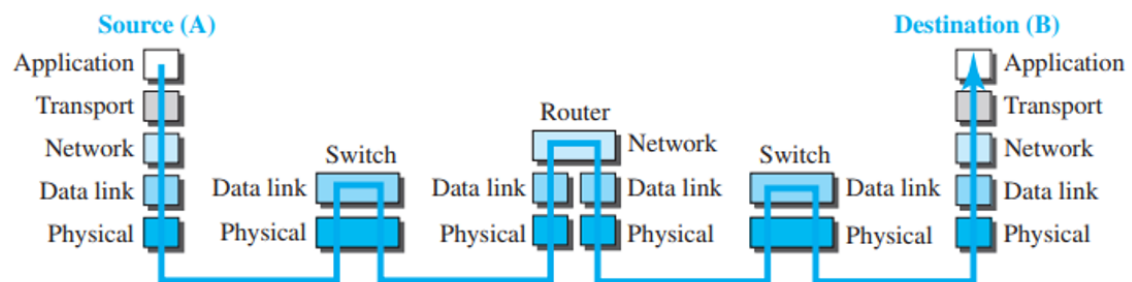
De exemplu, în figura de mai sus, routerul este implicat în trei link-uri, dar mesajul transmis de la sursa A la destinația B este implicat în două link-uri. Fiecare link poate folosi diferite protocoale de nivel legătură de date și de nivel fizic; routerul trebuie să primească un pachet de la link-ul 1 în baza unei perechi de protocoale și să-l livreze la link-ul 2 bazat pe o altă pereche de protocoale.

Figure 2.5 *Communication through an internet*



Un comutator de nivelul 2 (switch) într-un link este implicat doar în două nivele - legătură de date și fizic. Deși fiecare comutator (switch) din figura de mai sus are două conexiuni diferite, conexiunile sunt în același link, care folosește un singur set de protocoale. Aceasta înseamnă că, spre deosebire de un router, un comutator de nivelul 2 (switch) este implicat doar într-un singur nivel legătură de date și un singur nivel fizic.

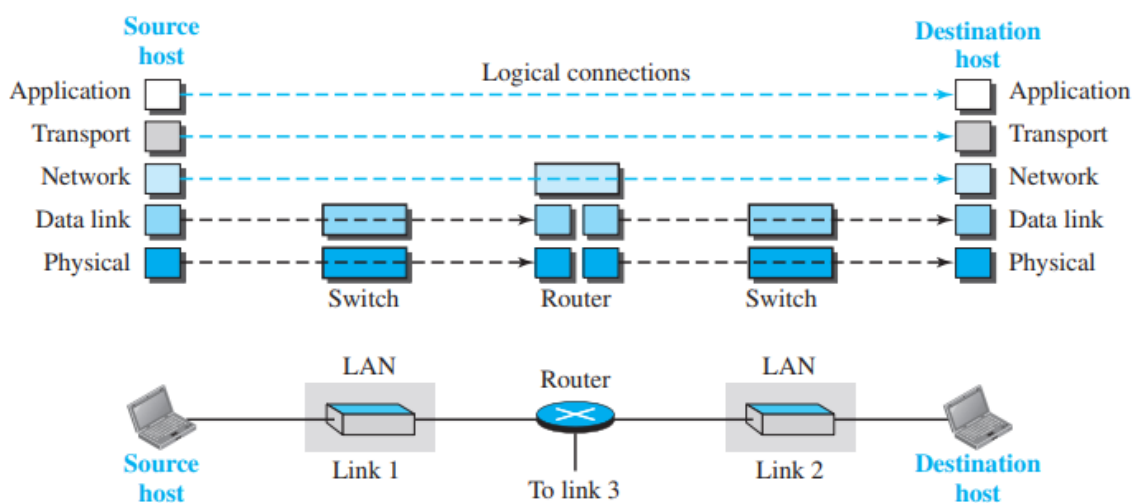
Figure 2.5 *Communication through an internet*



Nivelele suitei de protocoale TCP/IP

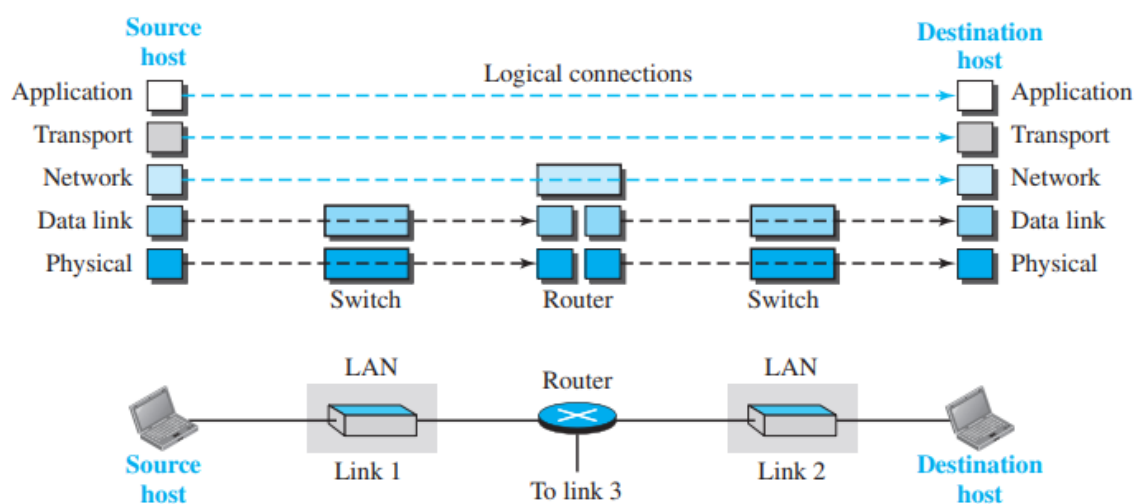
Discutăm pe scurt funcțiile nivelului suitei de protocoale TCP/IP. Pentru a înțelege mai bine sarcinile fiecărui nivel, trebuie să analizăm conexiunile logice dintre nivele. Figura 2.6 prezintă conexiunile logice în internetwork-ul considerat.

Figure 2.6 Logical connections between layers of the TCP/IP protocol suite



Utilizarea conexiunilor logice permite să înțelegem sarcinile fiecărui nivel. După cum arată figura, sarcinile nivelului aplicație, transport și rețea se referă la dispozitivele end-to-end. În schimb, sarcina nivelului legătură de date și cel fizic este de tip hop-to-hop, unde un hop este un host sau un router. Cu alte cuvinte, domeniul de acțiune al celor trei nivele superioare este internetwork-ul, iar domeniul de acțiune al celor două nivele inferioare este link-ul.

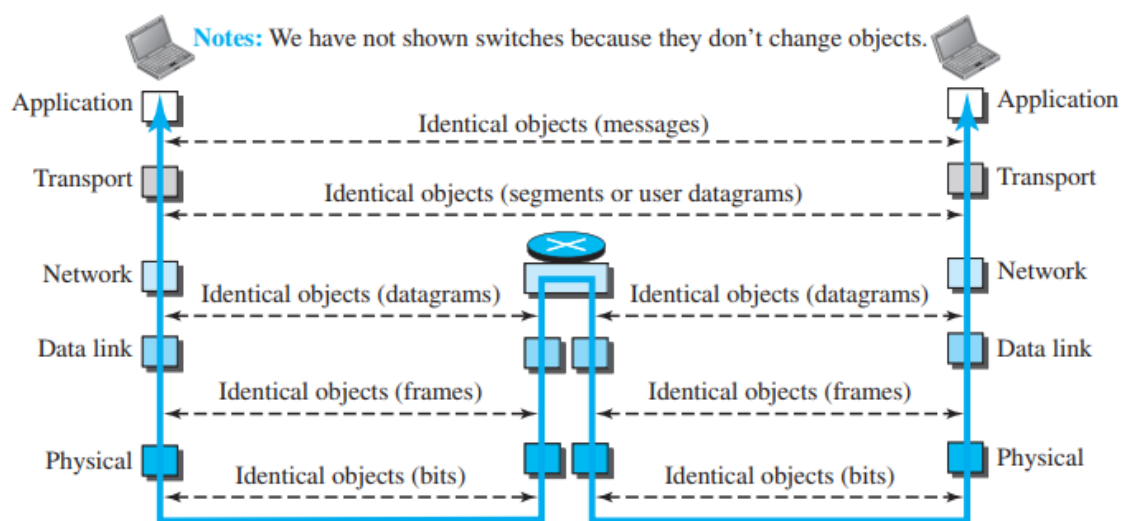
Figure 2.6 *Logical connections between layers of the TCP/IP protocol suite*



O altă aplicare a conexiunilor logice permite analiza unităților de date create la fiecare nivel. În cele trei nivele superioare, unitatea de date (pachetul) nu trebuie să fie modificată de router. În cele două nivele inferioare, pachetul creat de către host este modificat doar de routere, nu și de comutatoarele de nivelul 2 (switch).

Figura 2.7 prezintă al doilea principiu discutat anterior pentru ierarhizarea protocoalelor. Afișăm obiectele identice sub fiecare nivel în corespondență cu fiecare dispozitiv.

Figure 2.7 *Identical objects in the TCP/IP protocol suite*



Rețineți că, deși conexiunea logică la nivelul rețea este între cele două host-uri, putem spune doar că în acest caz există obiecte identice între două hop-uri, deoarece un router poate fragmenta pachetul la nivelul rețea și să trimită mai multe pachete decât cele primite. Rețineți că link-ul dintre două hop-uri nu modifică obiectul.

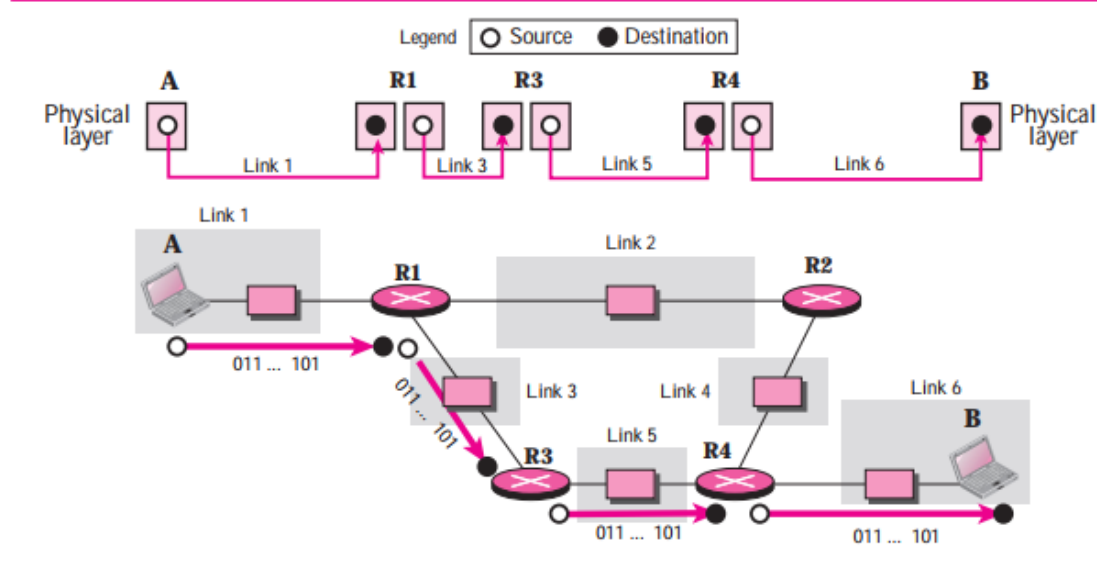
Descrierea fiecărui nivel

Nivelul fizic

Putem spune că nivelul fizic este responsabil de transportarea biților unui cadru (frame) de-a lungul link-ului. Deși nivelul fizic este cel mai de jos nivel din suita de protocoale TCP/IP, comunicarea între două dispozitive la nivelul fizic este încă o comunicare logică deoarece sub nivelul fizic există un alt nivel ascuns, mediul de transmisie. Două dispozitive sunt conectate printr-un mediu de transmisie (cablu sau aer).

Trebuie să știm că mediul de transmisie nu transportă biți; acesta transmite semnale electrice sau optice. Deci biții primiți într-un cadru (frame) de la nivelul legătură de date sunt transformați și trimiși prin mediul de transmisie, dar putem crede că unitatea logică de la nivelul fizic al celor două dispozitive este un bit. Există mai multe protocoale care transformă șirurile de biți în semnal. Le discutăm în Tema dedicată nivelului fizic, atunci când vom analiza mediile de transmisie.

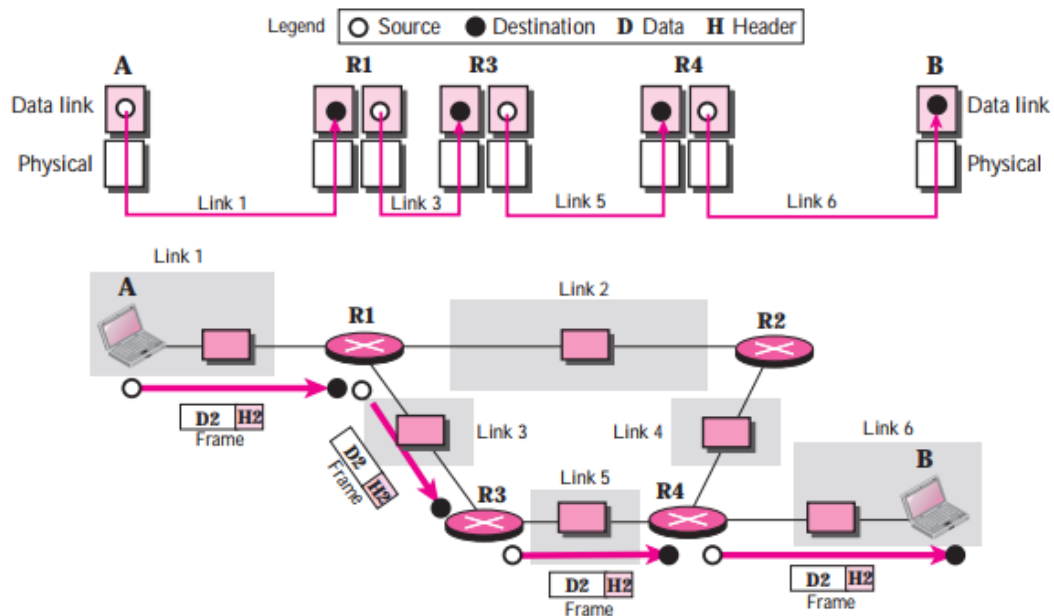
Figure 2.10 Communication at the physical layer



Nivelul legătură de date

Am văzut că un internetwork este format din mai multe link-uri (LAN-uri și WAN-uri) conectate prin routere. Pot exista mai multe link-uri prin care o datagramă poate călători de la un host la destinație. Routerele sunt responsabile pentru alegerea variantei optime de parcurgere a link-urilor. Totuși, atunci când routerul determină următorul link al rutei, nivelul legătură de date este responsabil de preluarea datagrammei și de transferarea acesteia prin acel link. Link-ul poate fi o rețea LAN prin cablu cu un comutator de nivel 2 (switch) sau o rețea LAN wireless. Putem avea diferite protocoale utilizate cu orice tip de link. Oricum, nivelul legătură de date este responsabil pentru transferarea pachetului prin acel link.

Figure 2.11 *Communication at the data link layer*



TCP/IP nu definește protocoale specifice pentru nivelul legătură de date. Modelul acceptă toate protocoalele standard și de tip proprietar. Orice protocol care poate prelua o datagramă și să o transporte printr-un link, este suficient pentru nivelul legătură de date.

Nivelul legătură de date preia o datagramă și o încapsulează într-un pachet numit cadru (frame).

Protocoalele de la nivelul legătură de date pot oferi servicii diferite. Unele protocoale de la acest nivel asigură detectarea și corectarea completă a erorilor, altele realizează numai detectarea erorilor.

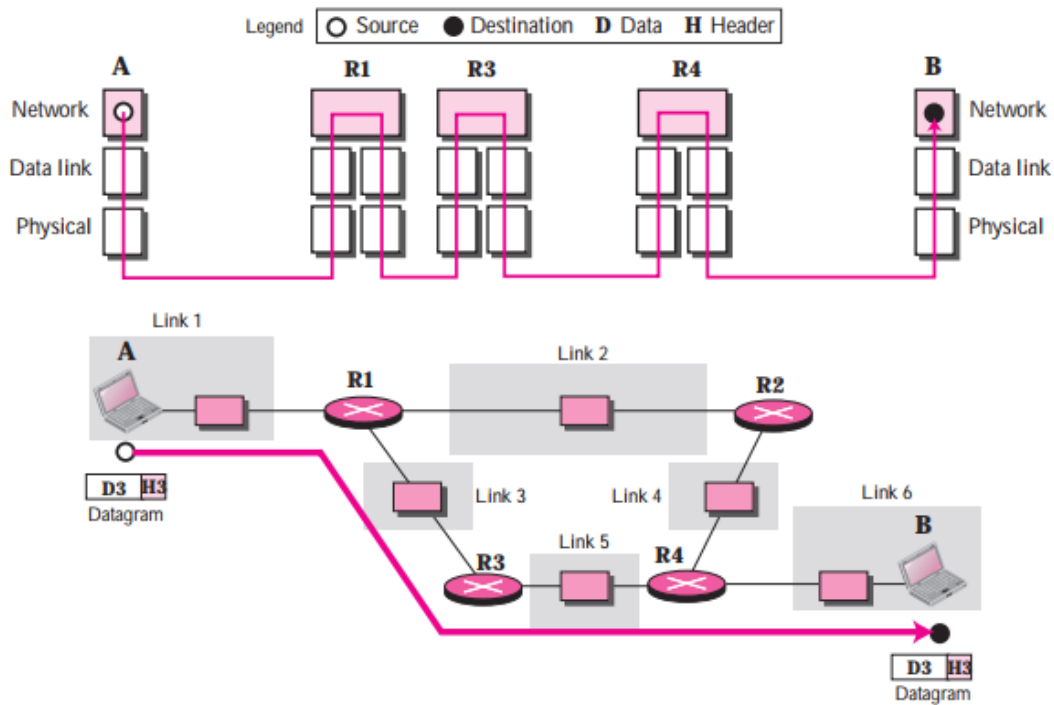
Nivelul rețea

Nivelul rețea este responsabil pentru realizarea unei conexiuni între calculatorul sursă și cel destinație. Comunicarea la nivelul rețea este host-to-host. Din moment ce pot exista mai multe routere în drumul de la sursă la destinație, routerele respective sunt responsabile pentru alegerea celei mai bune rute pentru fiecare pachet. Putem spune că nivelul rețea este responsabil pentru comunicarea host-to-host și dirijarea pachetului prin rutele posibile.

Putem să întrebăm de ce avem nevoie de nivelul rețea? Am fi putut adăuga sarcina de rutare la nivelul de transport și să renunțăm la acest nivel. Iată două argumente:

- Separarea funcțiilor distincte pe diferite nivele
- *Routerele nu au nevoie de nivelul aplicație și cel de transport.* Divizarea funcțiilor în rețea permite să folosim mai puține protocoale pentru routere.

Figure 2.12 *Communication at the network layer*



Nivelul rețea din Internet include protocolul principal, *Internet Protocol* (IP), care definește formatul pachetului la nivelul rețea, numit datagramă. Protocolul IP definește, de asemenea, formatul și structura adreselor utilizate la acest nivel. De asemenea, IP este responsabil pentru dirijarea unui pachet de la sursă la destinație, lucru realizat prin intermediul routerelor, fiecare dintre care trimite datagrama către următorul router din calea sa.

IP este un protocol fără conexiune (connectionless) care nu oferă vreun control al fluxului sau al erorilor și nici servicii de control al congestiei (aglomerării). Aceasta înseamnă că, în cazul în care oricare dintre aceste servicii este necesar pentru o aplicație, aplicația ar trebui să se bazeze doar pe protocolul de nivel transport.

Nivelul rețea include, de asemenea, protocoale de rutare unicast (unu la unu) și multicast (unu la mulți). Un protocol de rutare nu ia parte la rutare (este responsabilitatea IP), dar creează tabele de redirecționare (forwarding tables) pentru routere care să le ajute pe acestea în procesul de rutare. Nivelul rețea include, de asemenea, câteva protocoale auxiliare care ajută IP-ul în sarcinile sale de livrare și rutare:

- Protocolul Internet Control Message (ICMP) ajută IP-ul să raporteze unele probleme ce apar la rutarea unui pachet.
- Protocolul de gestionare a grupurilor de Internet (IGMP - Internet Group Management Protocol) este un alt protocol care ajută IP-ul în multitasking.
- Protocolul de configurare dinamică a host-ului (DHCP - Dynamic Host Configuration Protocol) ajută IP-ul să obțină adresa logică (IP-ul) a host-ului.
- Protocolul de rezoluție adresă (ARP - Address Resolution Protocol) este un protocol care ajută IP-ul să găsească adresa fizică (MAC-ul) a host-ului sau a unui router, atunci când este dată adresa logică (IP-ul).

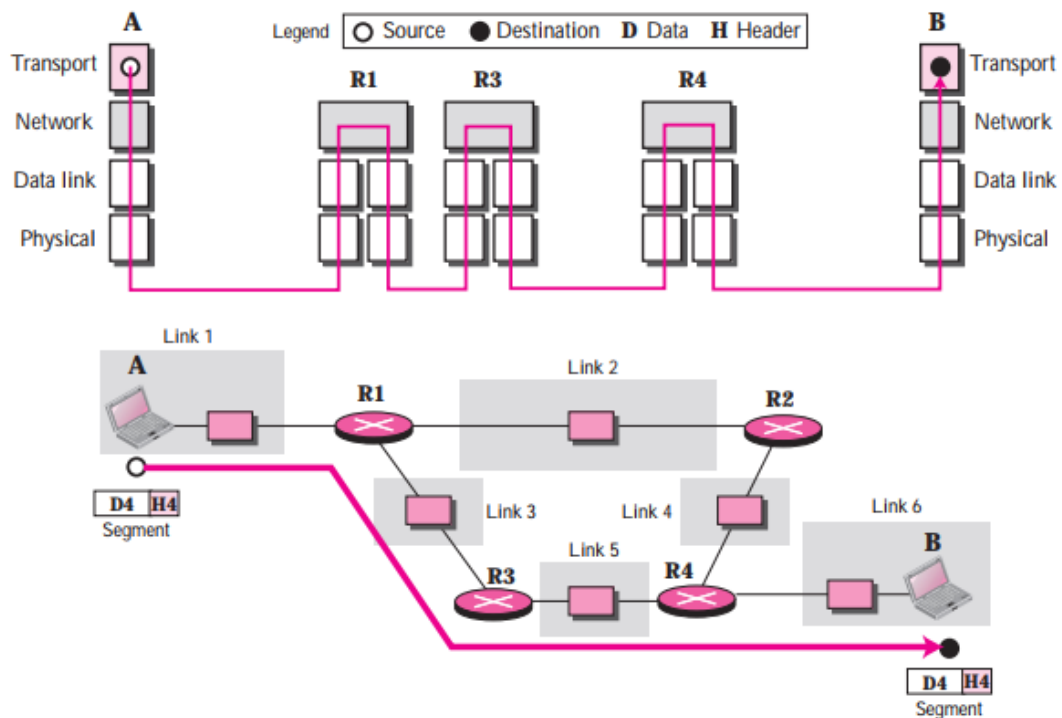
Nivelul transport

Conexiunea logică la nivelul transport este de asemenea de tip end-to-end. Nivelul de transport de la host-ul sursă preia mesajul de la nivelul aplicație, îl încapsulează într-un pachet de la nivelul transport (numit segment sau user datagramă în diferite protocoale) și îl trimite, prin conexiunea logică (imaginară), la nivelul transport de la host-ul destinație.

Cu alte cuvinte, nivelul transport este responsabil de furnizarea serviciilor către nivelul aplicație: să primească un mesaj de la un program de aplicație care rulează pe hostul sursă și să-l livreze la programul de aplicație corespunzător de la host-ul destinație.

Nivelul transport trebuie să fie independent de nivelul aplicație. În plus, vom vedea că avem mai multe protocoale la nivelul transport, ceea ce înseamnă că fiecare program de aplicație poate utiliza protocolul care se potrivește cel mai bine necesităților sale.

Figure 2.13 *Communication at the transport layer*



Există câteva protocoale la nivelul transport, fiecare având o anumită sarcină. Protocolul principal, Transmission Control Protocol (TCP), este un protocol orientat la conexiune, care stabilește mai întâi o conexiune logică între nivelele de transport de la cele două host-uri înainte de a transfera date. Se creează o conductă logică între două TCP-uri pentru transferul unui flux de octeți. TCP asigură:

- controlul fluxului (potrivind rata de trimitere a sursei cu rata de primire a destinației pentru a preveni supraaglomerarea destinației)
- controlul erorilor (pentru a garanta că segmentele ajung la destinație fără eroare și a retrimite pe cele corupte)
- controlul congestiei (pentru a reduce pierderea de segmente din cauza congestiei în rețea).

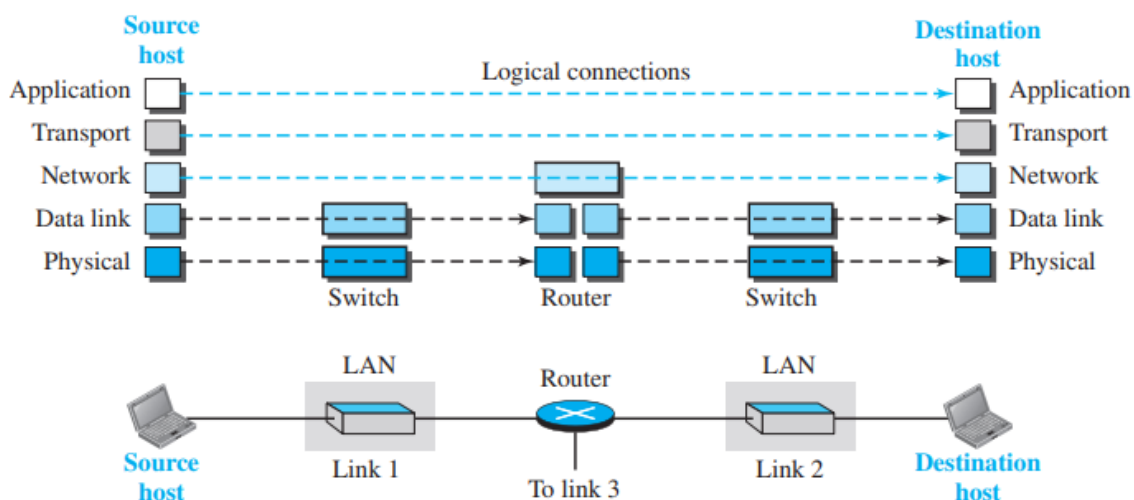
Celălalt protocol comun, *User Datagram Protocol* (UDP), este un protocol fără conexiune, care transmite user datagrame fără a crea mai întâi o conexiune logică. În UDP, fiecare user datagramă reprezintă o entitate independentă, fără a fi legată de cea anterioară sau următoarea (sensul termenului fără conexiune). UDP este un protocol simplu care nu oferă controlul fluxului, erorilor sau al congestiei. Simplitatea sa este atractivă pentru un program de aplicație care necesită transmiterea unor mesaje scurte și nu își poate permite retransmiterea pachetelor implicate în TCP, atunci când un pachet este corupt sau pierdut.

Un nou protocol, *Stream Control Transmission Protocol* (SCTP) este proiectat să corespundă noilor aplicații care apar în multimedia.

Nivelul aplicație

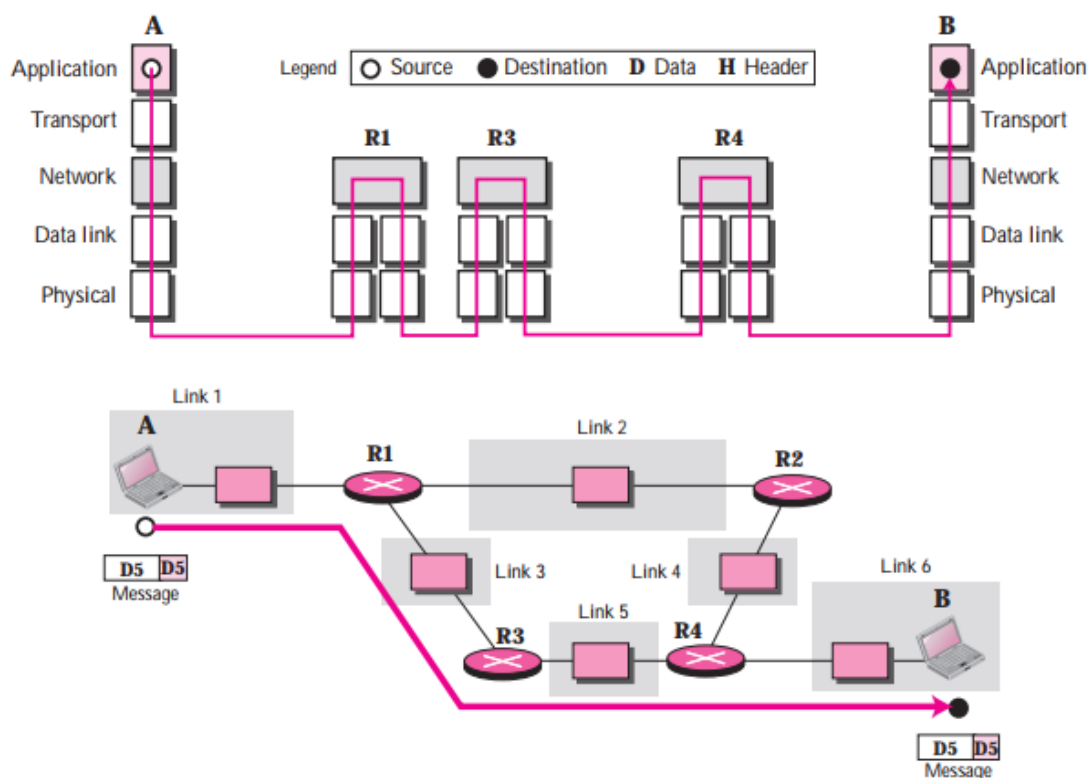
După cum arată Figura 2.6, conexiunea logică între cele două nivele de aplicație este end-to-end. Cele două nivele de aplicație interschimbă mesaje între ele, ca și cum ar exista o punte între cele două nivele. În realitate comunicarea se face prin parcurgerea tuturor nivelelor.

Figure 2.6 Logical connections between layers of the TCP/IP protocol suite



Comunicarea la nivelul aplicație se face între două procese (două programe care rulează la acest nivel). Pentru a comunica, un proces trimite o cerere către celălalt proces și primește un răspuns. Comunicarea proces-proces este sarcina nivelului aplicație. Nivelul aplicație din Internet include multe protocoale predefinite, dar un utilizator poate crea, de asemenea, o pereche de procese care vor fi rulate la cele două host-uri.

Figure 2.14 Communication at the application layer

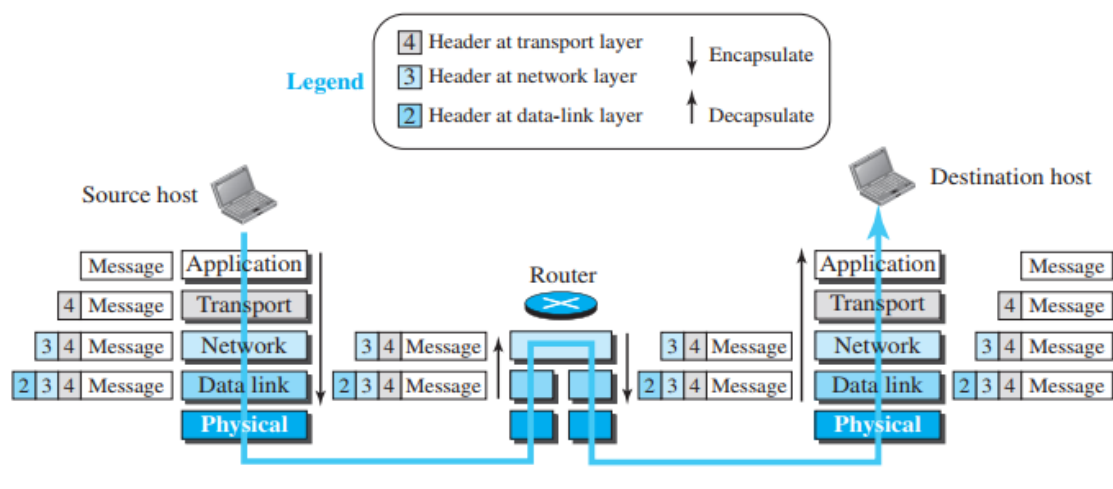


- Protocolul de transfer de hipertext (HTTP - Hypertext Transfer Protocol) este o tehnologie pentru accesarea World Wide Web (WWW).
- Protocolul simplu de transfer prin poștă electronică (SMTP - Simple Mail Transfer Protocol) este principalul protocol utilizat în serviciul de poștă electronică (e-mail).
- Protocolul de transfer de fișiere (FTP - File Transfer Protocol) este utilizat pentru transferul fișierelor de la un host la altul.
- Rețelele de terminale (TELNET - Terminal Network) și Secure Shell (SSH) sunt utilizate pentru accesarea de la distanță a unui site.
- Protocolul de gestionare a rețelei (SNMP - Simple Network Management Protocol) este utilizat de un administrator pentru a gestiona Internetul la nivel global și local.
- Sistemul de nume de domeniu (DNS - Domain Name System) este utilizat de alte protocoale pentru a găsi adresa nivelului de rețea (IP-ul) a unui calculator.
- Protocolul de gestionare a grupurilor de Internet (IGMP - Internet Group Management Protocol) este utilizat pentru a aduna utilizatorii într-un grup.

Încapsularea și decapsularea

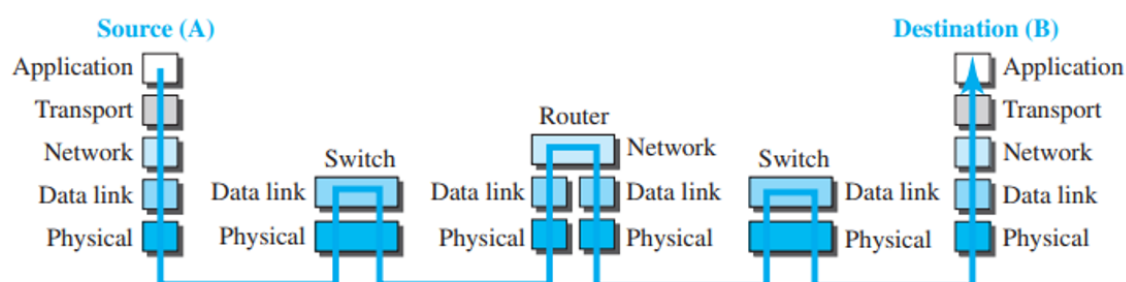
Unul dintre conceptele importante în arhitectura pe nivele ale protocoalelor pe Internet este încapsularea/decapsularea. Figura 2.8 prezintă acest concept pentru internetwork-ul de dimensiuni mici din Figura 2.5.

Figure 2.8 Encapsulation/Decapsulation



Nu am arătat nivelele pentru comutatoarele de nivelul 2 (switch-uri), deoarece nu se face încapsulare/decapsulare la trecerea prin acest dispozitiv. În Figura 2.8, prezentăm încapsularea de la host-ul sursă, decapsularea de la host-ul destinație și încapsularea și decapsularea datelor ce trec prin router.

Figure 2.5 Communication through an internet



Încapsularea de la host-ul sursă

La sursă se realizează doar încapsularea datelor.

1. La nivelul aplicație, datele care urmează să fi transmise poartă denumirea de mesaj. În mod normal, un mesaj nu conține niciun antet (header) sau trailer (remorcă), dar dacă se întâmplă, ne referim la tot conținutul ca un mesaj. Mesajul este transmis nivelului transport.
2. Nivelul transport preia mesajul, se adaugă un antet, care conține identificatorii programelor de aplicație sursă și destinație ce intenționează să comunice, plus alte informații care sunt necesare pentru livrarea end-to-end a mesajului, cum ar fi informațiile necesare pentru controlul fluxului, controlul erorii sau controlul congestiei. Rezultatul reprezintă un pachet de nivel transport, care se numește segment (în TCP) sau user datagramă (în UDP). Nivelul transport transmite pachetul format către nivelul rețea.
3. Nivelul rețea preia pachetul de la nivelul transport și adaugă propriul antet. Antetul conține adresele logice ale host-urilor sursă și destinație și alte informații utilizate pentru verificarea erorilor din antet, informații despre fragmentare și altele. Rezultatul reprezintă un pachet de nivel rețea, numit datagramă. Nivelul rețea transmite pachetul format către nivelul legătură de date.
4. Nivelul legătură de date preia pachetul de la nivelul rețea și adaugă propriul său antet, care conține adresele de nivel legătură de date (fizice - MAC) ale host-ului și a următorului hop (routerul). Rezultatul reprezintă un pachet de nivel legătură de date, care se numește cadru (frame). Cadrul este transmis nivelului fizic, care transmite semnalul corespunzător către destinație.

Decapsularea și încapsularea la nivelul routerului

La router, avem atât decapsulare, cât și încapsulare, deoarece routerul este conectat la două sau mai multe link-uri.

1. După ce șirul de biți este livrat nivelului legătură de date, acest nivel decapsulează (extrage) datagrama din cadru (frame) și o transmite nivelului rețea.
2. Nivelul rețea inspectează adresele sursă și destinație din antetul datagrammei și își consultă tabelul de redirecționare (forwarding table) pentru a găsi următorul hop către care urmează să fie livrată datagrama. Conținutul datagrammei nu trebuie să fie modificat de nivelul rețea al routerului, cu excepția cazului în care este necesar de fragmentat datagrama dacă este prea mare pentru a fi trecută prin următorul link. Datagrama este apoi transmisă nivelului legătură de date a următorului link.
3. Nivelul legătură de date al următorului link încapsulează datagrama într-un cadru și îl transmite nivelului fizic pentru a fi transmis mai departe.

Decapsularea la host-ul destinație

La host-ul destinație, fiecare nivel doar decapsulează pachetul primit, elimină header-ul corespunzător și livrează restul datelor protocolului de nivel superior până când mesajul ajunge la nivelul aplicație. Este necesar să spunem că decapsularea de la nivelul host-ului implică verificarea erorilor.

Adresarea în cadrul protocoalelor

Merită menționat un alt concept legat de stiva de protocoale pe Internet, *adresarea*. După cum am menționat anterior, avem o comunicare logică între perechile de nivele din acest model. Orice comunicare care implică două părți are nevoie de două adrese: adresa sursă și adresa destinație. Deși pare că avem nevoie de cinci perechi de adrese, câte o pereche pentru fiecare nivel, în mod normal avem doar patru, deoarece nivelul fizic nu are nevoie de adrese; unitatea de date la nivelul fizic este bitul, care cu siguranță nu poate avea o adresă. Figura 2.9 prezintă adresarea de la fiecare nivel.

Figure 2.9 Addressing in the TCP/IP protocol suite

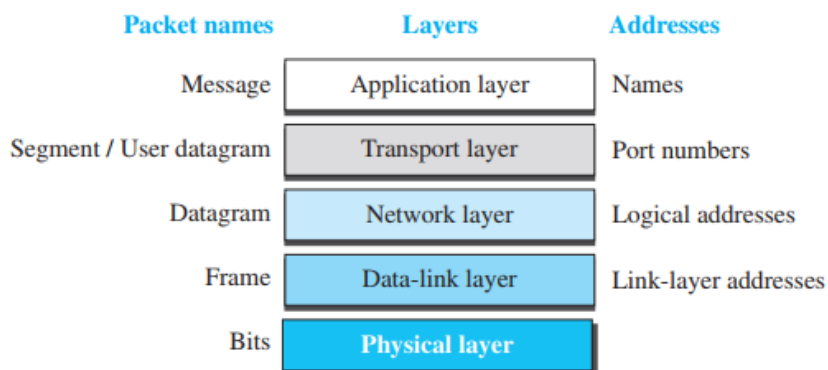
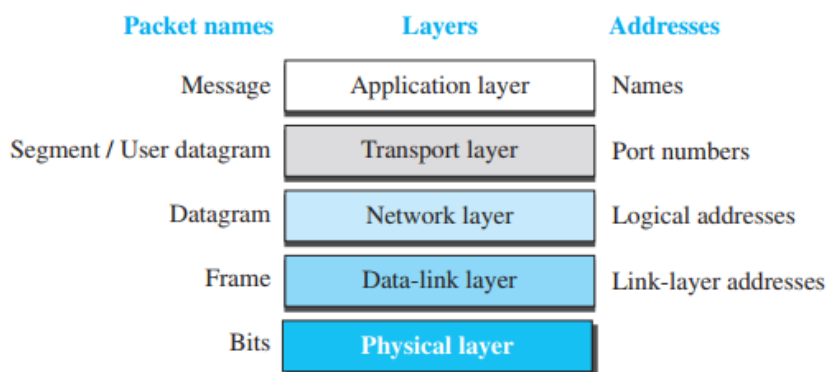


Figure 2.9 *Addressing in the TCP/IP protocol suite*



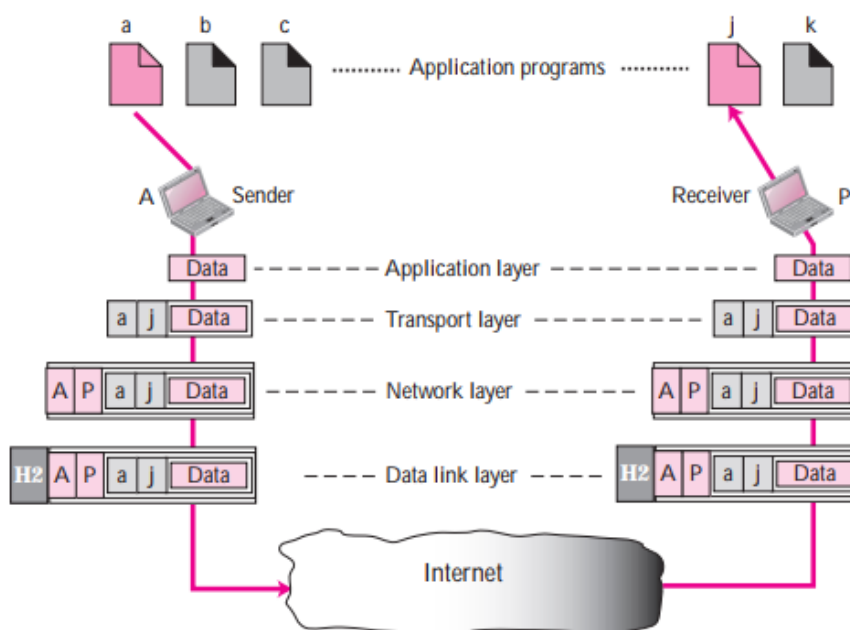
După cum arată figura, există o relație între nivel, adresa utilizată la acel nivel și numele pachetului la acel nivel. La nivelul aplicație, în mod normal, folosim nume pentru a defini site-ul care oferă servicii, cum ar fi someorg.com, sau adresa de e-mail, cum ar fi somebody@coldmail.com. La nivelul transport, adresele se definesc prin numere de port, iar acestea definesc programele de la nivelul aplicație de la sursă și destinație. Numerele de port sunt adrese locale care disting între mai multe programe ce rulează în același timp. La nivelul rețea, adresele sunt globale, cu întregul Internet ca obiectiv. O adresă de la nivelul rețea definește în mod unic conexiunea unui dispozitiv la Internet. Adresele de nivel legătură de date, deseori numite adrese MAC, sunt adrese definite local, fiecare definind un host sau un router concret într-o rețea (LAN sau WAN).

Adresele de porturi

Adresa IP și adresa fizică sunt necesare pentru ca pachetul de date să călătorească de la o sursă la destinație. Totuși, sosirea la host-ul destinație nu este obiectivul final al comunicațiilor de date pe Internet. De regulă, sistemul trimite de la un host la altul nu doar datele de interes. Astăzi, calculatoarele sunt dispozitive care pot rula mai multe procese în același timp. Obiectivul final al comunicării pe Internet este ca un proces să comunice cu un alt proces. De exemplu, calculatorul A poate comunica cu calculatorul C folosind TELNET. Simultan, calculatorul A comunică cu calculatorul B folosind protocolul de transfer de fișiere (FTP). Pentru ca aceste procese să primească simultan date, avem nevoie de o metodă pentru etichetarea diferitelor procese. Cu alte cuvinte, procesele au nevoie de adrese. În arhitectura TCP/IP, eticheta atribuită unui proces se numește adresă de port. O adresă de port în TCP/IP are o lungime de 16 biți.

Figura 2.18 prezintă două calculatoare care comunică prin Internet. Pe calculatorul sursă rulează în acest moment trei procese cu adresele de port a, b și c. Pe calculatorul destinație rulează două procese în acest moment, cu adresele de port j și k. Procesul a de la calculatorul sursă trebuie să comunice cu procesul j al calculatorului destinație. De menționat că deși ambele calculatoare folosesc aceeași aplicație, de exemplu FTP, adresele de port sunt diferite, deoarece un proces este program client, iar celălalt este program server.

Figure 2.18 Example 2.6: port numbers

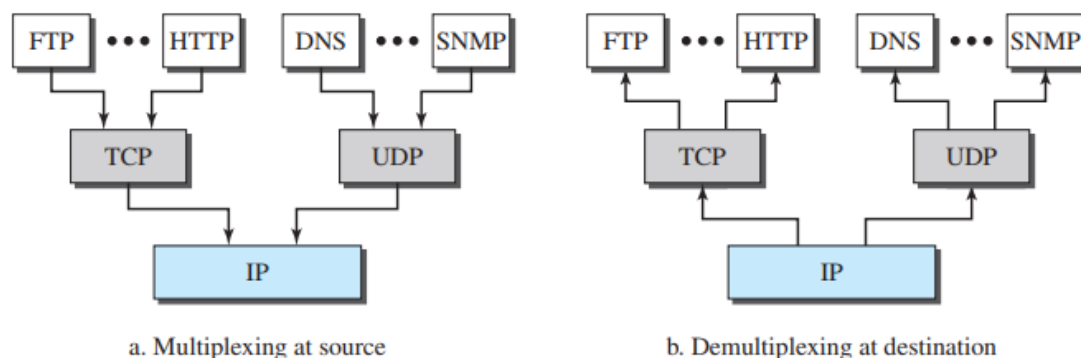


Pentru a arăta că datele de la procesul a trebuie să fie livrate procesului j și nu k, nivelul de transport încapsulează datele de la nivelul aplicație într-un pachet și adaugă două adrese de port (a și j), ale procesului sursă și destinație. Pachetul de la nivelul transport este apoi încapsulat într-un alt pachet la nivelul rețea, adăugându-se adresele logice ale sursei și destinației (A și P). În cele din urmă, acest pachet este încapsulat într-un cadru ce conține și adresele fizice ale sursei și destinației (mai exact, a hop-ului următor). În Figură nu am arătat adresele fizice pentru că acestea se schimbă de la hop la hop. De menționat că adresele logice și adresele de port rămân aceleași în drumul pachetului de la sursă la destinație. Există câteva excepții de la această regulă despre care vom discuta mai târziu.

Multiplexarea și demultiplexarea

Întrucât suita de protocoale TCP/IP folosește mai multe protocoale la unele nivele, putem spune că avem multiplexare la sursă și demultiplexare la destinație. Multiplexarea în acest caz înseamnă că un protocol de la un nivel poate încapsula un pachet venit de la mai multe protocoale de nivel superior (pe rând, câte unul); demultiplexarea înseamnă că un protocol poate decapsula și livra un pachet la mai multe protocoale de nivel superior (pe rând, câte unul). Figura 2.10 prezintă conceptul de multiplexare și demultiplexare la cele trei nivele superioare.

Figure 2.10 *Multiplexing and demultiplexing*



Pentru a putea multiplexa și demultiplexa, un protocol trebuie să aibă un câmp în antetul său pentru a identifica la ce protocol aparțin pachetele încapsulate. La nivelul transport, UDP sau TCP pot accepta un mesaj de la mai multe protocoale de nivelul aplicație. La nivelul rețea, IP poate accepta un segment de la TCP sau o user datagramă de la UDP. De asemenea, IP poate accepta pachete de la alte protocoale, cum ar fi ICMP, IGMP ș.a. La nivelul legăturii de date, un cadru poate transporta pachetul venit de la IP sau de la alte protocoale, cum ar fi ARP.

Modelul OSI

Deși, atunci când se discută despre Internet, ne referim la suita de protocoale TCP/IP, această suită nu este singurul model de arhitectură a rețelei. Înființată în 1947, Organizația Internațională pentru Standardizare (ISO - International Organization for Standardization) este un organism multinațional dedicat acordului mondial privind standardele internaționale. Aproape trei sferturi din țările din lume sunt reprezentate în ISO. Un standard ISO care acoperă toate aspectele comunicațiilor de rețea este modelul de interconectare a sistemelor deschise (OSI - Open Systems Interconnection). A fost introdus pentru prima dată la sfârșitul anilor '70.

Un sistem deschis reprezintă un set de protocoale care permite la oricare două sisteme diferite să comunice indiferent de arhitectura lor de bază. Scopul modelului OSI este de a arăta cum se poate facilita comunicarea între diferite sisteme, fără a necesita modificări la nivel de hardware și software de bază. Modelul OSI nu este un protocol; este un model de referință pentru înțelegerea și proiectarea unei arhitecturi de rețea flexibile, robuste și interoperabile. Modelul OSI se dorea a fi baza pentru crearea protocoalelor în stiva OSI.

Aşadar, modelul OSI reprezintă un ghid pentru dezvoltarea de protocoale de reţea universal compatibile. Este format din şapte nivele, fiecare definind o parte a procesului de transferare a informaţiei printr-o reţea (a se vedea Figura 2.11): fizic, legătură de date, reţea, transport, sesiune, prezentare, aplicaţie.

Figure 2.11 *The OSI model*

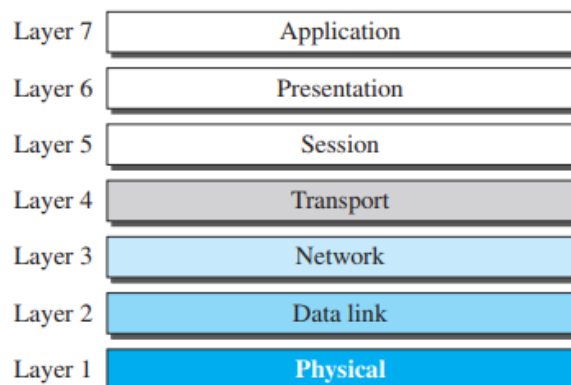
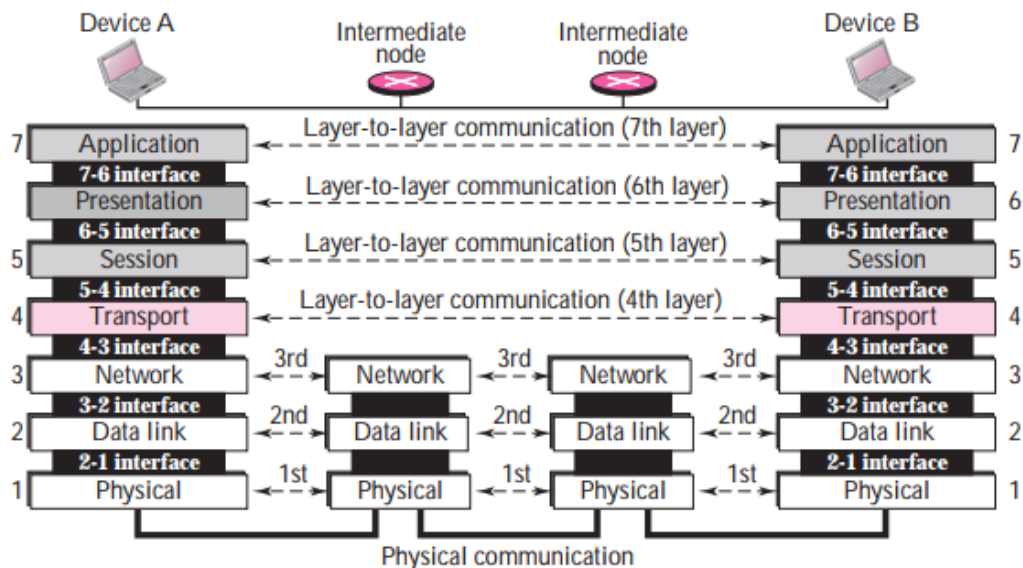


Figure 2.4 *OSI layers*



Nivelele fizic, legătură de date și rețea reprezintă un suport pentru rețea. Nivelele sesiune, prezentare și aplicație reprezintă un suport pentru utilizator, iar nivelul transport asigură legătura dintre nivelele ce susțin rețeaua și nivelele ce susțin utilizatorul.

Nivelul fizic coordonează transmiterea șirului de biți prin mediul fizic; Nivelul legătură de date este responsabil de livrarea unităților de date de la o stație la alta în cadrul unui link fără erori. Nivelul rețea este responsabil pentru livrarea de la sursă la destinație a pachetului prin mai multe link-uri din rețea. Nivelul transport este responsabil de livrarea de la proces la proces a mesajului. Nivelul sesiune stabilește, menține și sincronizează interacțiunile dintre dispozitivele de comunicare. Nivelul prezentare asigură interoperabilitatea între dispozitivele de comunicare prin transformarea datelor într-un format convenit de comun acord. Nivelul de aplicație permite utilizatorilor să acceseze rețeaua.

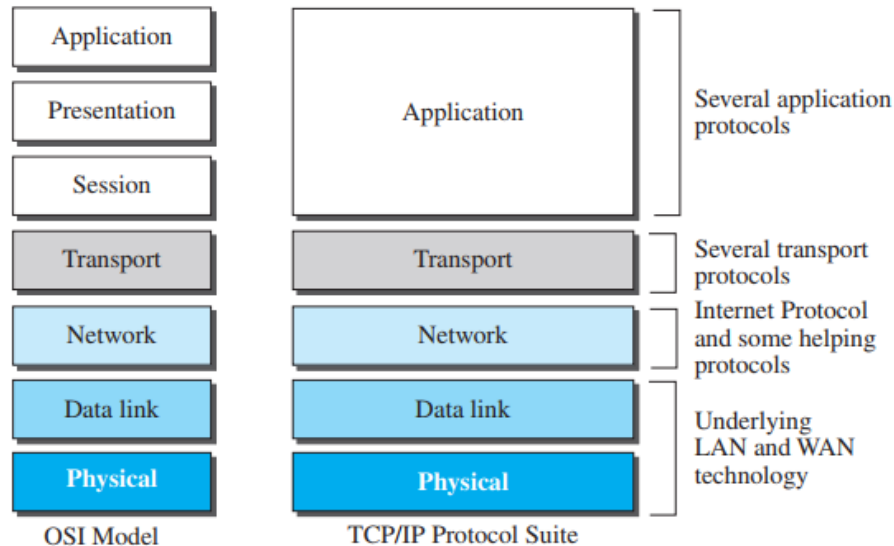
Figure 2.6 *Summary of OSI layers*

Application	To allow access to network resources	7
Presentation	To translate, encrypt, and compress data	6
Session	To establish, manage, and terminate sessions	5
Transport	To provide reliable process-to-process message delivery and error recovery	4
Network	To move packets from source to destination; to provide internetworking	3
Data link	To organize bits into frames; to provide hop-to-hop delivery	2
Physical	To transmit bits over a medium; to provide mechanical and electrical specifications	1

OSI versus TCP/IP

Când comparăm modelul OSI cu TCP/IP, descoperim că două nivele, sesiune și prezentare, lipsesc din suita de protocoale TCP/IP. Aceste două nivele nu au fost adăugate la suita de protocoale TCP/IP după publicarea modelului OSI. Nivelul aplicație din suită este considerat de obicei combinația a trei nivele în modelul OSI, așa cum este ilustrat în Figura 2.12.

Figure 2.12 *TCP/IP and OSI model*



Două argumente au fost menționate pentru această decizie. În primul rând, TCP/IP are câteva protocoale la nivelul transport. Unele dintre funcționalitățile nivelului sesiune sunt caracteristice unor protocoale ale nivelului transport. În al doilea rând, nivelul aplicație nu este doar o componentă software. Multe aplicații pot fi dezvoltate la acest nivel. Dacă unele dintre funcționalitățile menționate la nivelele sesiune și prezentare sunt necesare pentru o anumită aplicație, acestea pot fi incluse la dezvoltarea acelei aplicații software.

Lipsa succesului modelului OSI

Modelul OSI a apărut după suita de protocoale TCP/IP. Majoritatea experților au fost la început entuziasmați și au crezut că protocolul TCP/IP va fi înlocuit complet de modelul OSI. Acest lucru nu s-a întâmplat din mai multe cauze, dintre care menționăm doar trei, care sunt agreate de toți experții în domeniu:

- Modelul OSI a fost propus atunci când TCP/IP era în plină funcțiune și deja au fost investite sume enorme și mult timp pentru realizarea suitei TCP/IP, iar schimbarea ar fi costat mult.
- Unele nivele din modelul OSI nu au fost complet definite. De exemplu, deși serviciile furnizate de nivelele prezentare și sesiune au fost enumerate în document, protocoale reale pentru aceste două nivele nu au fost complet definite și nici nu au fost complet descrise, iar software-ul corespunzător nu a fost complet dezvoltat.
- Atunci când modelul OSI a fost implementat de careva organizație într-o nouă aplicație, aceasta nu a demonstrat un nivel suficient de ridicat de performanță pentru a determina autoritatea Internet să treacă de la suita de protocoale TCP/IP la modelul OSI.

Concluzii la Tema 1

Schimbul de date reprezintă transferul de date de la un dispozitiv la altul printr-un careva mediu de transmisie. Un sistem de comunicații de date trebuie să transmită cu acuratețe datele la destinația corectă în timp util. Cele cinci componente care alcătuiesc un sistem de comunicații de date sunt mesajul, expeditorul, receptorul, mediul de transmisie și protocolul. Textul, numerele, imaginile, fișierele audio și video reprezintă diferite forme ale informației. Fluxul de date între două dispozitive poate apărea într-unul din cele trei moduri: simplex, half-duplex sau full-duplex.

O rețea este un set de dispozitive de comunicare conectate prin link-uri media. Într-o conexiune punct la punct, două și doar două dispozitive sunt conectate printr-un link dedicat. Într-o conexiune multipunct, trei sau mai multe dispozitive partajează un link. Topologia se referă la aranjamentul fizic sau logic al componentelor unei rețele. Dispozitivele pot fi aranjate într-o topologie plasă, stea, magistrală sau inel.

O rețea poate fi clasificată ca o rețea locală (LAN) sau o rețea de zonă extinsă (WAN). O rețea LAN este un sistem de comunicații de date dintr-o clădire, uzină sau campus sau între clădirile din apropiere. Un WAN este un sistem de comunicații de date care cuprinde state, țări sau întreaga lume. Un internetwork este o rețea de rețele. Internetul este o colecție a unui număr enorm de rețele separate.

Istoria internetului a început cu teoria comutării pachetelor pentru traficul exploziv. Istoria a continuat când ARPA a fost interesată să găsească o modalitate de a conecta calculatoarele, astfel încât cercetătorii pe care i-au finanțat să își poată împărtăși rezultatele, rezultând în crearea de ARPANET. Internetul s-a născut când Cerf și Kahn au conceput ideea unui dispozitiv numit gateway care să servească drept hardware intermediar pentru a transfera date dintr-o rețea în alta. Suita de protocoale TCP/IP a deschis calea pentru crearea Internetului de astăzi. Invenția WWW, utilizarea multimedia și comunicarea peer-to-peer a contribuit la dezvoltarea rețelei Internet.

Un standard de internet este o specificație complet testată. Un proiect de internet este un document de lucru fără statut oficial și cu o durată de viață de șase luni. Un proiect poate fi publicat ca o Cerere de Comentariu (RFC).

Un protocol este un set de reguli care guvernează comunicarea. În arhitectura pe nivele de protocoale, trebuie să urmărim două principii pentru a oferi o comunicare bidirecțională. În primul rând, fiecare nivel trebuie să efectueze două sarcini opuse. În al doilea rând, două obiecte sub fiecare nivel din ambele părți ar trebui să fie identice. În arhitectura pe nivele de protocoale, trebuie să distingem între o conexiune logică și o conexiune fizică. Două protocoale la același nivel pot avea o conexiune logică; o conexiune fizică este posibilă numai prin nivelul fizic.

TCP/IP este o suită de protocoale ierarhice formată din cinci nivele: fizic, legătură de date, rețea, transport și aplicație. Nivelul fizic coordonează funcțiile necesare pentru a transmite un șir de biți printr-un mediu fizic. Nivelul legătură de date este responsabil pentru livrarea unităților de date de la o stație la alta, fără erori. Nivelul rețea este responsabil pentru livrarea de la sursă la destinație a unui pachet prin mai multe link-uri ale rețelei. Nivelul transport este responsabil pentru livrarea process-to-process a întregului mesaj. Nivelul aplicație permite utilizatorilor să acceseze rețeaua.

Patru nivele de adrese sunt utilizate pe Internet conform protocoalelor TCP/IP: adrese fizice (link), adrese logice (IP), adrese de porturi și adrese specifice. Adresa fizică, cunoscută și sub numele de adresă de link, este adresa unui nod așa cum este definită de LAN-ul sau WAN-ul corespunzător. Adresa IP definește în mod unic un host pe Internet. Adresa de port identifică un proces pe un host. O adresă specifică este o adresă user-friendly. Un alt model care definește arhitectura pe nivele a protocoalelor este modelul de interconectare a sistemelor deschise (OSI). Două nivele din modelul OSI, sesiune și prezentare, lipsesc din suita de protocoale TCP/IP. Aceste două nivele nu au fost adăugate la suita de protocoale TCP/IP după publicarea modelului OSI. Nivelul aplicație din suită este de obicei considerat a fi combinația a trei nivele în modelul OSI. Modelul OSI nu a înlocuit suita de protocoale TCP/IP, deoarece a fost finalizat când TCP/IP a fost complet finisat și în plină derulare și pentru că unele nivele din modelul OSI nu au fost complet definite.