

den alagúthoz hozzárendelnek egy súlyt. Egy csomagot csak akkor adnak át egy alagúton, amikor még annak elég súlya van, egyébként eldobják. Például az óceánokon keresztül haladó vonalakat rendszerint 128-as súlyúnak állítják be, így a csomagok a keletkezési kontinensre korlátozhatók azzal, hogy 127 vagy kisebb kezdeti *Élettartam* értéket adunk nekik. Egy alagúton való áthaladás után az *Élettartam* mezőt az alagút súlyával csökkentik.

Bár az MBone forgalomirányítási algoritmus működik, sok kutatást szenteltek a javítására. Egy javaslat megtartja a távolságvektor alapú forgalomirányítást, de az algoritmust hierarchikussá teszi azáltal, hogy az MBone állomásokat körzetekre osztja, és először azok felé irányít (Thyagarajan és Deering, 1995).

Egy másik javaslat, hogy a kapcsolatállapot alapú forgalomirányítás egy módosított formáját használják a távolságvektor alapú forgalomirányítás helyett. Egy IETF munkacsoport foglalkozik azzal, hogy az OSPF-et egy autonóm rendszeren belül képessé tegye a többesküldésre. A keletkező többesküldéses OSPF-et **MOSPF-nek** nevezik (Moy, 1994). A módosítás azt csinálja, hogy az MOSPF által felépített teljes térkép a szokásos forgalomirányítási információkon kívül a többesküldéses szigeteket és alagutakat is nyilvántartja. A teljes topológia birtokában magától értetődő kiszámítani a legjobb utat minden szigettől minden másik szigetig. Például a Dijkstra-algoritmus használható erre.

A kutatás egy második területe az AS-ek közti forgalomirányítás. Itt egy másik IETF munkacsoport egy **PIM-nek (Protocol Independent Multicast – protokollfüggetlen többesküldés)** nevezett algoritmust fejleszt (Huitema, 1995). A PIM-nek két változata van, attól függően, hogy a szigetek sűrűek (majdnem mindenki nézni akarja az adást) vagy ritkák (alig valaki akarja nézni az adást). Mindkét változat a szokásos egyesküldéses forgalomirányító táblázatokat használja egy, a jelenlegit elfedő topológia helyett, mint ahogy a DVMRP és az MOSPF teszi.

A sűrű PIM-ben az ötlet az, hogy a felesleges ágakat le kell nyesni. A nyesés a következőképpen működik: amikor egy többesküldéses csomag a „rossz” alagúton keresztül érkezik meg, egy nyeső csomagot küldenek vissza az alagúton az adónak, utasítva arra, hogy ne küldjön több csomagot a kérdéses forrástól. Amikor egy csomag a „jó” csatormán keresztül érkezett meg, akkor kimásolják az összes többi olyan alagútra, amelyek még nem nyesték le magukat. Ha az összes többi alagút lenyeste magát, és a helyi szigeten belül nincs érdeklődés a csatorna irányába, akkor az mrouter a „jó” csatormán is visszaküld egy nyeső üzenetet. Ily módon a többesküldés automatikusan alkalmazkodik, és csak oda jut el, ahova kell.

A ritka PIM másféleképpen működik. Az ötlet az, hogy gátoljuk meg, hogy az Internet csak azért telítődhessen, mert három ember Berkeleyben egy konferenciahívást akar lebonyolítani egy D osztályú címmel. A ritka PIM találkozási pontok felállításával működik. Egy ritka PIM többesküldéses csoportban minden forrás a találkozási pontokhoz küldi a csomagjait. Bármelyik, a csatlakozásban érdekelt állomás az egyik találkozási pontot kéri meg, hogy egy alagutat állítson fel hozzá. Ily módon minden ritka PIM forgalmat többesküldés helyett egyesküldéssel szállítanak.

Mindent egybevéve, a multimédia egy izgalmas és rohamosan fejlődő terület. Naponta jelentenek be új technológiákat és alkalmazásokat, de a terület egésze még valószínűleg évtizedekig fontos lesz.

7.8. Összefoglalás

A számítógép-hálózatok eredendően nem biztonságosak. Hogy az információt titokban tarthassuk, azt titkosítani kell. A titkosítási protokollok két általános osztályba sorolhatók: a titkos kulcsú (DES, IDEA) és a nyilvános kulcsú (RSA) osztályokba. Ezen protokollok használata magától értetődő, a nehéz rész a kulcszajdalkodás.

A titkosság biztosításán kívül a kriptográfiai protokollok hitelesítést is biztosíthatnak, így amikor Aliz azt gondolja, hogy Bobbal kommunikál, akkor valóban Bobbal kommunikál, és nem Trudyval. Végül, a kriptográfiát arra is fel lehet használni, hogy az üzeneteket oly módon lehessen aláírni, hogy a küldő elküldés után már nem tagadhatja le azokat.

Az Interneten a nevek kezeléséhez egy elosztott adatbázis rendszert, DNS-t használnak. A DNS IP címekből, levélváltásokból és egyéb információból álló rekordokat tárol. Egy DNS kiszolgáló lekérdezésével egy folyamat leképezhet egy Internet körzetnevet az azzal a körzettel való kommunikáláshoz használandó IP címre.

Ahogy a hálózatok egyre nagyobbak lesznek, egyre nehezebb lesz azokat menedzselni. Ezen ok miatt különleges hálózatmenedzselési rendszereket és protokollokat ötlöttek ki, amelyek közül az SNMP a legnépszerűbb. Ez a protokoll lehetővé teszi, hogy a menedzserek az eszközökön belül elhelyezett ügynökökkel kommunikáljanak, kiolvassák azok állapotát és parancsokat adjanak azoknak.

Négy fő hálózati alkalmazás az elektronikus levelezés, a USENET hírek, a World Wide Web és a multimédia (a hálózati videózás és az MBone). A legtöbb e-level rendszer a 821-es és a 822-es RFC-kben definiált levelezőrendszert használja. Az ebben a rendszerben elküldött üzenetek rendszerszintű ASCII fejrészeket használnak az üzenet tulajdonságainak meghatározásához. Ezeket az üzeneteket az SMTP használatával küldik el. Két rendszer létezik az e-level biztonságossá tételére, a PGP és a PEM.

A USENET hírek mindenféle témáról szóló hírcsoportok ezreiből állnak. Az emberek helyileg csatlakozhatnak a hírcsoportokhoz, majd üzeneteket postázhatnak az egész világnak, az NNTP protokollt használva, amely némileg emlékeztet az SMTP-re.

A World Wide Háló egy hipertext dokumentumok összekapcsolására szolgáló rendszer. Minden dokumentum egy HTML-ben frott oldal, esetlegesen más dokumentumokra mutató előkapcsokkal. Egy böngésző úgy jeleníthet meg egy oldalt, hogy egy TCP összeköttetést hoz létre a kiszolgálójával, elkéri a dokumentumot, majd lezárja az összeköttetést. Amikor a felhasználó kiválaszt egy előkapcsot, azt a dokumentumot hasonló módon lehet elhozni. Ily módon kapcsolódnak egymáshoz a dokumentumok az egész világon, mint egy hatalmas pókhálóban.

A multimédia gyorsan emelkedő csillag a hálózatok égboltján. Lehetővé teszi hang és kép digitalizálását és elektronikus továbbítását megjelenítés céljából. A legtöbb multimédia projekt az MPEG szabványokat használja, és az adatot ATM összeköttetéseken keresztül viszi át. Az MBone egy kísérleti világméretű digitális rádió- és televíziószoftver az Interneten.

Feladatok

1. Fejtse meg a következő egyábécés titkosítást. A nyílt szöveg, amely csak betűkből áll, Lewis Carroll egy jól ismert költeményének részlete.

kfd ktbd fzm eubd kfd pzyiom mztx ku kzyg ur bzha kfthcm
 ur mfdm zhx mfnm zhx mdzythc pzq ur ezsszcdm zhx gthcm
 zhx pfa kfd mdz tm sutythc fuk zhx pfdkfdi ntcn fzld pthcm
 sok pztz z stk kfd uamkdim eitdx sdruid pd fzld uoi efzk
 rui mubd ur om zid uok ur sidzfk zhx zyy ur om zid rzk
 hu foiaa mztx kfd ezindhkdi kfda kfzhgdx ftb boef rui kfzk

2. Fejtse meg a következő oszlopos eltolásos rejtjelezést. A nyílt szöveget egy népszerű számítógépes tankönyvből vettük, így a „computer” valószínűleg előfordul benne. A nyílt szöveg kizárólag betűkből áll (szóközők nélkül). A rejtett szöveget az olvashatóság érdekében ötkarakteres blokkokra tördeltük.

aaan cvlre turn dltme aeepb ytust iceat npmey iicgo gorch srsoc
 nntii imiha oofpa gsivt tpsit lbolr otoex

3. A 7.4. ábrán a P dobozok váltakoznak az S dobozokkal. Bár ez az elrendezés esztétikailag kellemes, biztonságosabb-e, mint először venni az összes P dobozt, majd az összes S dobozt?
4. Tegyük fel, hogy egy üzenetet a DES rejtett szövegű blokkokat egymás után fűző módjával titkosítottunk. A rejtett szöveg C_i blokkban egy bit véletlenül 0-ról 1-re változik az átvitel során. Mennyi nyílt szöveg fog összezavarodni ennek eredményeképpen?
5. Vegyük megint a rejtett szövegű blokkokat egymás után fűző módot. Egy 0 bit 1-gyé változtatása helyett egy külön 0-t szúrunk be a titkos szöveg folyamába a C_i blokk után. Mennyi nyílt szöveg fog emiatt összezavarodni?
6. Tervezze meg a DES egy támadását arra alapozva, hogy a nyílt szöveg csak ASCII nagybetűkből és szóközből, vesszőből, pontból, pontosvesszőből, kocsi-vissza-jelből és soremelés jelből áll. Semmit nem tudunk a nyílt szöveg paritásbitjeiről.
7. Hasonlítsa össze a rejtett szövegű blokkokat egymás után fűző módot a rejtett szöveget visszacsatoló móddal egy nagy állomány átviteléhez szükséges titkosítási műveletek szempontjából. Melyik a hatékonyabb és mennyivel?
8. Az RSA nyilvános kulcsú titkosítási rendszert használva, $a=1$, $b=2$ stb. mellett:

- (a) Ha $p=7$ és $q=11$, akkor adja meg d öt érvényes értékét!
- (b) Találja meg e -t, ha $p=13$, $q=31$ és $d=7$!
- (c) $p=5$, $q=11$ és $d=27$ használatával keresse meg az e -t és titkosítsa az „abcdefghij” szöveget!
9. A Diffie–Hellman kulcskicserélést használjuk Aliz és Bob közt egy titkos kulcs létrehozásához. Aliz a következőt küldi Bobnak: (719, 3, 191). Bob a következővel válaszol: (543). Aliz x titkos száma 16. Mi lesz a titkos kulcs?
10. A 7.14. ábrán látható protokollban változtasson meg kismértékben egy üzenetet, hogy ellenállóvá tegye azt a válaszoló támadással szemben. Indokolja meg, miért működik a változtatás!
11. A 7.17. ábra protokolljában miért küldik el A -t nyílt szövegben a titkosított viszonykulccsal együtt?
12. A 7.17. ábra protokolljában kimutattuk, hogy minden nyílt szövegű üzenetet 32 nullával kezdeni biztonsági kockázat. Tegyük fel, hogy minden üzenet egy felhasználónként eltérő véletlenszámmal kezdődik, amely gyakorlatilag egy második titkos kulcs, amelyet csak a felhasználó és a KDC ismer. Ez kiküszöböli-e az ismert nyílt szövegű támadást?
13. A Needham–Schroeder protokollban Aliz két kihívást hoz létre, R_A -t és R_{A2} -t. Ez túlzásnak tűnik. Nem lenne elég egy?
14. A 7.21. ábra hitelesítési protokolljában a 3. üzenetben az R_B -t a K_S -sel titkosítottuk. Szükséges ez a titkosítás, vagy elegendő lett volna azt nyílt szövegben visszaküldeni?
15. A 7.22. ábra aláírási protokollja a következő gyengeséggel bír: ha Bob összeomlik, elvesztheti a RAM-ja tartalmát. Milyen gondokat okozhat ez, és mit tehet, hogy ezeket megelőzze?
16. Miután Ellen bevallotta Marilynnek, hogy megráfalta őt Tom birtoklásának kérdésében, Marilyn ezt a problémát azzal oldotta meg, hogy a további üzenetek tartalmát egy diktafonnak mondja el, és az új titkárnő csak begépelje azokat. Ezután Marilyn úgy tervezte, hogy megvizsgálja a határidőnaplójában levő üzeneteket, miután begépelte azokat, hogy leellenőrizze, hogy pontosan az ő szavait tartalmazzák-e. Használhatja-e az új titkárnő a születésnap-támadást üzenetek hamisításához, és ha igen, hogyan? *Tipp:* igen, használhatja.
17. A vásárlás helyén levő termináloknak, amelyek mágnescsíkos kártyákat és PIN kódokat használnak, van egy végzetes hibájuk: egy rosszindulatú kereskedő mó-

- dosíthatja a kártyaolvasóját, hogy a kártyán levő összes információt és a PIN kódot is elfogja és tárolja, hogy a jövőben további (hamis) tranzakciókat postázhasson. A vásárlás helyén levő terminálok következő generációja olyan kártyákat fog használni, amelyeken teljes CPU, billentyűzet és egy kis képernyő is van. Gondoljon ki ehhez a rendszerhez egy olyan protokollt, amelyet a rosszindulatú kereskedők nem törhetnek fel.
18. A 7.27. ábrán adott információ szerint a *little-sister.cs.vu.nl* gép egy A, B vagy C osztályú hálózaton van?
 19. A 7.27. ábrán nincs a *rowboat* szó után pont. Miért nincs?
 20. Mi a tcp objektum *OBJEKTUMAZONOSÍTÓ*-ja?
 21. Át kell vinni egy SNMP egész számot, melynek értéke 200. Adja meg az ASN.1 átviteli szintaxis szerint elküldött bitek bináris reprezentációját!
 22. Mi az ASN.1 átviteli szintaxisban a következő 11 bites bináris füzér reprezentációja: 11100001111?
 23. Tegyük fel, hogy Önt felbérelte egy hidakat gyártó cég, hogy az egyik hídjukhoz az SNMP-nek megfelelő kódot írjon. Ön elolvasta az összes RFC-t, de még mindig vannak kérdései. Azt javasolja az IAB-nek, hogy valamely helyen adják meg az SNMP változókhöz használt nyelv teljes, formális nyelvtanát. Az IAB reakciója az, hogy egyetértenek Önnel, és egyben kijelölik erre a feladatra. A nyelvtant az 1442-es vagy az 1213-as RFC-hez kellene hozzávenni? Miért? *Tipp*: nem kell elhoznía az RFC-eket, elegendő információt adtunk meg a szövegben.
 24. Néhány e-leveél rendszer egy *Tartalom Visszaküldése* nevű fejrész mezőt is támogat. Ez azt határozza meg, hogy az üzenet törzsét nem kézbesíthetőség esetén vissza kell-e küldeni. Ez a mező a borítékhoz vagy a fejrészhez tartozik?
 25. Az elektronikus levelezőrendszereknek szükségük van címjegyzékekre, hogy az emberek e-leveél címét ki lehessen keresni. Ilyen címjegyzékek felépítéséhez a neveket szét kell törölni a szokásos összetevőkre (vezetéknév, keresztnév), hogy a keresést lehetővé tegyék. Vessen fel néhány olyan gondot, amelyet egy világméretű szabványnak meg kell oldania ahhoz, hogy elfogadható legyen.
 26. Egy bináris állomány 3072 bájt hosszú. Milyen hosszú lesz akkor, ha base64 kódolást használunk, minden 80 elküldött bájt után és a végén egy CR+LF párral?
 27. Vegyük az idézett-nyomtatható MIME kódolási sémát. Említsen meg egy problémát, amelyet nem említettünk a szövegben, és javasoljon egy megoldást.
 28. Adjon két okot, amiért a PGP az üzeneteket tömöríti.

29. Tegyük fel, hogy valaki felállít egy szabadságos démont, majd éppen kilépés előtt még elküld egy levelet. Sajnos a címzett is szabadságon van egy hétig, és szintén felállított egy szabadságos démont. Mi fog történni? Ide-oda fognak menni a dobozolt válaszok, amíg valaki vissza nem érkezik?
30. Feltéve, hogy az Interneten mindenki PGP-t használ, lehet-e egy tetszőleges Internet címre úgy PGP üzenetet küldeni, hogy azt minden érintett helyesen dekódolhassa? Indokolja a választ!
31. A PGP nem támogatja úgy a kanonizációt, mint a PEM. Miért nem?
32. Próbálja megtippelni, hogy mit jelenthet a következő smiley :-X (néha :-#-nak is írják).
33. Meddig tart egynapi híradagot szétosztani egy 50 Mb/s-os műholdas csatornán keresztül?
34. A 7.56. ábrán felsorolt parancsok közül melyek elméletileg redundánsak?
35. Egy nagy hálózat gépek $n \times n$ -es rácsából áll. Az összes belső csomópontnak négy szomszédja van, a szélein (sarkain) levőknek három (kettő). Ha egy m bájtos cikket postáznak valamely gépre az NNTP-t használva, hány bájt sávszélesség fogy el, amíg az az összes többi géphez eljut (leszámítva az NNTP által okozott többletet és csak az üzenetbájtokat számolva)?
36. Ismét az előző probléma, csak most azt számolja ki, hogy közelítőleg mennyi sávszélességre lenne szükség ennek az üzenetnek a levelezési lista használatával történő szétosztásához. Mennyivel több ez, mint az előző problémában?
37. Amikor a Háló-oldalakat kiküldik, akkor MIME fejrészek előzik meg azokat. Miért?
38. Mikor van szükség külső megjelenítőkre? Honnan tudja a böngésző, hogy melyiket kell használnia?
39. Képzeljük el, hogy valaki a Stanford Egyetem Számítástechnikai Tanszékén épp most írt egy programot, amelyet FTP-vel akar terjeszteni. A programot a következő FTP könyvtárba helyezi el: *fip/pub/freebies/newprog.c*. Mi lesz várhatóan ennek a programnak az URL-je?
40. A 7.60. ábrán, az címkében be van állítva az *ALT* paraméter. Milyen körülmények közt és hogyan használja ezt a böngésző?
41. Hogyan lehet egy képet HTML-ben kattinthatóvá tenni? Adjon egy példát!

42. Mutassa meg, hogy az <A> címkére szükség van arra, hogy az „ACM” karakterláncot egy, a <http://www.acm.org>-ra mutató élőkapocssá tegyük.
43. Tervezzen egy űrlapot egy új cég, az Interburger számára, amely hamburgerek rendelését teszi lehetővé az Interneten keresztül. Az űrlap tartalmazza a vásárló nevét, címét és városát, valamint a választott méretet (hatalmas vagy gigászi), és egy sajt-lehetőséget. A burgereket utánvétellel kell kifizetni, így nincs szükség hitelkártya-információra.
44. A Javának nincsenek struktúrái, mint a C-nek, vagy rekordjai, mint a Pascalnak. Van valami más lehetőség arra, hogy eltérő változók egy csoportját egyetlen adat-típusba vonjuk össze? Ha igen, mi az?
45. A 7.75. ábrán látható adatszerkezeteket használva, sorolja fel egy új URL ellenőrzésének pontos lépéseit, amelyek által láthatjuk, hogy benne van-e már az `url_table`-ben.
46. Tegyük fel, hogy azon erőfeszítései érdekében, hogy piaciorientáltabb legyen, a KGB elüzletiesedik és egy hirdetési ügynökséget bíz meg azzal, hogy tervezzen neki Háló-oldalt. Az Ön társaságát mint külső konzultáns céget felbérelték arra, hogy megvalósítsák azt. Írja meg a következő oldalon látható Háló-oldalt előállító HTML-t.

Üdvözljük a KGB WWW honlapján!

A nemrégiben végbement privatizáció következményeként a KGB örömmel jelenti be, hogy eddig csak nagyobb kormányok számára elérhető sok szép termék és szolgáltatás immáron üzleti alapon is elérhető.

Versenyképes árak! A diszkrét szolgáltatást biztosítjuk!

- Termékek
 - Nukleáris fegyverek (kis, közepes, nagy, jumbo)
 - Kéműhaldak (Tartsa a szemét a szomszédain!)
 - Alacsony radarláthatóságú szupersonikus repülőgépek (Húzzon el láthatatlanul barátai házaik felett!)
- Szolgáltatások
 - Az Ön tetszése szerinti szervezetbe beépített ember elhelyezése
 - Puccsok (kormányzati és vállalati egyaránt)
 - Segítség az Ön saját vegyifegyver-laboratóriumának felállításához.
- Alagsori kiadásítás
 - Felix Dzerzhinsky összegyűjtött művei (Korlátozott példányszámú kiadás!)
 - Légifelvétel Afganisztánról (1984 körül)
 - Minőségi bolgár gyártmányú tankok (95 százalékos engedmény!)

Webmaster@kgb.ru

47. A C-ben és a C++-ban egy egész szám méretét a nyelv nem határozza meg. A Javában igen. Adjon egy érvt a C módszere mellett, és egyet a Java módszere mellett.
48. Tegyük fel, hogy a Háló 10 millió oldalt tartalmaz, amelyek átlagosan egyenként 10 élőkapcot tartalmaznak. Egy oldal elhozása átlagosan 100 ms-ig tart. Mi a legrövidebb idő az egész Háló indexeléséhez?
49. Egy kompaktlemez 650 MB adatot tartalmaz. Használják tömörítést az audio CD-ken? Magyarázza meg az érvelését!
50. Mi a tömörítetlen VGA, 8 bit/képpontos, 40 kép/s-os mozgókép sávszélességigénye?
51. A 7.76.(c) ábrán a kvantálási zaj a 3 bites minták miatt következik be. Az első minta, a 0, még pontos, de a következő néhány nem az. Mi a periódus 1/32-nél, 2/32-nél és 3/32-nél levő minták százalékos hibaaránya?
52. Egy MPEG keretben előforduló 1 bites hiba befolyásolhat többet, mint azt a keretet, amelyben előfordult? Indokolja választát!
53. Vegyük a szövegben említett 100 000 ügyfeles videokiszolgáló példáját. Tegyük fel, hogy a filmek felét este 8 és 10 óra közt kell leadni. Mennyi filmet kell ezalatt az idő alatt egyszerre átvinnie? Ha minden film 4 Mb/s-ot igényel, hány OC-12-es összeköttetésre lesz szüksége a kiszolgálónak a hálózathoz?
54. Tegyük fel, hogy Zipf törvénye érvényes egy 10 000 filmet tartalmazó kiszolgálóhoz intézett kérésekre. Ha a kiszolgáló az 1000 legnépszerűbb filmet mágneslemezen, a többi 9000-et optikai lemezen tartja, adjon egy kifejezést az összes hozzáférés azon törtrészére, amely a mágneslemezhez irányul. Írjon egy kis programot, amely numerikusan kiértékeli ezt a kifejezést.
55. Az MPEG PES csomagok tartalmaznak egy mezőt, amely az aktuális átvitel copyright állapotát adja meg. Milyen felhasználása képzelhető el ennek a mezőnek?

8. Osztott alkalmazások

Írta: Dr. Harangozó József

Lektorálta: Ercsényi András

8.1. Bevezetés

E fejezetben három olyan alkalmazás bemutatására kerül sor, amelyek széles körben használatosak, és szorosan kapcsolódnak az előző fejezetben ismertetett alkalmazásokhoz, ugyanakkor onnan valamilyen okból kimaradtak.

A 7. fejezetben alkalmazás specifikus protokollokról és alkalmazásokról olvashatunk. Ezek között azonban nem szerepelnek olyan gyakori alkalmazások, mint amilyen a *fájlvitel* és a *terminálkezelés*. Mivel mindkét alkalmazás alapvető fontosságú, ezért itt foglalkozunk velük. Egyrészt a szerző eredeti koncepcióját megtartva a bemutatjuk TCP/IP alapú terminálkezelést és fájlvitelt, ugyanakkor mivel az OSI világ is tovább él, s számos helyen ilyen alapú alkalmazásokkal találkozhatunk, az OSI alapú alkalmazások ismertetését felvállaljuk. Az OSI alapú rendszerek bemutatása ezen kívül azért is lényeges, mert egy új fogalom, az ún. *virtuális eszköz* bevezetésére is sor kerülhet, ami adott esetben más alkalmazások tervezésénél is hasznos ismeret lehet.

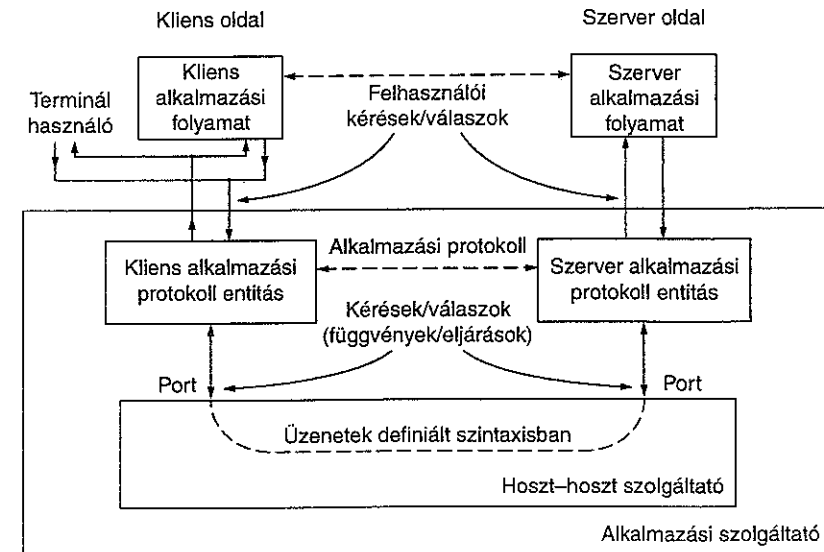
A harmadik bemutatásra kerülő alkalmazás az elektronikus levelező rendszer. A TCP/IP alapú elektronikus levelező rendszereket a 7.4 fejezet tárgyalja, de mivel az ISO alapú levelező rendszer szintén létezik és használják is, ismertetését szintén fontosnak tartjuk.

Mindhárom alkalmazás bemutatásával célunk az is, hogy az olvasó összehasonlítsa tehessen a TCP/IP alapú és az OSI alapú alkalmazások között.

TCP/IP alapú alkalmazási protokollok

Ahogy az az előzőekből látható, a TCP/IP protokoll-készletben nincsenek az alkalmazási folyamatot támogató viszony szintű és megjelenítési szintű funkciócsoportok, csupán alkalmazási réteg létezik. Így az alkalmazási funkcionális elem közvetlenül a TCP szolgáltatait igénybevéve tud kapcsolatot teremteni a távoli alkalmazási folyamattal.

Az **alkalmazási folyamat** fogalom azokat a futó programokat, szoftver termékeket jelenti, amelyek az alkalmazás megvalósításával szoros kapcsolatban vannak, azaz amelyek az alkalmazást megvalósítják. Egy *kliens alkalmazási folyamat* például egy olyan számítógépen futó program lehet, amelyik egy távoli másik gépen futó szerverhez, mint alkalmazási folyamathoz való hozzáférést intézi. Az **alkalmazási szolgáltató** teszi lehetővé, hogy a kliens alkalmazási folyamat a távoli gépen futó szerver alkal-



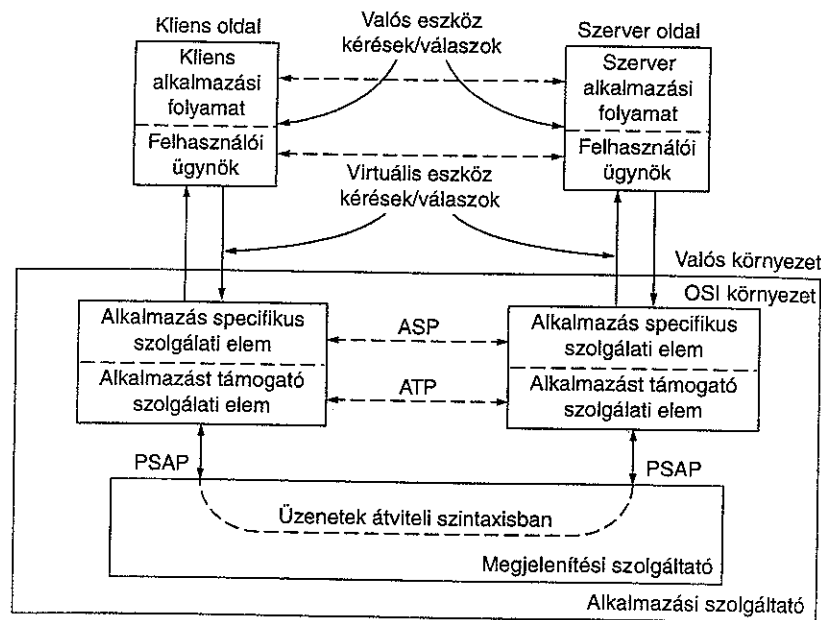
8.1. ábra. TCP/IP alapú alkalmazások

malizációs folyamatot nyílt módon elérje, vagyis úgy, mintha a szerver alkalmazási folyamattal együtt ugyanazon a gépen futna. Egy szemantiké képe ennek a 8.1. ábrán látható.

A *kliens oldali terminál használó* vagy *kliens alkalmazási folyamat* és a *szerver oldali szerver alkalmazási folyamat* az alkalmazási szolgáltató szolgáltatásainak igénybevételeivel tud egymással kommunikálni. E szolgáltatást *felhasználói kérések/válaszok* segítségével tudják igénybe venni. Az alkalmazási szolgáltató a kliens oldalon egy *kliens alkalmazási protokoll entitás*, szerver oldalon egy *szerver alkalmazási protokoll entitás* segítségével tudja a kéréseket kiszolgálni. Az egyes protokoll entitások az adott alkalmazási protokollnak a kliens gépben vagy szerver gépben implementált változatai. Az egyes protokoll entitások *portokon* keresztül érik el a *hoszt-hoszt szolgáltatót*, azaz a TCP vagy UDP protokoll alapján működő szolgáltatót. E szolgáltatót eléréséhez egységes formátumú (szabványos) kérések/válaszok szolgálnak. Az üzeneteket a szolgáltató *definiált szintaxisú adatokra* transzformálja, s ilyen formában továbbítja.

ISO alapú alkalmazási protokollok

Az OSI világban megkülönböztetnek valós környezetet és OSI környezetet. A **valós környezetben** valós eszközök (egyedi specifikációval rendelkező eszközök: terminálok és távoli folyamatok, fájlszerverek és kliensek, levelező szerverek és kliensek, stb.) kommunikálnak egymással. Itt minden valóságos, ahogy az az valós életben is előfordul. Az **OSI környezetben** elképzelt (virtuális) eszközök kommunikálnak egy-



ASP: Alkalmazás specifikus protokoll

ATP: Alkalmazást támogató protokoll

8.2. ábra. OSI alapú alkalmazások

mással egységesített, előre definiált formában. Itt minden képzetes, elképzelt. Erre éppen a **nyílt rendszer** koncepció miatt van szükség, hiszen egy rendszer akkor és csak akkor tekinthető nyíltnak, ha egy előre kialakított specifikáció-gyűjteménynek, szabályhalmaznak, szabványnak megfelel.

Az OSI környezetben az alkalmazási folyamatok közötti kommunikáció megvalósítása érdekében virtuális eszközt definiálnak. A **virtuális eszköz (virtual device)**¹ valójában egy adatszerkezet és az azon értelmezett operációk halmaza. Az adatszerkezet a valós eszköz legfontosabb általános tulajdonságait írja le, míg az operációk halmaza a valós eszközön végzett tevékenységek leírására szolgál. A virtuális eszköz segítségével a valós alkalmazásokat leképezik annak érdekében, hogy a nyílt rendszeren keresztül történő üzenetátvitel egységes kezeléssel történhessen, függetlenül az egyes valós eszközök sajátosságaitól. A virtuális eszköz adott esetben lehet *virtuális fájl tároló* egy fájl átviteli rendszernél, *virtuális terminál* egy terminálkezelő rendszernél, *virtuális levelező* egy elektronikus levelező rendszernél, *virtuális készülék* egy gyártásautomatizálási rendszernél, stb. A virtuális eszköz definiálásához szükség van a használatát lehetővé tevő szolgáltatói primitív halmaz definiálására is.

¹ A lektor megjegyzése: Ez a koncepció ma már általános a TCP/IP környezetben is. Erre két példa az Xwindow és a Web mint a legújabbkori virtuális terminál.

A virtuális eszköz fogalmának bevezetése lehetővé teszi azt, hogy a *kliens oldali és a szerver oldali alkalmazási folyamatok az alkalmazási szolgáltatóval egységes kérés/válasz* halmaz segítségével tartsák a kapcsolatot egy adott típusú alkalmazásnál, függetlenül az adott típusú alkalmazásnál használatos egyedi változatoktól, valós eszközöktől. Mivel azonban a virtuális eszközök a valós eszközök általános jellemzőit hordozzák csak, a valós eszköz és a virtuális eszköz között valaminek a segítségével leképezést kell végrehajtani. Az az elem, amely e leképezést elvégzi a *felhasználói elem* vagy *felhasználói ügynök (User Agent, UA)*.

Ahogy a 8.2. ábra is mutatja felhasználói ügynök nem része az alkalmazási szolgáltatónak, azon kívül álló elem. Rendszerint külön vagy az alkalmazási folyamathoz kapcsolva könyvtári eljárásokként, függvényekként implementálják, majd az alkalmazási folyamathoz kapcsolják. Ez teszi lehetővé azt, hogy a valós eszközöktől jövő *valós kérések* az alkalmazási szolgáltatóhoz már transzformált, *virtuális eszköz kérések* formájában érkeznek.

Mind a kliens oldali, mind a szerver oldali *alkalmazási funkcionális elem* két szolgáltatói elemet tartalmaz: egy alkalmazás specifikus (pl. fájl átvitelét vagy levelezést lehetővé tevő) szolgáltatói elemet, valamint egy alkalmazást támogató (pl. alkalmazási szintű összeköttetés vezérlő) szolgáltatói elemet. Ez azt jelenti, hogy az OSI alkalmazási rétege két alrétgre bontható: a felső alrét az alkalmazás specifikus szolgáltatói elemeket, míg az alsó alrét az alkalmazást támogató szolgáltatói elemeket foglalja magába. Az *alkalmazás specifikus szolgáltatói elem* az, amely a virtuális eszköz kéréseket lekezeli és a válaszokat generálja, az *alkalmazást támogató szolgáltatói elem* pedig az, amely az alatta levő megjelenítési réteg felé szolgáltatói primitívek segítségével szolgáltatói kérést jelent be, illetve onnan válaszokat fogad. A megjelenítési szolgáltató a *kliens oldali szolgáltatói ponton (PSAP)* és a *szerver oldali szolgáltatói ponton (PSAP)* keresztül tartja a kapcsolatot a megfelelő alkalmazási szolgáltatói elemmel.

Ha az alkalmazási folyamatok által elküldött adatok szintaxisa különböző, akkor megjelenítési réteg azokat *egységes átviteli szintaxisú* adatsorozattá konvertálja, így a hálózaton keresztül az adatok ilyen formában kerülnek továbbításra.

8.2 TCP/IP és OSI alkalmazások

A továbbiakban három fontos hálózati alkalmazást tekintünk át: a terminálkezelő, a fájlátviteli és az elektronikus levelező rendszereket. Bemutatjuk mindhárom rendszer funkcionális felépítését, legfontosabb tulajdonságait mind a TCP/IP alkalmazási protokollok, mind az OSI alkalmazási protokollok figyelembevételével. A TCP/IP protokoll készletből az alábbi alkalmazási protokollok kerülnek szóba:

TELNET

FTP

SMTP.

Tekintettel arra, hogy az előző fejezetekben az SMTP protokollokról már volt szó, itt csak hivatkozni kívánunk az ott elmondottakra.

Az OSI protokoll készlet alábbi alkalmazási protokolljaival kell foglalkoznunk a tekintett alkalmazásokkal kapcsolatban:

VT

FTAM

MOTIS.

8.2.1. Terminálkezelés

TELNET

A TELNET protokoll egy termiálhasználó vagy egy alkalmazási folyamat számára lehetővé teszi azt, hogy egy távoli gépen futó interaktív alkalmazási folyamatot elérjen, azzal együttműködjön. Egy ilyen interaktív folyamat lehet például egy szövegszerkesztő, amellyel a terminál használója úgy kommunikálhat, mintha mindketten ugyanazon a gépen futnának.

A TELNET háromféle szolgálatot nyújt használóinak. Az első egy szabványos interfész a távoli rendszerhez. Ennek neve *hálózati virtuális terminál* (*Network Virtual Terminal, NVT*). A kliens oldali alkalmazási programnak nem kell ismerni a távoli szerveren futó interaktív alkalmazási program részleteit, csupán csak azt az interfészt, ahogy az elérhető. Második szolgálata lehetővé teszi a kliens és a szerver számára az *opció egyeztetést*, vagyis azt, hogy szabványos parancsokkal és válaszokkal a két fél egyeztesse az adatcsere feltételeit, hogy mindkét oldalon azonos beállítások legyenek. A harmadik szolgálat a két oldal közötti *szimmetrikus kapcsolat* fenntartásáról gondoskodik. A TELNET ugyanis nem kényszeríti a klienst arra, hogy csak billentűzetről vigyen be adatokat vagy képernyőre vigye ki és jelenítse meg azokat, megengedi azt is, hogy a kliens a kliens gépen futó tetszőleges alkalmazási program legyen.

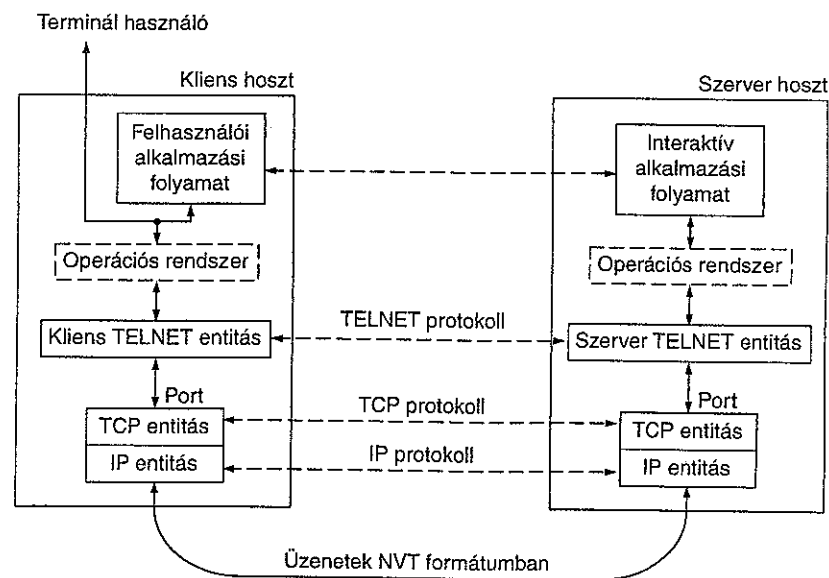
Ahogy az a korábbiakból ismert, a TCP/IP hálózatokban minden szerver alkalmazási folyamatnak van egy *saját címe*, amely két részből áll: a hoszt hálózati IP címéből és egy, a hoszton belüli helyi portszámából. Az *IP cím* használatos arra, hogy az IP protokoll alapján a routereken keresztül csomagok a hálózatban a megfelelő célhosztot megtalálják. A *portszám* szolgál a TCP vagy az UDP protokollt használva a hosztban annak a speciális alkalmazási folyamatnak az azonosítására, amely az adott portra fixen kapcsolódva a kliensek valamilyen speciális szolgálat elvégzésére vonatkozó kérését várja, s amelyen keresztül e szolgálatot nyújtja. Ezek általában ún. *jólismert címek*. TELNET esetén e jólismert cím: 23.

Amikor egy kliens alkalmazási folyamat *hívást kezdeményez* egy megfelelő szerver folyamat szolgáltatásainak igénybevételére, egy szabad portszámot vesz igénybe a kliens hosztban, majd a saját port és a kívánt TELNET szolgáltatáshoz tartozó szerver hosztban lévő fix, jólismert 23-as port között egy szállítási szintű összeköttetést kezdemé-

nyez. A szabad portnak választható portszám 1024-nél nagyobb számartományból választható. A szerver hoszt helyét a hálózatban a hoszt IP címe jelöli ki. Az összeköttetés létrehozásában a kliens hosztban lévő TCP entitás és a szerver hosztban lévő TCP entitás játszik szerepet, s ezen a szállítási összeköttetésen keresztül bonyolódik a terminál használója és a távoli interaktív alkalmazási folyamat közötti adatcsere.

Ahogy az a 8.3. ábrán látható, a terminál használója vagy a kliens oldali felhasználói alkalmazási folyamat a kliens TELNET entitást a helyi operációs rendszeren keresztül tudja elérni. A *terminál használója* vagy a *felhasználói alkalmazási folyamat* által kiadott parancsokat, adatokat a helyi *operációs rendszer* kezeli le és továbbítja a *kliens TELNET entitásnak*, amely a TCP megbízható összeköttetésen keresztül továbbítja a megfelelő *szerver oldali TELNET entitáshoz*. Ennek a feladata a felhasználótól érkező parancsok, adatok továbbítása a helyi operációs rendszeren keresztül az *interaktív alkalmazási folyamathoz*. E közvetítő tulajdonsága miatt a szerver oldali TELNET entitást szokták *pszeudo terminálnak* is nevezni. Az interaktív alkalmazási folyamat által kiadott adatok ugyanezen az úton-módon kerülnek vissza a kliens oldali terminálhoz és jelennek meg a képernyőn, vagy jutnak el a felhasználói alkalmazási folyamathoz.

A kliens oldalon és a szerver oldalon lévő TELNET entitás *parancsok és válaszok* segítségével kommunikál egymással. A kliens oldali TELNET entitás a terminálról jövő parancsokat és adatokat egy szabványos formátumú karaktorsorozattá konvertálja a hálózaton keresztül történő továbbítás végett. Mivel e parancsok szabványos formátummal rendelkeznek, ezért ezt a formátumot *hálózati virtuális terminálnak* (*Network Virtual Terminal, NVT*) nevezik. ASCII karaktereket használnak mind a pa-



8.3. ábra. TELNET terminál szolgálat

Parancs	Decimális	Jelentés
IAC	255	Értelmezd a következő bajtot parancsnak
EC	247	Karakter törlés
EL	248	Sor törlés
IP	244	Folyamat megszakítása
AO	245	Kivitel abbahagyása
SB	250	Opció megbeszélés kezdete
SE	240	Opció megbeszélés vége
WILL	251	Opció kérés
BRK	243	Kivitel megállítása
DMARK	242	Kivitel teljesítése
DO	253	Opció elfogadása

8.4. ábra. Néhány NVT parancssor

rancsok, mind az adatok továbbítására. Ha az átviendő adatok formátuma vagy ábrázolási formája eltér az adott gépben használt kód vagy adatábrázolástól, akkor a TELNET entitás *konvertálja* azokat a használatos ASCII kódú karakterekre.

Az NVT formátumban az összes parancs és adat a következőképpen néz ki. A hét bites ASCII kódú karaktereket 8 bites bajtokká konvertálják, ahol a legmagasabb helyiértéken lévő bit értéke $MSB = 0$. Minden karakter jelentése megegyezik az eredeti jelentéssel, még a vezérlő karakterek is. Például a decimális 7 értékű *BEL* (csöngető-jel) vagy a 12 értékű *FF* (új oldal) jelentése megmaradt. A *CR-LF* pár új sor kijelölésére szolgál. Az eredeti jelentésű vezérlő karakterek mellett egy kiegészítő parancs készlet is definiálásra került. Ez esetben az $MSB = 1$. Mivel az MSB eltér az egyszerű karakterektől, e parancsok könnyűszerrel megkülönböztethetők a szabványos ASCII karakterektől. Néhány ilyen parancs felsorolása és jelentése a 8.4. ábrán látható. E parancsok a távoli gépen futó interaktív alkalmazási folyamat vezérlésére szolgálnak. Ilyen például az *AO* parancs az aktuális kivitel megszakításának kérésére, vagy a *BRK* a kivitel felfüggesztésére, a *DMARK* a kivitel folytatására és befejezésére.

Az adatok átvitele normális esetben 7 bites ASCII karakterekben történik. Lehe-

Név	Kód	Jelentés
Bináris küldés	0	8-bites bináris adatok
Hurkolás (echo)	1	Vett karakter visszaküldése a küldőnek
Állapot (status)	5	A vevő oldali TELNET állapotának lekérdezése
Időzítő jel	6	Időzítő jel beszúrása a visszaküldött adatáramba
Terminál típusa	24	A távoli terminál típusának lekérdezése
Vonali mód	34	Teljes sorok küldése egyedi karakterek helyett

8.5. ábra. NVT opciók, kódjaik és jelentésük

tőség van 8-bites bajtokból álló füzerek (stringek) küldésére, továbbá bináris adatokat tartalmazó blokkok továbbítására is. Azért, hogy ezt meg lehessen tenni, a kezdeményező félnek **opció egyeztetésre** vonatkozó parancsot kell küldenie. Erre szolgál többek között az *SB* és az *SE* parancs. Néhány opció és kódja, valamint ezek jelentése a 8.5 ábrán látható.

Egy tipikus opció egyeztetés parancs például 8-bites bináris adatok átvitelének kérésére a következő parancssorozattal írható le:

IAC, SB, WILL, '0', SE

A visszajövő válasz pedig a felajánlott opció elfogadásáról:

IAC, SB, DO, '0', SE

Bináris átviteli módban a csupa 1-et tartalmazó adatbajjt kettőzve kerül továbbításra azért, hogy a vevő adatbajtnak és ne *IAC* parancsnak tekintse.

VT

Az ISO-OSI nemzetközi szabványként is elfogadott terminálkezelő protokollja a **virtuális terminál (Virtual Terminal, VT) protokoll**.

Míthogy a terminálok szabványosítása sikertelen kísérlet maradt, a különböző gyártók által előállított eszközök más-más ún. *ESC (escape) szekvenciákat* használnak a kurzor mozgatására, a különféle üzemmódok be/kikapcsolására. Ugyanígy sokféleség van a billentyűzet kialakítása, működtetése terén is. Azok a felhasználói programok, amelyek e beviteli-kiviteli eszközöket használják, igencsak elbonyolódnak, ha minden variációt figyelembe kívánnak venni.

A gyakorlatban többféle terminál létezik. A legrégebben használt típus például a **sorgörgető (scroll) üzemmódú terminál**, amely egy-egy karaktert kezel, a **lap (page vagy screen) üzemmódú terminál**, amely a teljes képernyőn lévő karaktereket kezeli, és az **úrlap (form) üzemmódú terminál**, amely a képernyő kiválasztott mezijén lévő szavakat/karaktereket kezeli. E típusokon belül is számos változat létezik.

A **virtuális terminál** fogalom bevezetése nagymértékben leegyszerűsíti e sokféleség kezelését. Ez valójában egy, a valós terminál absztrakt állapotát ábrázoló absztrakt adatszerkezet. Miközben a képernyőn az adatszerkezet aktuális állapotát látjuk, az adatszerkezetet a számítógép is és a billentyűzeten keresztül a felhasználó is változtatni tudja. A billentyűzetről az új bemeneti adatokat a számítógép lekérdezi, módosítja az adatszerkezetet, majd a kimenteti adatok megváltoztatása eredményeként a változás a képernyőn is megjelenik.

A **virtuális terminál (VT) protokoll** hasonló szolgáltatásokat nyújt a felhasználóknak, mint a TELNET. Lehetővé teszi azt, hogy a terminál használója, vagy a kliens oldali alkalmazási folyamat hozzáférjen egy távoli gépen, a szerveren futó másik alkalmazási folyamathoz, azzal információcsere bonyolítson le.

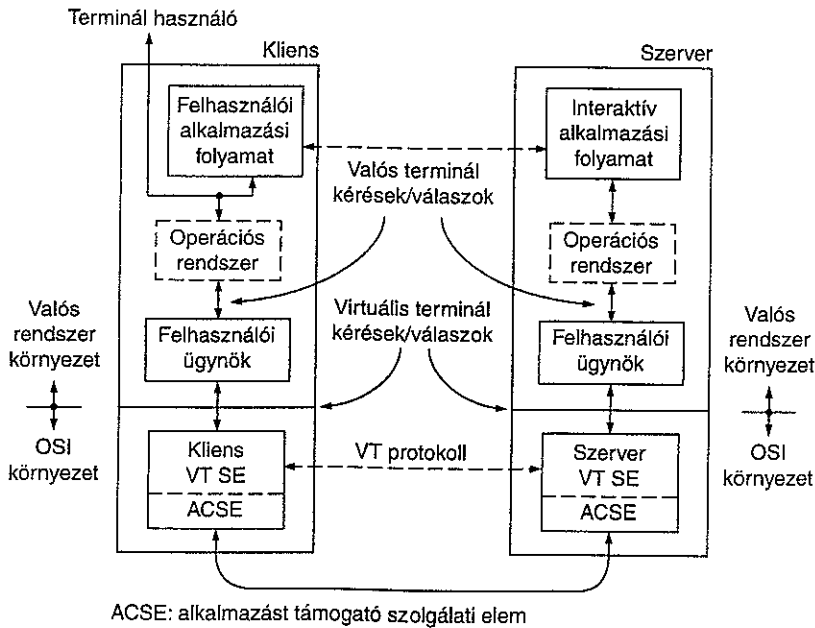
A TELNET-tel ellentétben azonban a *kliens VT entitás* az ún. **VT szolgálati elem**

(Service Element, SE) nem biztosít közvetlen terminál interfészt. Ezt a feladatot a felhasználói ügynök (User Agent, UA) látja el. A UA együttműködik a kliens VT szolgáltatási elemmel a virtuális terminál definiált tulajdonságainak megfelelően. Ha szükséges az UA lekéréseket hajt végre a két terminál tulajdonságai között. A virtuális terminál funkcionális vázlata a 8.6. ábrán látható.

A terminálok igen nagy változatossága miatt azonban nincs csak egyetlen virtuális terminál. Ehelyett a VT szolgáltatási elem *eszközt* ad két távoli felhasználónak ahhoz, hogy megbeszéljék, egyeztessék speciális, az alkalmazás által megkívánt virtuális terminál tulajdonságait. Mielőtt bármiféle együttműködésre sor kerülne, a két félnek meg kell egyeznie az ún. **virtuális terminál környezetben (Virtual Terminal Environment, VTE)**, amelyt az alkalmazáshoz fognak használni. Egyes esetekben csak néhány, más esetekben viszont nagyon sok ilyen opcionális paramétert kell egyeztetni.

Az egyeztetés megkönnyítése érdekében **szabványos opciókészleteket**, más néven **profilokat (Standard Profiles)** alakítanak ki a működési paraméterek, tulajdonságok egy-egy halmazából. Egy adott típusú egér vagy billentyűzet például egy-egy ilyen profillal jellemezhető. A két felhasználó vagy elfogadja az előre elkészített speciális profilt, vagy egyeztetésbe kezd a specifikus paraméterek megváltoztatására.

A két távoli VT felhasználó **osztott adatszerkezeten (Conceptual Communication Area, CCA)** keresztül kommunikál egymással. Mindkét VT szolgáltatási elem egy-egy ilyen különálló példányát kezeli a CCA-nak. A CCA tartalmának megváltozását a

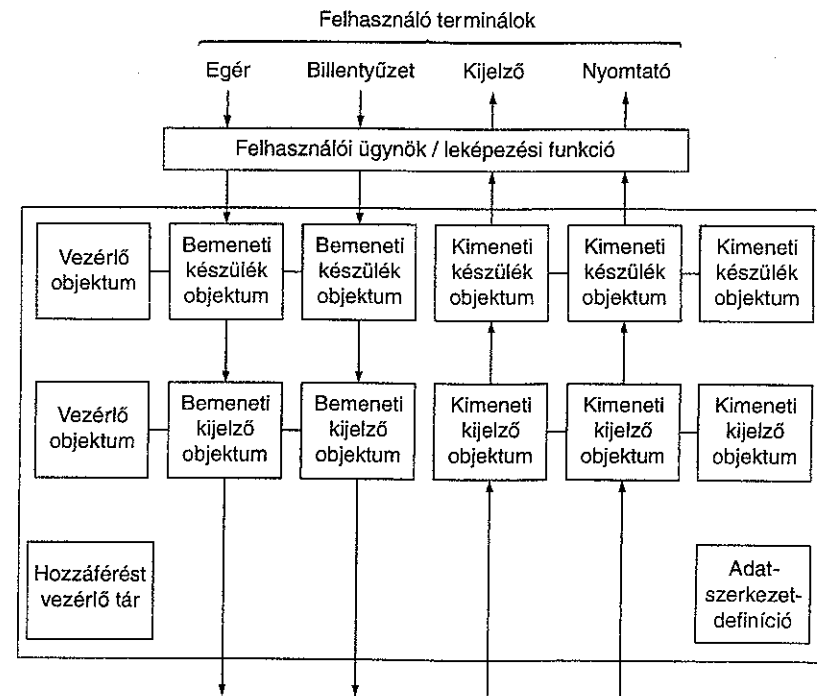


8.6. ábra. Virtuális terminál szolgálat

helyi felhasználói ügynök (UA) kezdeményezi egy megfelelő szolgálati primitívnek az interfészre történő kiadásával. A szükséges változtatások először a helyi CCA-ban mennek végbe, majd a VT protokoll segítségével egy VT PDU-ban a kérés átkerül a távoli rendszer VT szolgáltatási eleméhez, ahol a szükséges módosítás a CCA-n megtörténik, s egy szolgálati primitív segítségével az interfészen keresztül a távoli alkalmazási folyamathoz jut.

A CCA több **adatszerkezetet** tartalmaz. Ezek együtt írják le a használni kívánt virtuális terminál jellemzőit. A használt objektumok a 8.7. ábrán láthatók.

Minden beviteli és kiviteli készülékhez (billentyűzet, egér, kijelző, nyomtató, stb.) három objektum tartozik, amelyek az adott készülék tulajdonságait írják le. A **kijelző objektumok (display objects)** a valós eszközzel kapcsolatos eseményeknek a virtuális terminállal történő megjelenítését szolgálják. Ez egy háromdimenziós karaktertömbből áll, ahol a tömb minden eleme a virtuális terminálhoz kapcsolódó karakterkészlet egyetlen karakterét és annak attribútumait tartalmazza. Az attribútumok lehetnek: font, kiemelés foka, előtér színe, háttér színe. A kódszókészlet és az attributum kiválasztása az egyeztetés tárgya lehet. A **vezérlő objektumok (control objects)** azoknak a terminálkezelő funkcióknak, tulajdonságoknak a modellezésére szolgálnak, amelyeket normális esetben nem a felhasználók kezdeményeznek, amelyek nincsenek



8.7. ábra. VT protokoll adatszerkezetei

összefüggésben a szöveg megjelenítésével (pl. csengetőjel működtetése, lámpák be/ki-kapcsolása, funkcióbillentyűk és egérgombok modellezése). A **készülék objektumok (device objects)** a kapcsolódó valós készülék modellezésére szolgálnak. Minden készülékhez van egy ilyen objektum. Egy adott objektum bool-változók halmaza, s azt jelzi, hogy egy adott állapot van vagy nincs. Ez az interfész a valós eszköz és a virtuális eszköz között.

A VT protokoll a virtuális terminál szinkron és aszinkron működési módját támogatja. **Szinkron módú** működés esetén a bevitel és a kivetel az összeköttetés mindkét oldalán kombinálódik, azaz az egyik oldali változtatás a CCA-ban a másik oldalon is változtatást eredményez. Az adatok konzisztens kezelése érdekében a CCA-hoz kétoldaltól történő hozzáférést korlátozni kell. Erre vezérjel továbbításos (token passing) módszert vezettek be. A vezérjel tárolására egy **hozzáférés vezérlő tár (Access Control Store, ACS)** szolgál. **Aszinkron módú** működés esetén a bevitel és a kivetel szétválik, s ez lehetővé teszi azt, hogy mindkét oldalról egyidejűleg átvitelt kezdeményezzenek. Ilyenkor a CCA-t mindkét oldalon beviteli és kiviteli szempontból külön kezelik úgy, hogy az egyik oldali kimeneti adatstruktúra a másik oldali bemeneti adatstruktúrával van összekötve.

Az **adatszerkezet definíció (Data Structure Definition)** blokk a kijelző, a vezérlő és a készülék objektumokhoz kapcsolódó paraméterek típusának definícióját tartalmazza.

8.2.2. Fájlvitel

FTP

Egy fájl szerver szolgáltatásainak elérése nagyon sok alkalmazásban alapvető fontosságú. A fájlvitel egyik leggyakrabban használt hálózati alkalmazás. A TCP/IP architektúra fájlvitelt megvalósító protokollja a **fájlviteli protokoll (File Transfer Protocol, FTP)**. Feladata, hogy egy terminál előtt ülő felhasználó vagy egy felhasználói alkalmazási folyamat elérjen egy távoli gépen lévő fájlkezelő rendszert és együttműködjön vele.

Az FTP a TCP megbízható szállítási szolgáltatán alapulva az alábbi szolgáltatásokat nyújtja használóinak.

Interaktív hozzáférés. Jóllehet az FTP-t a legtöbb esetben programok használják, lehetőséget ad arra is, hogy emberi használatra alkalmas interaktív interfészen keresztül is hozzá lehessen férni a távoli szerverhez. Például a távoli fájlserver könyvtárában lévő fájlok kilistázhatók ily módon..

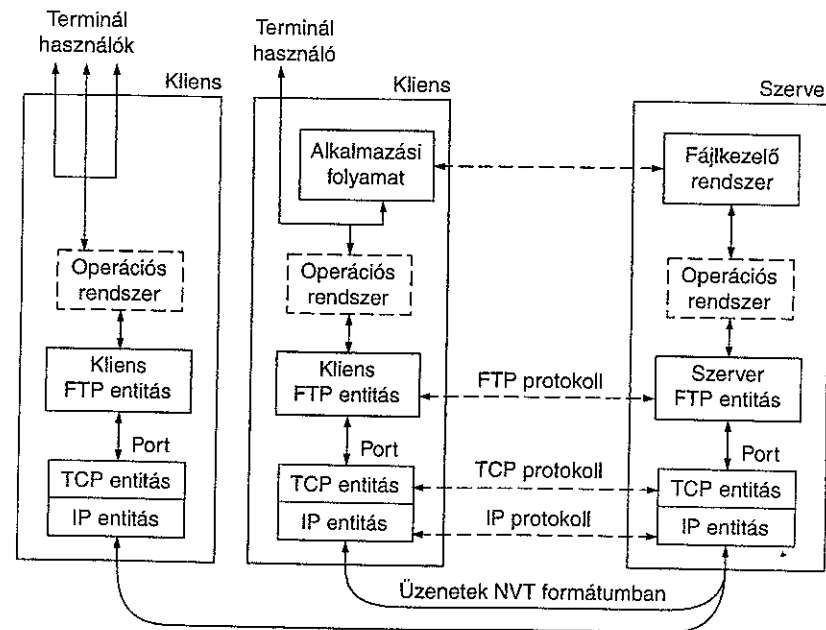
Formátum specifikáció. Lehetővé teszi a felhasználónak azt, hogy a kezelt fájl struktúráját és a tárolt adatok formátumát meghatározza. A lehetséges fájlstruktúrák a következők lehetnek: struktúrátlan fájl, stuktúrált fájl és véletlen hozzáférésű fájl.

Struktúrátlan fájl lehet például egy szövegfájl vagy egy bináris adatfájl. Ez a két FTP entitás között egy transzparens bitáram formájában megy át. **Struktúrált fájl** definiált típusú fix méretű rekordok sorozatából áll. A fájl átvitele ilyen fix hosszúságú blokkokban történik, de elképzelhető tömörített forma is. A **véletlen hozzáférésű fájl** változó hosszúságú rekordok sorozata. A rekord, amely az operatív tár egy lapja,

fejrézésében a hossz és típus információon kívül a rekord relatív elhelyezkedésére vonatkozó adat van. Átvitele változó hosszúságú blokkokban történik. Tömörítés itt is megvalósítható. A tárolt adatok formátumának meghatározása azt jelenti, hogy például definiálják azt, hogy egy fájl szöveget vagy bináris egészeket tartalmazzon, illetőleg egy szövegfájl az ASCII vagy a EBCDIC karakter készletet használja.

Hitelesség vizsgálat. A felhasználóknak, mielőtt egy adott fájlhoz hozzáférnek azonosítaniuk kell magukat a login név és jelszó elküldésével. A szerver az illetéktelen hozzáférési kísérletet visszautasítja.

Más szerverekhez hasonlóan az FTP szerver is lehetővé teszi a **többszörös hozzáférést**, azaz több kliens egyidejű kiszolgálását. A 8.8. ábra egy ilyen funkcionális modellt mutat. A kliens oldali FTP entitást elérheti egyrészt a terminál használója vagy használói, másrészt egy kliens oldali alkalmazási folyamat. A távoli szerveren elérhető szolgáltatások megegyeznek a szokásos fájlkezelő rendszerek szolgáltatásaival. Olyanok, mint könyvtári fájlok listázása, új fájlok létrehozása, fájlok tartalmának kinyerése, adatok frissítése egy adott fájlban, stb. A szerver FTP entitás egy jólismert címen (21-es port) várja a kliensektől jövő kéréseket. A kliensek egyidejű kéréseire a szerver is egyidejűleg többnek válaszol. Ahogy egy kérés megérkezett, a szerver FTP entitás kapcsolatba lép a szerver operációs rendszerén keresztül a helyi fájlkezelő rendszerrel, továbbítja a kliensek kérését feldolgozás végett úgy, mintha e kérések helyileg keletkeztek volna.



8.8. ábra. FTP fájlvitel többszörös kiszolgálással

A többszörös kérések kezelésére a szerver oldali FTP entitás egy új FTP protokoll-folyamatot hoz létre minden újonnan érkező kérés fogadására. Normális esetben egyetlen master folyamat van a jólismert portra kapcsolódva, amelyhez minden kérés befut. Ez a master vagy **vezérlő folyamat**. A vezérlő folyamat mindvégig létezik, és az adott kapcsolattal összefüggő különféle vezérlési feladatokat látja el. Ilyen például a jelszóval megerősített login eljárás és az átviendő fájlok struktúrájának és azok adattípusainak definiálása. Ez definiálja továbbá azt, hogy tömörítés kell-e, s ha igen, az milyen eljárást szerint történjen. Sok implementációban még egy másik folyamat is van az átviendő adatok kezelésére. Ezt az ún. **adattovábbító (data transfer) folyamatot** a vezérlő folyamat hozza létre, és csupán a fájlátvitel időtartamára létezik. Ilyen esetben egyetlen FTP összeköttetés két szállítási (TCP) összeköttetésen alapul: az egyik a vezérlőüzenetek, a másik a fájl tartalmának átvitelére szolgál. Természetesen mindez láthatatlan mind a kliens oldali, mind a szerver oldali felhasználó számára.

A két FTP vezérlő folyamat között áramló üzenetek formátuma *egységes szintaxis* alapján képződik, s a korábban megismert *NVT formátumú* azzal a különbséggel, hogy az FTP-nél opcióegyeztetés nem szükséges.

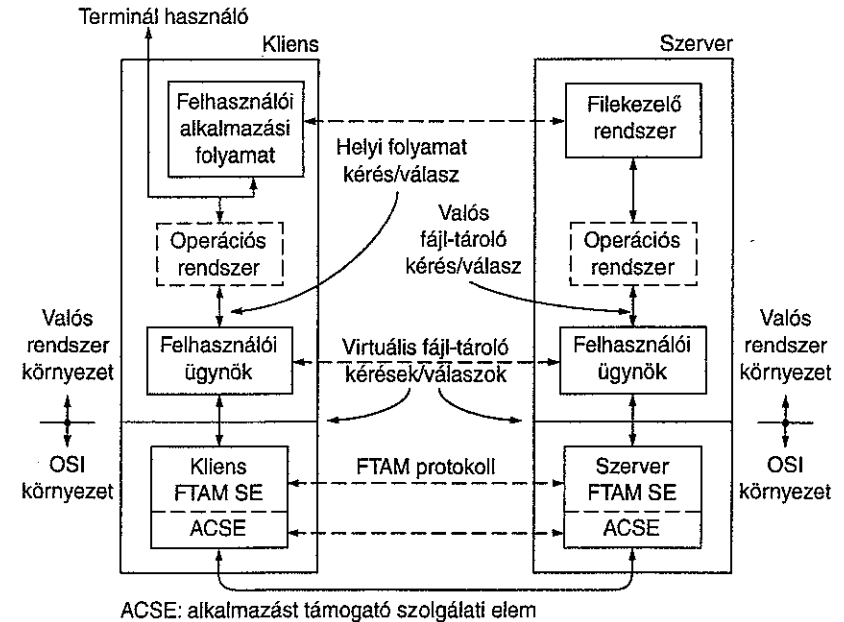
FTAM

Az ISO-OSI nemzetközi szabványként elfogadott fájlátviteli protokollja az **FTAM (File Transfer Access and Management)**.

Az FTAM az FTP-hez hasonló szolgáltatásokat nyújt felhasználóknak. Egy kliens gépen futó alkalmazási folyamat számára lehetővé teszi, hogy egy távoli szerver gépen futó fájlkezelő rendszerrel együttműködjön úgy, mintha maga is a fájlserver gépen futna.

Általánosságban egy fájlserver három tulajdonság jellemző: a fájlstruktúra, a fájlattribútumok és a fájlműveletek. Minden fájlserver valamilyen fájlstruktúrát kezel. A leggyakrabban alkalmazott fájlstruktúrák a következők. A *strukturálatlan* fájlstruktúra adatok strukturálatlan halmaza. Ezen az írás-olvasáson kívül semmilyen belső művelet nem végezhető. Az *egyszintű* (flat) fájlstruktúra rekordok rendezett sorozata. A rekordok hossza és típusa különböző lehet. Mivel a rekordok címkével rendelkeznek, a címke és a pozíció alapján egyedi rekordok kijelölhetők, kereshetők, rajtuk műveletek végezhetők. Egy UNIX állomány például egybájtos rekordok sorozatának tekinthető. A *hierarchikus* fájlstruktúra a leggyakrabban használt megoldás. Egy fa egyes csomópontjaihoz címke vagy egy adatrekord – esetleg mindkettő – tartozik. Egy adatrekord fabejárással a gyökértől a rekordhoz tartozó címkeig vezető úttal pozícionálható, rajta vagy vele különféle fájlműveletek végezhetők. Az **attribútumok** a fájlok fontos jellemzői. Minden fájl rendelkezik attribútumokkal. Olyan jellemzők szerepelnek itt, mint a *fájl neve, mérete, a létrehozás időpontja, a tulajdonos azonosítója, az engedélyezett műveletek, elérhetősége*, stb. A **fájlműveletek** a fájlok létrehozására, törlésére, kiválasztására, a fájlhoz való hozzáférésre, az adatok ki- és bevitelére, adatok mozgására szolgálnak.

Az FTAM funkcionális vázlatát a 8.9. ábra mutatja. A kliens oldali terminál használó vagy felhasználói alkalmazási folyamat a helyi operációs rendszer segítségével a felhasználói ügynök (UA) felé. Az UA e helyi kezdé-

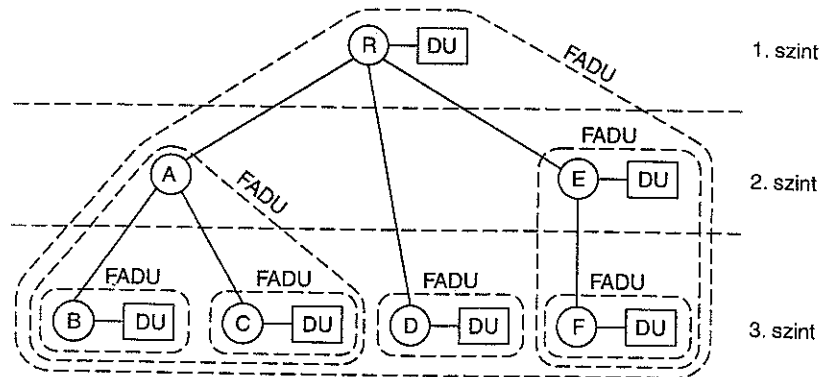


8.9. ábra. FTAM fájl átvitel

ményezést feldolgozza, és virtuális fájlátviteli kéréssé transformálva a kliens FTAM szolgálati elemnek (SE) továbbítja. A kliens FTAM SE az alkalmazást támogató szolgálati elem (ACSE) segítségével az alsóbb szint, a megjelenítési szinten keresztül egy összeköttetést kezdeményez a távoli szerver gép ACSE-vel. Ezen összeköttetés révén a kliens és a szerver FTAM SE között a virtuális fájlátviteli kérések átvitelre kerülnek, s a szerver oldali FTAM SE a saját felhasználói ügynökének továbbítja azokat. Az UA visszatranszformálja valós fájlátviteli kéréssé, s a szerver operációs rendszerének segítségével a fájlkezelő rendszerhez jut, amely kielégíti a beérkezett kérést.

Az FTAM fájlkezelő rendszere egy **virtuális fájlátviteli modell** (Virtual File Store Model) alapul. A modell gyakorlatilag megegyezik a hierarchikus szerkezetű fájlátviteli, felépítést a 8.10. ábra szemlélteti. A fájlstruktúra egy gyökér csomópontból, belső csomópontokból és levelekből áll. Mindezeket irányított élek kötik össze. Egy csomópont csak egyetlen szinthez tartozik. Minden csomópont lehetőséget ad egy alfa elérésére, amelyet **fájlérési adategységnek (File Access Data Unit, FADU)** neveznek. A fájlátviteli lévő fájl tartalma egy vagy több **adategységben (Data Unit, DU)** található. Legfeljebb egy DU csatlakozhat egy csomóponthoz. Ez azt jelenti, hogy a DU a FADU csomópont azonosításával elérhető. A FADU-n belül az egyes csomópontok a következő feltételezett sorrenddel érhetők el: R, A, B, C, D, E, F.

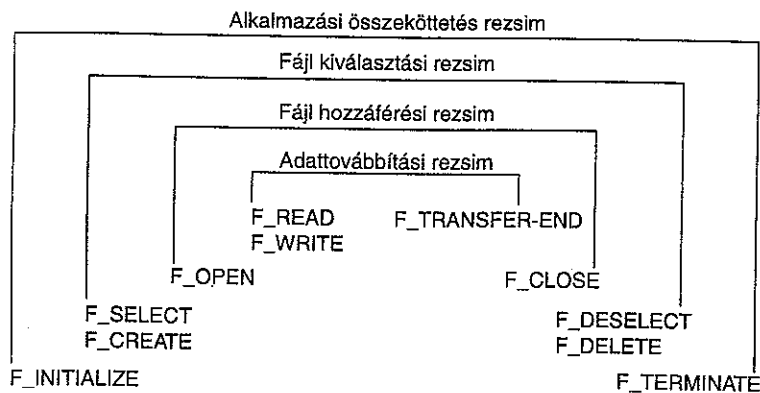
Egy DU *adattípussal* (skalár, vektor, halmaz) rendelkező objektum, amely adatok atomi elemeit, az ún. **adatelemeket** tartalmazza. Minden adatelemhez egy *absztrakt*



8.10. ábra. Virtuális fájlrendszer modell

szintaxis tartozik (karakter, oktet, egész, stb.). A DU-ban az összes elemre ez vonatkozik. Az elemek egymásközötti kapcsolata *faszerkezetű*, jóllehet más viszony is használható (pl. egyedi elemek, vektor). A fábejárás a szokásos sorrendben történik, s a kiolvasott adatok ilyen sorrendben kerülnek a megjelenítési entitáshoz. A megjelenítési entitás minden egyes elemet *önálló elemnek* tekint, s a megfelelő, egyeztetett *átviteli szintaxist* használva az *eredeti sorrend* megtartásával továbbítja azokat.

A fájlrendszer elvégzendő műveleteket megfelelő *szolgálati primitívek* segítségével lehet kiváltani. A primitíveket és a velük kezdeményezett szolgálatokat egymásbaágyazott csoportokba szervezik, és egy-egy ilyen csoportot *rezsimek* (*regime*) neveznek. A legkülső rezsim az *alkalmazási összeköttetés létrehozására* irányul. Ezen belül a *fájl kiválasztása*, majd a *fájlhoz való hozzáférés* kiváltása, legbelül a *fájlban lévő adatok továbbítása* található. Az egyes rezsimek és a köztük lévő kapcsolat a 8.11. ábrán látható.



8.11. ábra. FTAM szolgálati rezsimek

8.2.3. Elektronikus üzenetkezelés és levelezés

EMAIL

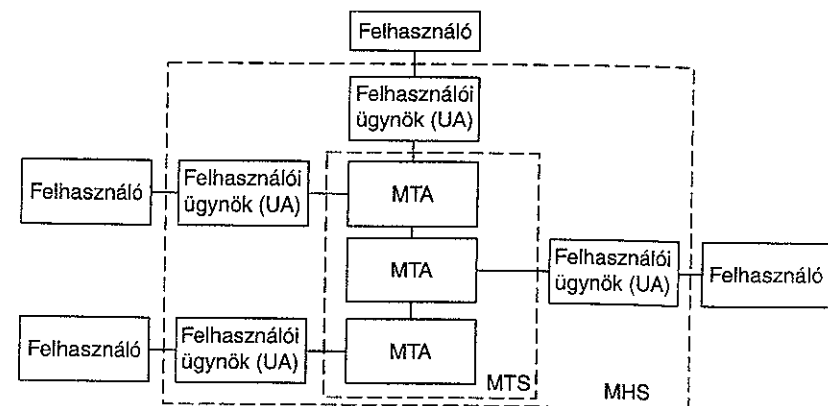
A TCP/IP alapú levelező rendszer és *egyszerű levelező protokoll* (*Simple Mail Transmission Protocol, SMTP*) ismertetése a 7.4 fejezetben megtalálható, az ott elmondottak nem kívánnak kiegészítést.

MOTIS

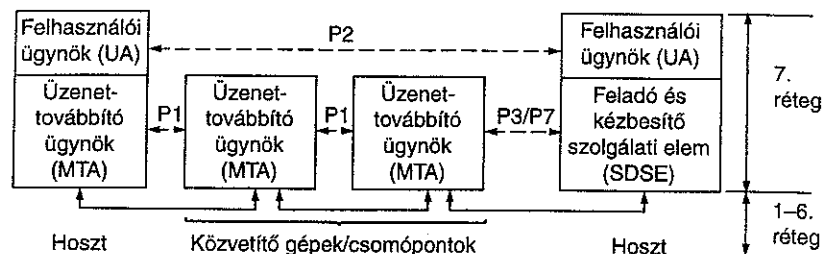
Az ISO-OSI levelezési és üzenetkezelési rendszere (**Message Oriented Text Interchange Standard, MOTIS**) a *CCITT X.400* ajánlásán alapul. Hasonló feladatokra alkalmas, mint a TCP/IP architektúrában az *SMTP protokoll*. Gyakorlatilag a MOTIS egy teljes üzenet és levéltovábbító rendszer, s nem egyetlen protokoll. Igen gyakran csak egyszerűen **üzenetkezelő rendszernek** (**Message Handling System, MHS**) nevezik.

A MOTIS egy nemzetközi üzenettovábbító rendszer, amelynek felépítése nagyon hasonló az általános postai levéltovábbító rendszerhez. Egy adott protokoll készletet használ az üzenetkezeléshez szükséges funkciók megvalósítása érdekében. A különféle entitások és a velük kapcsolatos protokollok a 8.12–14. ábrákon láthatók.

A rendszerhez kapcsolódó interfészhez egy terminál (rendszerint személyi számítógép, PC) kapcsolódik. Mint ilyen, megfelelő feldolgozási képességgel és megfelelő tárhelykapacitással rendelkezik ahhoz, hogy a felhasználó interaktív módon készíthessen üzeneteket (leveleket) vagy olvashassa el azokat, illetőleg tallózhasson a korábban beérkezett üzenetek között. Mindezek az ún. **felhasználói ügynök (UA)** feladatai közé tartoznak. Feladata továbbá az **üzenetátviteli rendszerhez (Message Transfer System, MTS)** történő csatlakozás, az interfész biztosítása annak érdekében, hogy más felhasználóknak üzenetet lehessen küldeni, illetve azokról üzenetet lehessen kapni



8.12. ábra. MOTIS üzenettovábbító rendszer struktúrája



8.13. ábra. MOTIS üzenetváltó rendszer protokolljai

vagy a kapott üzenetet megnézni. A felhasználói ügynök más, felhasználó által írt speciális feladatokat is elláthat. Mínt hogy az üzenetkezelő rendszert különböző típusú üzenetek továbbítására használják, például személyek közötti üzenetcsere lebonyolítására, az UA-nak képesnek kell lenni arra, hogy egy hasonló, távoli gépen lévő társ-UA-val kommunikáljon úgy, hogy az átvitt üzenetnek ugyanolyan jelentésűnek kell lenni mindkét terminálnál. Ez a szerepe a **P2 protokollnak**, amely **személyközi üzenetváltó (Interpersonal Messaging, IPM) protokollként** ismert.

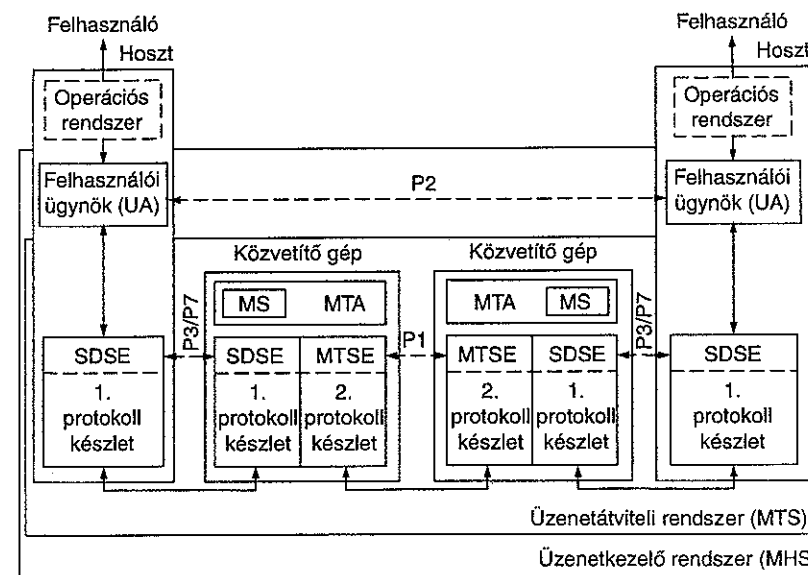
Amint az üzenet elkészült és az UA hozzáadta a saját protokoll vezérlő információját, azonnal továbbítódik a társentításhoz (szolgáltatási elemhez). Itt két lehetőség van. Ha a hosztban mind az UA, mind az **üzenetváltó ügynök (Message Transfer Agent, MTA)** megtalálható, akkor az UA az MTA interfészen keresztül az üzenetet az MTA-hoz kerül, majd az onnan a közvetítő gépeken keresztül a címzethez jut. Az MTA-k közötti forgalom a **P1 protokoll** segítségével valósul meg.

Abban az esetben, ha az UA és az MTA nem ugyanabban a gépben van, akkor az UA a megfelelő interfészen keresztül az üzenetet a **feladó és kézbesítő szolgáltatási elemnek (Submission and Delivery Service Element, SDSE)** adja, hogy továbbítsa az MTA-hoz. Az SDSE szerepe az, hogy ellenőrizze az üzenetek elküldését az azonos szinten lévő MTA-khoz, és vételét az MTA-któl. Az MTA-k itt úgy funkcionálnak, mint helyi postahivatalok. Az SDSE és a helyi MTA közötti forgalom ellenőrzésére a **P3 protokollt** használják. A P3 protokoll szabályokat tartalmaz az elküldés és a kézbesítés elvégzésével kapcsolatos teendők ellenőrzésére, valamint a számlázás elvégzésére.

Az UA-k globálisan egyedi nevek használatával kommunikálnak egymással. Az **üzenetkezelő rendszeren (Message Transfer System, MTS)** belül azonban teljes címetet használnak, megjelenítési szolgálatelérési pont (PSAP) címetet.² Így egy üzenet vétele esetén a helyi MTA először elvégzi a név-cím leképezést. A nevek és a címek szerkezetét a CCITT X.400 ajánlása definiálja. Az MTA-val kapcsolatban áll egy **katalógus kezelő ügynök (directory handling agent)**, amelyik elvégzi a leképezési munkát. Lényegében tehát, ha minden hoszt mondjuk egy X.25 hálózathoz kapcsolódik, akkor a címek az X.121 ajánlásnak megfelelő módon képződnek.

Amint az MTA megkapja a címeket, feldolgozza azokat úgy, hogy az UA-tól kapott üzenet fejrészébe, mintegy elektronikus borítékra, felírja a feladó és a címzett cí-

² A lektor megjegyzése: A teljes leképezési sor: Feladó/Címzett neve → Feladó/Címzett cím → PSAP cím → é.i.t.



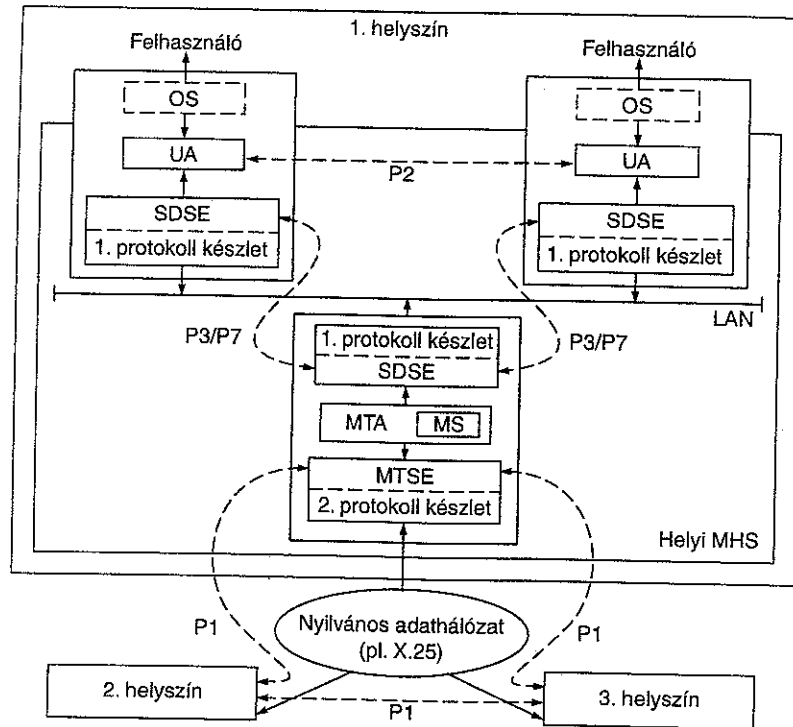
8.14. ábra. MOTIS üzenetváltó rendszer funkcionális modellje

mét a többi, az MTS számára fontos vezérlő információval együtt. Az üzenetet ezután a megfelelő **P1 protokoll** segítségével továbbítja a következő MTA-nak.

Amint a címzett UA-hoz tartozó MTA veszi az üzenetet, megkísérli kézbesíteni a terminálhoz tartozó SDSE számára egy **P3 protokoll** segítségével. Mivel azonban a felhasználói terminál normális esetben az MTA-tól távol van, elképzelhető hogy a terminált kikapcsolták, s így nem működőképes. Ilyen esetre az MTA egy **üzenet tárral (Message Store, MS)** rendelkezik, s ebbe tölti be a nem kézbesíthető üzenetet. Amikor a terminál ismét üzemképes állapotba kerül, a felhasználó az UA közbeiktatásával egy negyedik, **P7 protokoll** segítségével nyerheti ki az MS tartalmát.

Több helyszínen elhelyezkedő üzenetváltó rendszert tartalmazó nagyterjedésű üzenetkezelő rendszert mutat a 8.15. ábra. Az ilyen rendszernek gondoskodnia kell az egyes helyszínek (site) üzenetkezeléséről is, meg az egyes helyszínek közötti üzenetforgalom kezeléséről is. A közölt példán egyetlen helyszínen lévő több terminál (személyi számítógép) egy helyi hálózathoz kapcsolódik. Az egymásközötti üzenetforgalom a P2 protokoll segítségével bonyolódik. Amennyiben valamelyik terminálról egy távoli helyszíntre kell üzenetet küldeni, egy MTA-ra van szükség, amely ugyancsak a helyi hálózatra kapcsolódik. Az MTA és a terminál ezek után a P3/P7 protokollok segítségével kommunikálnak. Az adott helyszínen lévő MTA és a távoli MTA viszont a P1 protokollt használja kommunikációjához. Konkrét példánál maradva a P3/P7 lehet valamilyen LAN protokoll, a P1 pedig az X.25 protokoll, ISDN vagy más nagyterjedésű hálózati protokoll.

Az **üzenet formátumát** is a használt protokollok határozzák meg. A hálózaton egy



8.15. ábra. Többhelyszines üzenettovábbítás

üzenet borítékban kerül továbbításra. A *boríték* tartalmazza az üzenetet. A boríték tartalmazza az üzenetet. Az üzenet maga megint fejrészből és szövegrészből (test, body) áll. A *boríték*, amelyet az MTS használ, tartalmazza az üzenet azonosítóját, a feladó és a címzett címét, az üzenet tartalmára, titkosítására vonatkozó adatokat. IPM esetén az *üzenet fejrészt* az UA használja, s itt található az üzenet azonosítója, a feladó és a címzet neve, s más, az üzenet tartalmára vonatkozó jellemző. Az üzenet a felhasználónak szól, ő fogja feldolgozni. Az üzenet tartalmát hordozó adatok formátuma szabványos karakterkészlettel (ISO 646) vagy ún. nemzeti kódolású karakterkészlettel előállított szöveg, fax, továbbá Telex, Videotex vagy Teletex formátumú adat lehet.

9. Ajánlott olvasmányok és irodalomjegyzék

Befejeztük a számítógép-hálózatok tanulmányozását, de ez csak a kezdet. Sok érdekes témával nem tudtunk az őket megillető részletességgel foglalkozni, míg másokat teljes egészében kihagytunk helyhiány miatt. Ebben a fejezetben a további olvasáshoz ajánlott műveket és az irodalomjegyzéket tesszük közzé, azon olvasók kedvéért, akik folytatni szeretnék a számítógép-hálózatok tanulmányozását.

9.1. Javaslato a továbbolvasáshoz

A számítógép-hálózatok és elosztott rendszerek minden tekintetében bőséges irodalom áll rendelkezésre. Négy folyóirat, amely rendszeresen közöl cikkeket ebben a témában: az *IEEE Transactions on Communications*, az *IEEE Journal on Selected Areas in Communications*, a *Computer Communication Review* és a *Computer Networks and ISDN Systems*. Sok más folyóirat alkalmanként szintén közöl ilyen témájú cikkeket.

Az IEEE két folyóiratot ad ki, *IEEE Network Magazine* és *IEEE Communications Magazine* címmel, amelyek felméréseket, oktatási anyagokat és esettanulmányokat tartalmaznak a hálózatok témaköréből. Az előbbi az architektúra, szabványok és a szoftver területére helyezi a hangsúlyt, az utóbbi pedig a kommunikációs technológiákra (üvegszálak, műholdak stb.).

Ezekon kívül sok olyan évenkénti vagy kétévenkénti konferencia van, amelyekhez gyakran kapcsolódnak hálózatokkal és elosztott rendszerekkel foglalkozó cikkek. Ilyenek a *SIGCOMM' 9x*, a *The International Conference on Distributed Computer Systems*, a *The Symposium on Operating System Principles* és a *The N-th Data Communications Symposium*. Ezekon kívül az IEEE sokkötetnyi hálózati témájú cikket adott ki újra kényelmes puhakötésű formátumban.

Alább néhány javaslatot sorolunk fel kiegészítő olvasáshoz, a könyv fejezetei szerinti bontásban.

9.1.1. Bevezetés és általános művek

Bell, *Communications*

Egy kitűnő áttekintéshez a távközlésbeli irányzatokról, beleértve a telefont, az ATM-et, az ISDN-t, a drótnélküli LAN-okat, az Internetet és a személyhívókat, ez a cikk feltétlenül szükséges.

Comer, *The Internet Book*

Bárkinek, aki egy könnyen érthető bevezetőt keres az Internethez, érdemes ezt megnéznie. Comer leírja az Internet történetét, növekedését, technológiáját, protokollját és szolgáltatásait úgy, hogy a kezdők is megérthetik, de a könyv olyan sok anyagot tartalmaz, hogy a szakmaibb érdeklődésű olvasók érdeklődésére is számot tarthat.

Jabbari és mások, *Network Issues for Wireless Communication*

Ez a bevezetés a cellás rádiórendszerekhez felőleli a hívásirányítást, a forgalomirányítást, a jelzéseket és a korszerű mobil távközlési rendszerek más területeit.

Kwok, *A Vision for Residential Broadband Service*

Ha azt akarja tudni, hogy a Microsoft hogyan gondolja a video-on-demand szolgáltatás megszervezését, ez a cikk önnek szól. A Microsoft fő ATM tervezője magyarázza el cégének elképzelését. Röviden összefoglalva, a Microsoft ötlete: ATM az otthonokba, ez a követendő irány. Felejtsek el az összes „realisztikus” (vagyis alkalmi) megoldást, mint pl. az ADSL, és csinálják helyesen.

Le Boudec, *The Asynchronous Transfer Mode: A tutorial*

Az ATM egy feljövőben levő technológia, és ez a cikk alapos bevezetőt ad hozzá. Mind a fizikai, mind az ATM, mind az AAL réteget tárgyalja. Ezekon kívül az utolsó rész az ATM körüli vitáról szól.

Pahlavan és mások, *Trends in Local Wireless Networks*

A drótnélküli LAN-ok szerepe a jövőben minden bizonnyal növekedni fog. Ebben a cikkben a szerzők foglalkoznak a mostani helyzettel és az irányzatokkal a drótnélküli LAN technológiák és a frekvenciatartomány használatának területein.

Siu és Jain, *A Brief Overview of ATM*

Az ATM rendszerek sok jellegzetességével foglalkoznak ebben a bevezető jellegű cikkben, de a hangsúly a LAN emuláción és a forgalom menedzselésén van. Ugyancsak ez a cikk a bevezető a *Computer Communication Review* ATM technológiájáról szóló különkiadásához.

9.1.2. A fizikai réteg

Awdeh és Mouftah, *Survey of ATM Switch Architectures*

Akit érdekel az ATM kapcsolók tervezése, nézzen körül itt. Az általános, kapcsol-

lókról szóló bevezető és a puffereelési stratégiák után a szerzők sokfajta crossbar, disjoint-path és banyan elvű switchet tárgyalnak. A cikk több mint 200 hivatkozást is tartalmaz más cikkekre.

Bellamy, *Digital Telephony*

Ez a megbízható könyv tartalmaz mindent, amit valaha is tudni akart a telefonrendszerről, és még többet is. Különösen érdekesek az átvitelről és nyalábolásról, digitális kapcsolásról, üvegszálakról és az ISDN-ről írott fejezetek.

De Prycker, *Asynchronous Transfer Mode*, 2. kiadás

A 4. fejezet rengeteg információt tartalmaz az ATM kapcsolókról. Az elveket számos példakapcsolón keresztül mutatja be, többek között a knockout, Roxanne, Coprin és Athena kapcsolókon.

Held, *The Complete Modem Reference*, 2. kiadás

Minden elképzelhető tudnivaló a modemekről itt van, az amerikai és kanadai kormányzati megfelelőségi előírásoktól kezdve a modulációs technikákon és szabványokon keresztül egészen addig, hogy hogyan keressük meg egy beteg modem hibáját.

IEEE Communications Mag., 1995. január, *Wireless Personal Communications*

Ez a különkiadás hét cikket tartalmaz a drótnélküli személyi kommunikáció különböző területeiről. Együttesen felölelik a terjedést, hozzáférési módszereket, vevőelvetket, rendszerszempontokat és a hálózati kérdéseket.

Metcalfe, *Computer/Network Interface Design: Lessons from Arpanet & Ethernet*

Bár a mérnökök már több évtizede építenek hálózati interfészeket, mégis gyakran elgondolkozunk, tanultak-e valamit a gyakorlatból. Ebben a cikkben az Ethernet tervezője elmondja, hogyan építsünk hálózati interfészt, és mihez kezdjünk vele, ha már egyszer megvan. Nem köntörfalaz, elmondja, mit csinált rosszul, és azt is, mit jól.

Padgett és mások, *Overview of Wireless Personal Communications*

Egy bevezető a cellás és drótnélküli távközlő rendszerekhez, valamint a kettő összehasonlítása. Az amerikai és európai szabványokat is tárgyalja.

Palais, *Fiber Optic Communication*, 3. kiadás

Az üvegszálakról szóló könyvek általában a szakemberekhez szólnak, de ez felfoghatóbb, mint a legtöbb. Foglalkozik hullámterelőkkel, fényforrásokkal, fényérzékelőkkel, csatlókkal, modulációval, zajjal és még sok más témával.

Pandya, *Emerging Mobile and Personal Communications Systems*

Ha egy rövid és aranyos bevezető érdekli a kézbevehető személyi távközlési rendszerekről, ezt a cikket érdemes megnéznie. A kilenc oldal közül az egyik tartalmaz egy listát arról a 70 rövidítésről, amelyek a másik nyolc oldalon fordultak elő.

Partridge, *Gigabit Networking*

Számos ATM kapcsoló leírásán kívül az 5. fejezet összehasonlítja a kimeneti és bemeneti puffereket, és mindkettő teljesítményére képleteket vezet le.

Spragins és mások, *Telecommunications Protocols and Design*

A 2. fejezet egy jó bevezetőt tartalmaz az átviteli technológiákhoz, beleértve a rézdrótot, az üvegszálat, a cellás rádiót és a műholdakat. A Nyquist- és Shannon-korlátokat és következményeiket is bővebben tárgyalja.

9.1.3. Az adatkapcsolati réteg

Black, *Data Link Protocols*

Ez egy teljes könyv az adatkapcsolati rétegről. A gyakorlatra helyezi a hangsúlyt, sok anyagot tartalmaz a HDLC, LLC, PPP és más, kereskedelmileg fontos protokollokról.

Holzmann, *Design and Validation of Computer Protocols*

Az adatkapcsolati (és hasonló) protokollok formálisabb oldala iránt érdeklődő olvasó itt megtalálja, amit keres. Az ilyen protokollok specifikációját, modellezését, helyességét és tesztelését is tárgyalja ez a könyv.

Spragins és mások, *Telecommunications Protocols and Design*

Azon olvasók, akik a hibadetektáló és hibajelző kódokról többet akarnak tudni, nézzék meg e könyv 6. fejezetét. Ez az alapvető adatkapcsolati protokollok elveit is tartalmazza, ezen könyvhöz hasonló mélységben. A 7. fejezet tovább tárgyalja ezeket, és részletesen foglalkozik különféle adatkapcsolati protokollokkal.

Walrand, *Communication Networks: A First Course*

A 4. fejezet foglalkozik adatkapcsolati protokollokkal, a teljesítményképesség vizsgálatára helyezve a hangsúlyt. A véges állapotgép és Petri-háló alapú helyességmegközelítést szintén tárgyalja.

9.1.4. A közeghozzáférési alréteg

Abeyandara és Kamal, *High-Speed Local Area Networks and Their Performance*

Mivel a nagy sebességű LAN-ok éppen nagy sebességük miatt érdekesekek, jól jöhet egy cikk, amely a teljesítményüket vizsgálja és tárgyalja. E cikk középpontjában különféle busz, gyűrű, fa és csillag topológiájú LAN-ok állnak, valamint késleltetési és kihasználtsági jellemzőik.

Jain, *FDDI Handbook – High-Speed Networking Using Fiber and other Media*

Az FDDI alapos tárgyalásához (szép oktatási anyagokkal a száloptikához és a SONET-hez), jó választás ez a könyv. Az FDDI hardverről és szoftverről szóló hosszú

részeken kívül szól egy rész a teljesítményről és még tanácsot is ad üvegszálas kábel vásárlásához.

Perlman, *Interconnections: Bridges and Routers*

A hidak (és forgalomirányítók) hiteles, de szórakoztató tárgyalásához Perlman könyvét kell forgatni. A szerző tervezte az IEEE 802 feszítőfás híd algoritmusait, valamint a DECnet útirányító algoritmusait, és láthatóan szakértője a témának.

Stallings, *Local and Metropolitan Area Networks*, 4. kiadás

A könyv magját a három IEEE 802 LAN alkotja, de előfordul anyag más LAN-okról és MAN-okról is.

Walrand, *Communication Networks: A First Course*

Mint Stallings fenti könyve, az 5. fejezet tárgyalja az alapvető 802-es anyagot, meg az FDDI-t és a DQDB-t. A hangsúly a protokollok teljesítményképesség-vizsgálatán van.

9.1.5. A hálózati réteg

Comer, *Internetworking with TCP/IP*, 1. kötet, 3. kiadás

Comer alapművet írt a TCP/IP protokollkészletről. A 4–11. fejezetek az IP-vel és a hozzá kapcsolódó hálózati rétegbeli protokollokkal foglalkoznak. A többi fejezet első-sorban a felsőbb rétegekkel foglalkozik, ezeket is érdemes elolvasni.

Huitema, *Routing in the Internet*

Ha mindazt tudni akarja az internetes forgalomirányításról, amit csak lehet, akkor ez a könyv önnek szól. Mind a kiejthető algoritmusokat (pl. RIP, CDIR és MBONE), mind a ki nem ejthető algoritmusokat (pl. OSPF, IGRP, EGP és BGP) nagy részletességgel tárgyalja. Az újabb sajátosságok, mint a többesküldés, a mobil IP és az erőforrás-foglalás szintén megtalálhatók itt.

Perlman, *Interconnections: Bridges and Routers*

A 9. fejezetben Perlman sok kérdést körüljár az egyes- és többesküldéses forgalomirányító algoritmusok tervezése terén, úgy WAN-ok, mint LAN-okból álló hálózatok esetén, valamint különféle eszközökben történő megvalósításukat is leírja. A szerző láthatóan törődik a témával, mivel a 9.13.10. fejezet címe: „Saját véleményem az IP-alapú hálózati rétegbeli többesküldésről.”

Sterbenz és mások, *Report on the IEEE ComSoc Gigabit Networking Workshop*

Mielőtt a gigabites hálózatokat használatba lehetne venni, jó pár alapvető kérdést kell még megválaszolni. Egy kulcsfontosságú ezek közül az, hogy ezek a hálózatok mit fognak használni: ATM-et, TCP/IP-t vagy mindkettőt. Hogy a kérdést jobban megérthessük, az IEEE egy szakmai összejövetelt szervezett 1995 áprilisában. Ennek az összefoglalóját olvashatjuk itt. Schulzrinne kritikáját az ATM-ről bárkinek megéri elolvasni, aki azt hiszi, hogy az ATM a megoldás a világ távközlési problémáira.

Stevens, *TCP/IP Illustrated*, 1. kötet

A 3–10. fejezetek példákkal illusztrált összefoglaló tárgyalást adnak az IP-ről és az ide kapcsolódó protokollokról (ARP, RARP és ICMP).

Yang és Reddy, *A Taxonomy for Congestion Control Algorithms in Packet Switching Networks*

A szerzők gigondoltak egy rendszertant a torlódásvezérlő algoritmusokhoz. A fő kategóriák: a forrás által vezérelt nyílthurkú, a cél által vezérelt nyílthurkú, az explicit visszacsatolásos zárthurkú, és az implicit visszacsatolásos zárthurkú. Ezzel a rendszer-tannal írják le és osztályoznak 23 létező algoritmust.

9.1.6. A szállítási réteg

Comer, *Internetworking with TCP/IP*, 1. kötet, 3. kiadás

Mint fenn említettük, Comer alapművet írt a TCP/IP protokollkészletről. A 12. fejezet az UDP-ről, a 13. a TCP-ről szól.

Mogul, *IP Network Performance*

A cikk, címe ellenére, legalább annyira, ha nem még inkább, szól a TCP-ről és a hálózati teljesítményről általában, mint kifejezetten az IP teljesítményéről. Tele van hasznos útmutatókkal és ökölszabályokkal.

Stallings, *Data and Computer Communications*, 4. kiadás

A 12. fejezet a szállítási protokollokról szól. Kivonatolja az elméleti szolgáltatásokat és mechanizmusokat, valamint részletezi az OSI és a TCP szállítási protokollokat.

Stevens, *TCP/IP Illustrated*, 1. kötet

A 17–24. fejezetek példákkal illusztrált összefoglaló tárgyalást adnak a TCP-ről.

9.1.7. Az alkalmazási réteg

Anderson R., *Why Cryptosystems Fail*

Anderson szerint a banki rendszerek biztonsága gyatra, de nem azért, mert a ravasz behatolók feltörnek a DES-t PC-iken. A valódi problémák a becestelen alkalmazottaktól (egy bankhivatalnok a magáéra módosítja egy ügyfél postacímét, hogy elfogja a bankkártyát és a PIN kódot) a programozási hibákig (minden ügyfél ugyanazt a PIN kódot kapja) terjednek. Ami különösen érdekes, az a bankok válasza, amikor szembesítik őket a hibákkal: a rendszereink tökéletesek, ezért minden hibát az ügyfél okoz, vagy az csalásból ered.

Berghel, *The Client Side of the Web*

Egy könnyen érthető bevezető a Web-böngészőkhöz és a tulajdonságokhoz, amelyeket ezek támogat(hat)nak. A fő témák a HTML/HTTP szabványoknak megfelelés,

a teljesítmény, az újrakonfigurálhatóság, az együttműködés a munkaasztallal, és a segítségük a tájékozódáshoz. Ezen szempontok szerint kilenc népszerű böngészőt hasonlít össze.

Berners-Lee és mások, *The World Wide Web*

Egy áttekintés a Webről, és arról, hogy merre tart, attól az embertől, aki feltalálta. A cikk középpontjában a Web szerkezete, a HTTP és a HTML áll, valamint a jövőbeli irányzatok.

Carl-Mitchell és Quarterman, *Practical Internetworking with TCP/IP and UNIX*

Az 5. fejezet egy kellemes bevezetőt tartalmaz a nevekhez és a DNS-hez, beleértve a neveket kiadó hatóságokat, a működési felépítést és a DNS-adatbázist.

Choudbury és mások, *Copyright Protection for Electronic Publishing on Computer Networks*

Bár számos könyv szól a titkosítási algoritmusokról, csak kevés szól arról, hogyan lehet ezeket arra használni, hogy a felhasználókat az általuk megfejthető dokumentumok továbbadásában megakadályozzuk. Ez a cikk különféle mechanizmusokat ír le, amelyek segíthetnek a szerzői jog védelmében az elektronikus korban.

Furht és mások, *Design Issues for Interactive Television Systems*

A hálózati videózással kapcsolatban sok összetett technikai kérdés merül fel a rendszer felépítésével, a hálózat topológiájával, a kiszolgálók tervezésével és a végbe-rendezések tervezésével kapcsolatban. Ebben a cikkben a szerzők oktatási anyagot adnak néhány kulcsproblémával kapcsolatban, és néhány kivizsgálás alatt álló megoldást.

Handley és Crowcroft, *The World Wide Web – Beneath the Surf*

Míg a WWW-ről szóló könyvek 99 százaléka csak valamilyen böngésző használatát mondja el, vagy érdekes URL-eket sorol fel, addig ez elmagyarázza a Web belső működését. A kliens oldalt, a szerver oldalt és a HTML-t is elmagyarázza, emészthető méretű falatokban.

Kaufman és mások, *Network Security*

Ez a megbízható és gyakran szellemes könyv az első, ahova információért fordul az ember a hálózati biztonsággal kapcsolatban. Titkos és nyilvános kulcsú algoritmusok és protokollok, üzenet-hash-ek, hitelesítés, Kerberos és az elektronikus levelezés – mindről bőven találunk magyarázatot. A legjobb részek a szerzők közti (sőt, egy szerző közti) párbeszédnek, alsó indexszel jelölve, mint pl.: „Én₂ nem tudtam magamat, rávenni, hogy nagyon pontosan...”

Kumar, *Mbone: Interactive Multimedia on the Internet*

A könyv címlapján ez áll: „Fedezze fel, hogyan tudja sugározni, hirdetni és megjeleníteni a termékeit az Interneten.” Szerencsére erről a témáról máshol nem esik említés.

tés a könyvben. Ami viszont benne van, az az Mbone felépítése és megvalósítása, sok anyaggal arról, hogyan működik és hogyan használható.

Nemeth és mások, *UNIX System Administration Handbook*

A 16. fejezet egy hosszú bevezető a DNS-hez. Belemegy az apró-cseprő részletekbe, számos példával illusztrálva a különböző fájlkat és erőforrás-bejegyzéseket. A DNS szerverek menedzseléséhez szükséges programokat és más eszközöket is tárgyalja bizonyos részletességgel.

Rose, *The Internet Message*

Ha az elektronikus leveleit egy árnyalatnyi tekintélyrombolással tálalva szereti, ez a könyv önnek szól. A szerző nem áll annak fölötte, hogy időről időre hordószónokként hirdesse, mi a világ baja. Ha közelebbről nézzük, nem is rossz az ízlése.

Schneier, *Applied Cryptography*, 2. kiadás

Ez a monumentális összefoglalás az NSA legrosszabb rémálma: egyetlen könyv, amely minden ismert titkosítási algoritmust leír. Ami még rosszabb (vagy jobb, nézőpont kérdése), a könyv a legtöbb algoritmust (C nyelvű) futtatható programként tartalmazza. Ezen felül több mint 1600 hivatkozást is tartalmaz a titkosítási irodalomra. Ha valóban titokban akarja állományait tartani, olvassa el ezt a könyvet.

Steinmetz és Nahrstedt, *Multimedia: Computing, Communications and Applications*

Bár némileg kaotikus, ez a könyv valójában a multimédia sok területét felöleli. Hosszabban tárgyalt témái közé tartoznak: a hang, az állóképek, a mozgóképek, a tömörítés, az optikai tárolás, a multimédia operációs rendszerek, a hálózati kérdések, a hipertext, az adatfolyamok szinkronizálása és a multimédia-alkalmazások.

Van der Linden, *Just Java*

Amikor egy könyv első fejezete a következő címet viseli: „Jöjj a házamba, szólj a pók a légynek”, fogadni lehet arra, hogy az vagy egy gyermekmese, vagy a World Wide Web-ről szól. Ez a Hálóról szól, különösen a Java nyelvről és annak környezetéről. Azok kedvéért, akik a Javával akarnak játszani, a könyvet a teljes Java rendszer egészíti ki CD-ROM-on.

9.1.8. Osztott alkalmazások

A. S. Tanenbaum, *Computer Networks*, 2nd Edition, Prentice-Hall International, London, 1992; *Számítógép-hálózatok*. Novotrade Kiadó, Budapest, 1992.

Elsősorban az OSI alapú rendszereket, de a TCP/IP alapú rendszereket is tárgyalja. E rendszerek sok érdekes tulajdonságára rámutat.

F Halsall, *Data Communications*, Computer Networks and Open Systems, 3rd Edition Addison-Wesley, 1994

Az összes TCP/IP és OSI alapú alkalmazási protokollok leírását tartalmazza. A könyv jó áttekintő képet ad az egyes alkalmazásokról. Értékelni, összehasonlítani a könyv alapján könnyű.

W. Stallings, *Data and Computer Communications*. 5th Edition, Prentice-Hall, 1997

A TCP/IP alapú alkalmazások közül elsősorban a levelező SMTP protokoll és az FTP leírások szempontjából használható könyv.

D.E. Comer, *Internetworking with TCP/IP*. 3rd Edition. Vol.1, Prentice-Hall, 1995

Elsősorban a TCP/IP alapú alkalmazásokkal, alkalmazási protokollokkal foglalkozik. Apró részletek is megtudhatók a TELNET, az FTP és az SMTP protokollokkal kapcsolatban.

R.J. Cypser, *Communications for Cooperating Systems OSI, SNA, and TCP/IP*. Addison-Wesley, 1992

Széles látókört adó könyv, amely a különféle alkalmazásokkal is foglalkozik. Az X.400 üzenetkezelés jó leírását adja.

9.2. Irodalomjegyzék

Abesundara, B. W.–Kamal, A. E.: *High-Speed Local Area Networks and Their Performance*, *Computing Surveys*, vol. 23, pp. 221–264, June 1991.

Abramson, N.: *Development of the ALOHANET*, *IEEE Trans. on Information Theory*, vol. IT-31, pp. 119–123, March 1985.

Adam, J. A.: *Privacy and Computers*, *IEEE Spectrum*, vol. 32, pp. 46–52, Dec. 1995.

Adams, N.–Gold, R.–Schilit, B. N.–Tso, M. M.–Want, R.: *An Infrared Network for Mobile Computers*, *Proc. USENIX Mobile and Location-Independent Computing Symposium*, USENIX, pp. 41–51, 1993.

Anderson, R. J.: *Why Cryptosystems Fail*, *Commun. of the ACM*, vol. 37, pp. 32–40, Nov. 1994.

Armbruster, H.: *The Flexibility of ATM: Supporting Future Multimedia and Mobile Communications*, *IEEE Commun. Magazine*, vol. 33, pp. 76–84, Feb. 1995.

Armitage, G. J.–Adams, K. M.: *How Efficient is IP over ATM Anyway?* *IEEE Network Magazine*, vol. 9, pp. 18–26, Jan./Feb. 1995.

Arnold, K.–Gosling, J.: *The Java Programming Language*, Reading, MA: Addison-Wesley, 1996.

AT&T–Bellcore: *Observations of Error Characteristics of Fiber Optic Transmission Systems*, CCITT SG XVIII, San Diego, Jan. 1989.

Awdeh, R. Y.–Mouftah, H. T.: *Survey of ATM Switch Architectures*, *Computer Networks and ISDN Systems*, vol. 27, pp. 1567–1613, Nov. 1995.

Bakne, A.–Badrinath, B. R.: *I-TCP: Indirect TCP for Mobile Hosts*, *Proc. Fifteenth Int'l. Conf. on Distr. Computer Systems*, IEEE, pp. 136–143, 1995.

Balakrishnan, H.–Seshan, S.–Katz, R. H.: *Improving Reliable Transport and Hand-off*

Egyedi és összekapcsolt X.25 hálózatok

F.1. Bevezetés

Az X.25 hálózatokról egy nagyon rövid összefoglaló az 1.6.2. fejezetben található. Tekintettel azonban arra, hogy Európában, de különösen Magyarországon a nyilvános és magán adathálózatok nagyrésze ezt a technológiát használja, érdemes közelebbről részletesebben is megismerni az ilyen hálózatok jellemzőit.

Az itt leírtak az 5. fejezetben tárgyalt hálózati szintű működéssel vannak szoros kapcsolatban, annak kiegészítésére szolgálnak. Úgy lehet ezeket tekinteni, mint az internet és az ATM hálózatok mellett egy harmadik, nálunk széles körben használt adathálózat típus ismertetését.

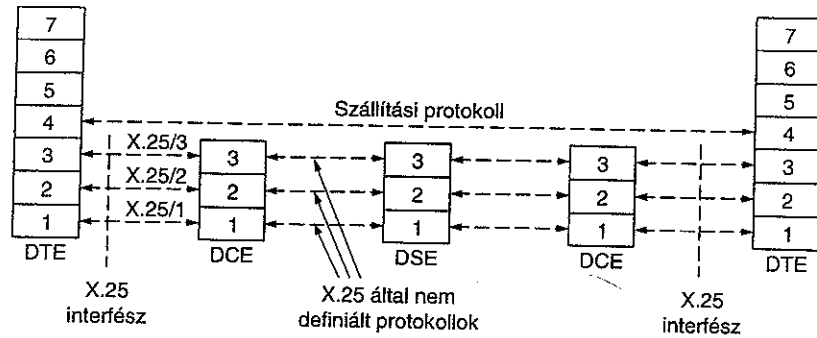
F.2. X.25 ajánlás

Csomagkapcsolt adathálózati szolgálat elérésére kidolgozott, nemzetközileg elfogadott megállapodás a CCITT X.25 ajánlása, amely egy **csomag-végberendezés (DTE)** és a **hálózat lezáró végpontja (DCE)** közötti *interfészt* definiálja. A DCE a szolgáltató oldali **hálózati csomópontot vagy routert (Data Switching Exchange, DSE)** elérő áramköröket zárja le, és a szolgáltató működéséhez szükséges jelkonverziókat végzi el. Valójában azonban az X.25 egy *protokoll készlet*, amely fizikai szinten a DTE és a DCE közötti bitforgalom, adatkapcsolati szinten az DTE és a DSE közötti keretforgalom, hálózati szinten a hálózat két végpontján lévő DTE-k közötti csomagforgalom szabályozását szolgálja. A következőkben a CCITT eredeti terminológiájához ragazkodunk, mivel adathálózati körökben ezek terjedtek el.

Az X.25 interfész (CCITT értelemben az a felület, amelyen keresztül a hálózati szolgáltató szolgálatai elérhetők, azaz itt a DTE és a DCE határfelülete) és a kapcsolódó protokollok helyét az OSI protokollhierarchiában az F.1 ábra mutatja.

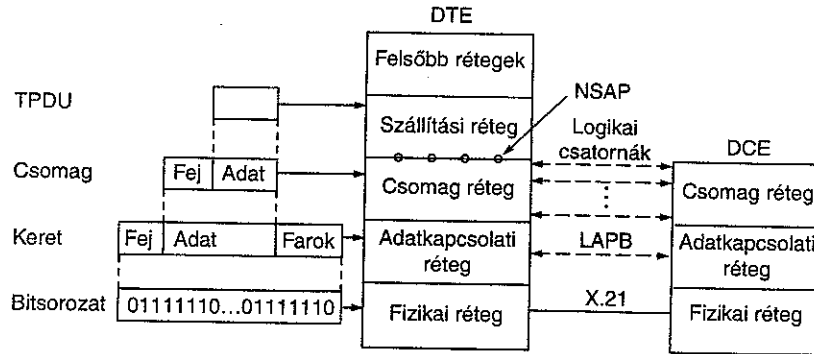
Ahogy az ábráról is látható az X.25-nek lokális jelentősége van, mivel csak a DTE és DCE közötti kommunikációval foglalkozik, s nem specifikálja a csomópontok közötti kommunikáció szabályait, a DTE–DTE közötti kommunikációt pedig a magasabb szintű szállítási protokoll specifikálja.

Az utóbbi években széles körben elterjedt, legtöbbször csak X.25-ös hálózatnak nevezett csomagkapcsolt adathálózat felhasználói vagy más néven *előfizetői inter-*



F.1. ábra. X.25 interfész és protokollok

fészét tehát a CCITT X.25 ajánlása definiálja, amely három réteget tartalmaz. A legalsó réteg a X.21 ajánlást foglalja magába a fizikai interfész definiálására a DTE és a szolgáltató által támogatott DCE között. Az adatkapcsolati réteg protokollja a HDLC egy részhalmaza a LAPB, amely hibamentes adattovábbítás szolgálatot nyújt a felette levő hálózati rétegeknek az alatta levő fizikai réteg szolgáltatásainak felhasználásával. Az itt csomag rétegek nevezett hálózati réteg pedig megbízható adattovábbító szolgálatot nyújt a felette levő szállítási réteg számára az ottkézelt TPDU-k továbbítására. Az egyes rétegekben megjelenő PDU-k és a köztük lévő relációk az F.2. ábrán láthatók. A továbbiakban megvizsgáljuk az egyes rétegeket részletesebben is.



F.2. ábra. Az X.25 protokolljai és adatformátumai

F.2.1. Fizikai réteg

Az X.25 fizikai rétege a DTE és a DCE közötti interfész mechanikai, villamos funkcionális és eljárásbeli vonatkozásaival foglalkozik. Ezeket ténylegesen nem az X.25, hanem az X.21 ajánlás definiálja, az X.25 csak hivatkozik rá. Az X.21 duplex bitsoros

szinkron digitális átviteli utat biztosít a DTE és a helyi DSE között. Az adatsebesség 600 b/s-tól 64 kb/s-ig változhat. Annak érdekében, hogy a még nagyszámban használatos analóg berendezések is csatlakoztathatók legyenek az X.21 bis ajánlás használatos, amely az RS 232/V.24 ajánlás egy részhalmaza.

F.2.2. Adatkapcsolati réteg

Az X.25 adatkapcsolati rétege a HDLC protokoll egy részhalmaza a LAPB protokollt használja. A HDLC különféle működési módokat definiál. Ezek közül egyik az ún. aszinkron kiegyenlített mód (Asynchronous Balanced Mode), amely a LAPB protokoll alapja. E működésmód kétpontos topológiában szimmetrikus, duplex összeköttetést definiál, amelyben mind a DTE, mind a DCE lehet egyidőben elsődleges és másodlagos is. Ennek megfelelően mind a DTE, mind a DCE bármikor kezdeményezhet forgalmat, küldhet parancsokat és fogadhat válaszokat. A keret címezője a megszo- lított másodlagos partner címét tartalmazza. A keret adatmezőjében a csomag rétegben előállított csomagok kerülnek továbbításra.

F.2.3. Csomag (hálózati) réteg

A csomag réteg az OSI hivatkozási modell hálózati rétegének felel meg. A szállítási réteg a hálózati réteg szolgáltatásait a hálózati szolgálatelérési ponton (NSAP) keresztül éri el. Ez a pont a hálózaton belül egy címmel adható meg. A csomag réteg működésének előfeltétele az, hogy az egyes szakaszokon az átvitel biztonságosan megtörténjen. A csomag réteg logikai csatornákat kezel. A DTE ebben a rétegben adja meg a DCE-nek az összeköttetés felépítéséhez szükséges vezérlő információt, amely alapján a DCE, a saját forgalomirányító tábláját felhasználva, kiválasztja szomszédját. Az ilyen elemi összeköttetések, logikai csatornák sorozatából azután kialakul a kezdeményező DTE és a címzett DTE közötti virtuális áramkör.

A csomag szinten kezelt PDU-k teszik lehetővé a virtuális áramkörök felépítését, azok működését, majd bontását. A két DTE közötti virtuális áramkör kialakításához logikai csatornákat használnak. A virtuális áramkör kapcsolt (SVC) vagy állandó (PVC) lehet. Kapcsolt esetben minden új igény esetén az összeköttetést fel kell építeni. Állandó esetben az összeköttetés állandóan fennáll, sem felépíteni, sem lebontani nem kell. A logikai csatorna a DTE és a DCE vagy két DCE között létezik, és foglalt vagy szabad állapotú lehet. Foglalt, ha egy adott sorszámú csatornát valamely fél valamely virtuális áramkörhöz hozzárendelte, szabad, ha ilyen hozzárendelés még nem történt meg. A logikai csatornák száma elméletileg 4096, mégpedig 16 csoportban csoportonként 256 logikai csatorna formájában. A virtuális áramkör tehát ilyen logikai csatornák sorozataként alakul ki.

Csomagformátum

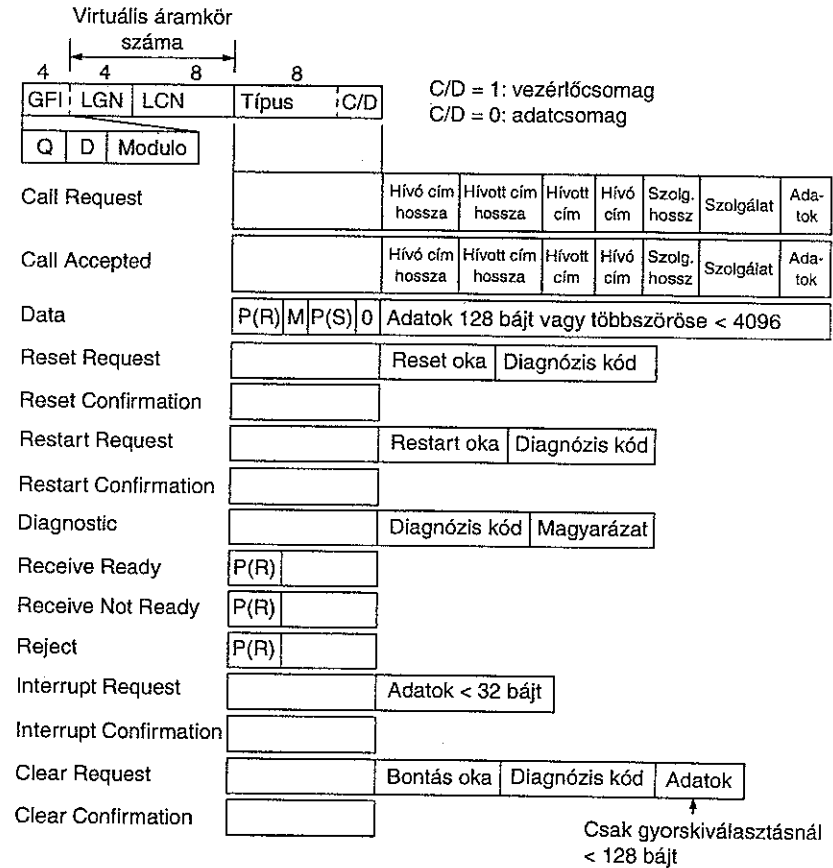
Különböző PDU-k felsorolása és használata az F.3. ábrán látható. A DTE és a DCE közötti párbeszédben használatos PDU nevek az összeköttetés két oldalán különböző nevek szerepelnek. Mindegyik PDU-nak fix fejrésze van, amely a *csoporthatár azonosítót* (*Group Format Identifier, GFI*), a *logikai csoport számot* (*Logical Group Number, LGN*) és a *logikai csatorna számot* (*Logical Channel Number, LCN*) tartalmazza (F.4. ábra). A 4 bites *GFI* az adat-minősítő *Q* bitet, a kézbesítés megerősítésére szolgáló *D* bitet és két *modulo* bitet tartalmaz. Az adatcsomag fejrészeiben szereplő *Q* bit a végfelhasználó szempontjából minősítheti a csomagot, jelezvén, hogy a csomag adatmezőjében lévő TPDU vajon vezérlő- vagy adat-információ-e. A *D* bit beállításával jelzi a kezdeményező DTE, hogy az elküldött csomagra jövő választ a helyi DCE-től (a kézbesítés nem garantált, $D=0$) vagy a távoli DTE-től várja (a kézbesítés garantált, $D=1$). A *modulo* bitek a forgalom szabályozás szempontjából érdekes sorszámozásra szolgálnak: a csomagok modulo 8 vagy modulo 128 sorszámozása lehetséges. Az *LGN* és az *LCN* együttesen a *virtuális áramkör azonosítóját* (*Virtual Circuit Identifier, VCI*) adja. A logikai csatornaszám rendelhető hozzá a csomagot egy bizonyos logikai csatornához, és teszi lehetővé a vonal többszörös kihasználását (multiplexelést). Az SVC esetén minden összeköttetés felépítésénél új csatornaszámot adnak ki, míg a PVC esetén ez a csatornaszám fix. A logikai csatornákon egymástól függetlenül egyidejűleg több, DTE-k közötti összeköttetés is kialakítható. Tekintettel arra, hogy mind a DTE, mind a DSE logikai csatornaszám kezelést végez, az ütközések elkerülése érdekében a logikai csatornák lefoglalása nem azonos módon történik. A DTE felőli kezdeményezés esetén a legmagasabb, a DSE felőli kezdemé-

Csomag (PDU) típus		
DTE → DCE	DCE → DTE	Feladata
Call Request Call Accepted	Incoming Call Call confirmation	Összeköttetés létesítése
Clear Request DTE Clear Confirmation	Clear Indication DCE Clear Confirmation	Összeköttetés bontása
DTE Data Interrupt Request	DCE Data Interrupt Confirmation	Adatátvitel
DTE Receiver Ready DTE Receiver Not Ready DTE Reject Reset Request DTE Reset Confirmation	DCE Receiver Ready DCE Receiver Not Ready Reset Indication DCE Reset Confirmation	Forgalom szabályozás
Restart Request DTE Restart Confirmation	Restart Indication DCE Restart Confirmation	Újraszinkronozás
Diagnostic	Diagnostic	Hálózati hibák jelentése a felhasználóknak

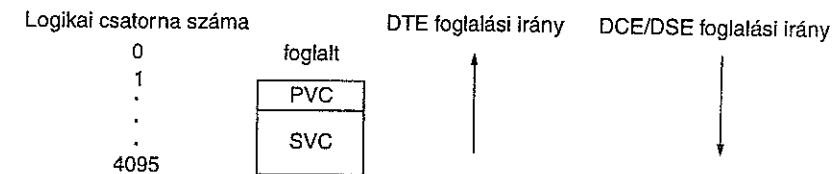
F.3. ábra. Csomagtípusok

nyezés esetén a legalacsonyabb csatornaszámnál kezdődik a kiosztás, ahogy azt az F.5 ábra mutatja.

A csomag következő mezője a csomag *típusát* jelöli. A vezérlő csomagokban a mező LSB bitjének értéke 1, míg az adatcsomagokban itt 0 szerepel.



F.4. ábra. Csomagformátumok



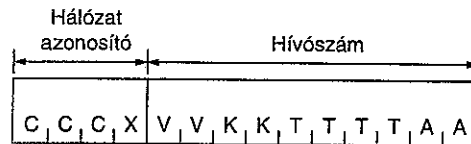
F.5. ábra. Logikai csatornaszám kiosztása

Az összeköttetést kezdeményező és elfogadó (*Call Request* és *Call Accepted*) csomagokban a következő két mező a hívott és a hívó DTE címének hosszát adják meg. Mindkét címhosszat 4-biten lehet megadni. A következő mezők a címzett és a küldő DTE hálózati címét (NSAP) tartalmazza.

A nemzetközi hálózatokban használatos NSAP címeket a CCITT X.121 ajánlás definiálja. Az NSAP címnek globálisan egyedi címnek kell lennie. Az ISO és a CCITT által felügyelt, nemzetközileg használható címek hierarchikus struktúrát követnek. A teljes cím 14 decimális számjegyből áll (F.6. ábra). Első négy számjegy a hálózat azonosítója (*Data Network Identification Code, DNIC*). A DNIC első három számjegye az országkód (CCC). Magyarországra például a CCC=216. A következő számjegy a különféle hálózattípusok megkülönböztetésére szolgál (X). A nyilvános csomagkapcsolt hálózat esetén X=1. A fennmaradó 10 számjegyre az alábbi kiosztás honosodott meg:

- két számjegyen a virtuális magánhálózat azonosítója szerepel (MATÁV-nál ez 50)
- két számjegyen a hivatkozott (telefon)körzet száma
- négy számjegyen a terminál azonosító száma
- két számjegy alácímzésre, címkitérésre szolgál, de általában nem használatos.

Az adatcsomagban és három másik vezérlő csomagban (*RR, RNR, REJ*) a típusmező vételi sorszámot tartalmaz a forgalom szabályozás érdekében. Az adatcsomagban ezen kívül adási sorszám és egy *M bit* is található. Az adatmezőben maximálisan 128 bájtnyi felhasználói adat lehet, jóllehet egyes szolgáltatók ennél nagyobb hosszúságú adatmezőt is használnak (4096). A maximális csomaghosszt (pl. 128 bájtt) meghaladó méretű adatblokkok átvitele esetén az adatblokkot több csomagban viszik át. Ilyenkor az *M=1* jelzi azt, hogy további összetartozó adatcsomagok következnek. Az utolsó adatcsomagban *M=0* kerül beállításra. Az *M bit* szolgál a vevőben a csomagok összefogására, az adatállomány kialakítására.



CCC : országkód
 X : hálózat kódja
 VV : virtuális magánhálózat kódja
 KK : körzetszám
 TTTT: csomagterminál azonosító
 AA : alácímzés, címkitérés

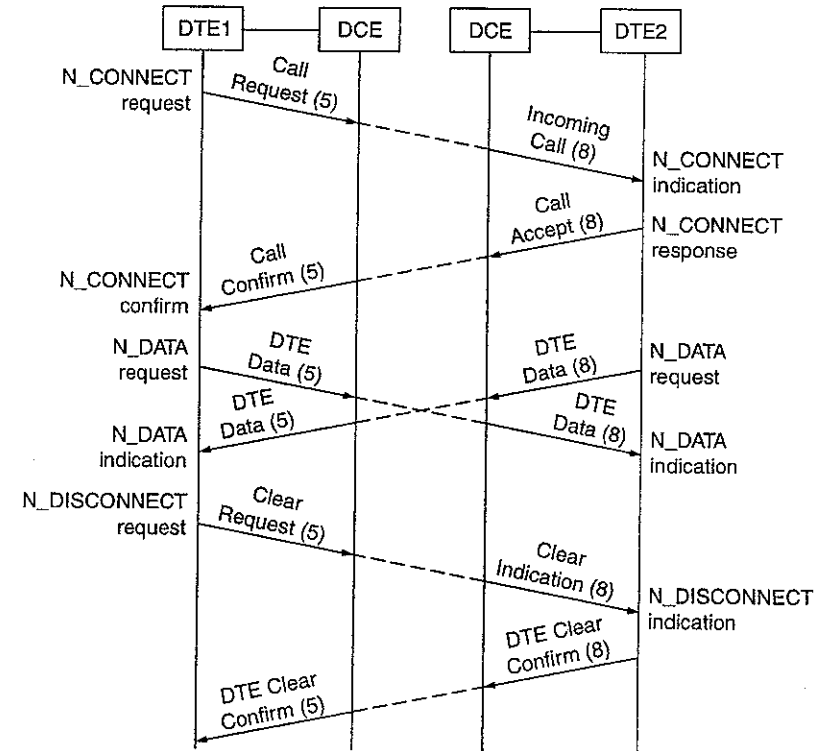
F.6. ábra. Hálózati címszerkezet

Legfeljebb 32 adatbájt felhasználói adatot lehet küldeni a megszakításkérés tartalmazó PDÜ adatmezőjében is, illetve gyors kiválasztás esetén az összeköttetést kérő csomagban is 16 max. 128 bájttal terjedelemben.

Az összeköttetést kezdeményező és elfogadó csomag *szolgálat* mezője a kiválasztott és megbeszélésre elküldött működési paramétereket tartalmazza. Ilyenek lehetnek például a kiterjesztett sorszámok használata, nem szabványos ablakméret beállítása, nem szabványos csomaghossz beállítása, fordított díjszabás kérése/elfogadása, ARQ típus kiválasztása (n-nel visszalépés, szelektív ismétlés), gyors kiválasztás használata és mások.

A virtuális áramkör felépítése és lebontása

A virtuális áramkör felépítésének és lebontásának folyamata az F.7. ábrán követhető végig. A DTE1 egy *N_CONNECT.request* primitívvel kezdeményezi az összeköttetés felépítését az NSAP-on keresztül. Ennek hatására egy *Call Request* csomag generálódik



F.7. ábra. Virtuális áramkör felépítése és lebontása. Adattovábbítás

dik egy szabad logikai csatorna számmal (pl. 5). A DCE a beérkező kérést továbbadja a szomszédjának az általa kiválasztott szabad logikai csatorna számmal. Végül az utolsó csomópont DCE-jétől a DTE2-höz egy *Incoming Call* csomagként érkezik meg az utolsó DCE által adott logikai csatornaszámmal (pl. 8)¹. DTE2 a saját NSAP-ján *N_CONNECT.indication* primitíven keresztül értesíti használóját a beérkezett hívásról, aki *N_CONNECT.response* primitívvel válaszol. Ez egy *Call Accepted* csomag formájában 8-as logikai csatornaszámmal továbbítódik a DTE2-től a DCE-hez, amely szomszédjának továbbítja ugyanazon az útvonalon ahogy a kérés érkezett. Ennek megfelelően, a lánc végén lévő DCE 5-ös logikai csatornaszámmal fogja továbbítani a kezdeményező DTE1-nek a *Call Accepted* csomagot. DTE1 ekkor a saját NSAP-ján keresztül értesíti használóját az összeköttetés létrejöttéről, aki ezután adatok továbbítását fogja kérni *N_DATA.request* primitív segítségével. DTE1 ezt 5-ös logikai csatornaszámmal továbbítja a szomszéd DCE-nek egy *DTE Data* csomagban, s a lánc végén a *DTE Data* csomag 8-as logikai csatornaszámmal kerül továbbításra a DTE2-höz, aki *N_DATA.indication* primitíven keresztül értesíti használóját a beérkezett adatokról. Mivel a virtuális áramkör duplex szolgáltatást enged meg, egyidejűleg a másik irányban is hasonló folyamat mehet végbe ugyanezekkel a logikai csatornaszámokkal. Az összeköttetés bontása *N_DISCONNECT.request* primitív kibocsátásával kezdeményezhető. Ennek hatására *Clear Request* csomag generálódik a DTE1-ben, amely végighaladva a láncon *Clear Indication* csomagként jut el a DTE2-höz. Ez értesíti a használóját *N_DISCONNECT.indication* primitívvel, majd nyugtázza egy *DTE Clear Confirm* csomag visszaküldésével. E csomag a DTE1-be érkeve, ott is az összeköttetés lebontását eredményezi.

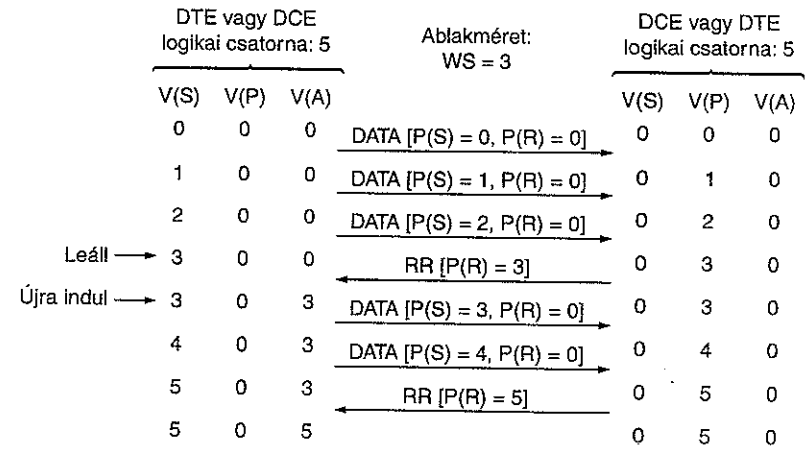
Kis mennyiségű felhasználói adat esetén az összeköttetés létesítésével járó többlet-forgalom csökkentése érdekében használják a **gyors kiválasztást**. Ekkor a *Call Request* csomag egy 16–128 bájttal hosszúságú adatmezőt is tartalmazhat, s nem szükséges külön *DTE Data* csomagot küldeni. Ilyenkor az összeköttetés létesítése után közvetlenül annak bontása következik.

Forgalomszabályozás

A **forgalomszabályozás** a csomagszinten sokkal hangsúlyosabb, mint a hibavédelem, mert a csomagtovábbítás a megbízható adatkapcsolati szolgáltatón alapul. A forgalomszabályozás a csúszó- vagy forgóablakos (sliding window) mechanizmuson alapul, s nagyon hasonlít az adatkapcsolati szintű forgóablakos megoldáshoz. A csomagáramot logikai csatornánként szabályozzák. Az áramlás mindkét irányában külön-külön, azaz a DTE DCE irányú csomagáram és a DCE DTE irányú csomagáram külön szabályozott.

A csúszó- vagy forgóablakos mechanizmus implementálásához két sorszám szükséges: az *elküldött csomag sorszáma* $P(S)$ és a *vett csomag sorszáma* $P(R)$. A $P(R)$ minden adatcsomagban megtalálható és az ellenirányba elküldött csomagáramra vo-

¹ A lektor megjegyzése: Az *Incoming Call* csomag tulajdonképpen a *Call Request* csomag, csak a hívott oldalon jelenik meg, és a hívott oldal lokális logikai csatornaszámát tartalmazza.



F.8. ábra. Csúszóablak mechanizmus

natkozik. Ha nincs továbbítandó adat, akkor a nyugtázó *RR csomag* kerül elküldésre, s a $P(R)$ ebben is megtalálható, szerepe az előzőével azonos. Az *adási ablak* méretét az összeköttetés létesítésekor kell meghatározni (pl. default értéke = 2). Csak olyan csomagok továbbíthatók, amelyek adási sorszáma az adási ablakon belül van. A maximális ablakméret 7, ha modulo 8 sorszámozást használnak. Az implementáció érdekében mind a DTE mind a DCE három változót kezel:

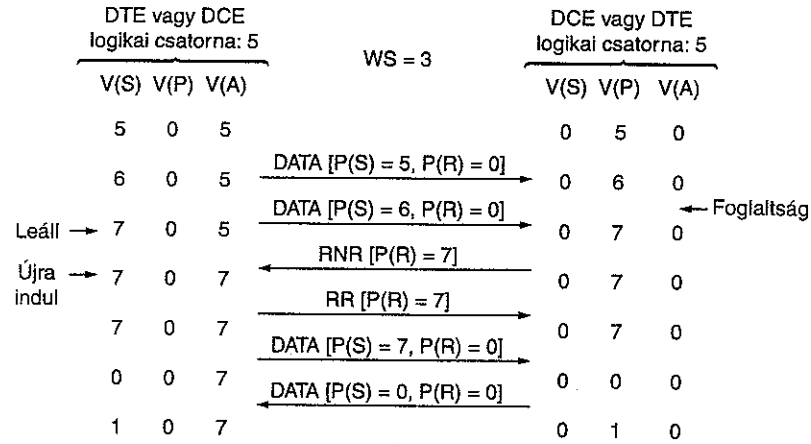
$V(S)$: az elküldött sorszám, az adott logikai csatornán elküldött csomag sorszáma $P(S)$

$V(R)$: a vett sorszám, az adott logikai csatornán legközelebb legközelebb ellenirányban elküldendő $P(S)$ sorszám, vagyis a legközelebb várt csomag sorszáma

$V(A)$: nyugta-változó az adási ablak alsó küszöbértékének jelölésére; aktuális értéke a beérkező csomag $P(R)$ értékével egyenlő.

Kezdetben mindhárom változó értéke 0. Minden adatcsomag adásra történő előkészítésénél $P(S) = V(S)$, és a GFI mezőben lévő bit értékétől függően modulo 8 vagy modulo 128 szerint növekedik. Minden adatcsomag vételénél a csomagban lévő $P(R)$ értéke betöltődik a $V(A)$ változóba. az adás a maximális ablakméret eléréséig folyamatos lehet, vagy az adat vagy *RR csomagban* lévő olyan $P(R)$ -ig, amely előreviszi az aktuális $V(A)$ -t (F.8. ábra). Ha a vevőben rendelkezésre álló pufferek száma kisebb, mint a szükséges szám, vagyis a vevő foglalttá válik, akkor a vevő *RNR csomag* küldésével válaszol². Minden ilyen csomagban lévő $P(R)$ az adott logikai csatornán lévő

² A lektor megjegyzése: Lényegében az *RNR csomag* nem használatos. Ha ugyanis nincs elég puffer, akkor a vevő nem küld vissza *RR-t*, ezáltal az ablak alsó szélé nem fordul tovább. Ennek megfelelően az F.9. ábrának is csak elvi jelentősége van.



F.9. ábra. RNR működés

V(A) új értékét definiálja. Az RNR vételek az adó beszünteti a csomagok küldését, amíg a partnertől RR csomagot nem kap (lásd F.9. ábra).

Sürgős felhasználói adatok továbbítására is van mód. Ilyenkor a kezdeményező DTE egy Interrupt Request csomagot generál és küld a DCE-nek. Ilyenkor az ilyen csomag a folyamatos adatcsomag áramot megszakítja, abba beékelődve kerül továbbításra.

Hibavédelem

A hibavédelem a Reset és a Restart csomagok elküldését jelenti. A reset eljárás csak adatátviteli fázisban használatos és csak egyetlen virtuális áramkörre van hatással. A restart eljárás ezzel szemben az összes létező virtuális áramkörre hatással van.

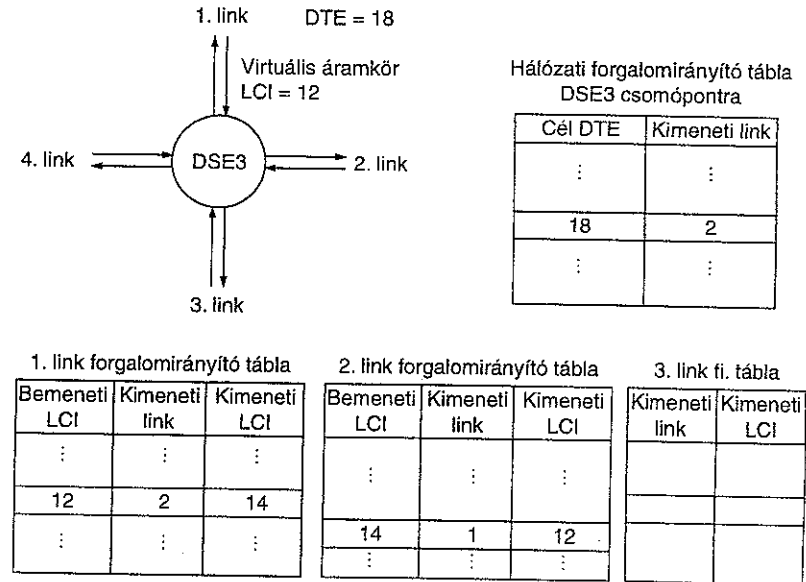
A reset eljárás a virtuális áramkör alapállapotba állítására szolgál, például hibaelhárítás esetén. Ilyenkor az összes tárolt csomag törlődik. A Reset utáni újraindítás a szállítási réteg feladata. A reset kiadásának oka lehet hamis P(S) vagy P(R) vétele, adási ablakon kívüli csomag vétele.

A restart eljárást akkor használják, amikor a DTE összes virtuális összeköttetését le akarják bontani. Erre rendszerint akkor van szükség, ha a DTE és a DCE közötti szinkron olyan mértékben elromlik, hogy az már veszélyezteti az összes aktív összeköttetést. Ilyen eset fordulhat elő például, ha egy DCE-től továbbított Incoming Call csomag olyan logikai csatornaszámot tartalmaz, amelyet a DTE már használ más virtuális áramkörnél.

Forgalomirányítás

Jóllehet a forgalomirányítás nem része az X.25 ajánlásnak, amikor a virtuális áramkör létrejön, azt az egész hálózaton keresztül úgy kell létrehozni, hogy az adatcsomagok ezen az áramkörön haladjanak csomópontról csomópontra a hálózaton keresztül. Különböző forgalomirányító algoritmusok használatosak. Eltekintve a konkrét algoritmusoktól, közös jellemzőjük az, hogy a virtuális áramkör létesítése a csomópontok (DSE-k) által kezelt forgalomirányító táblák segítségével történik. Egy csomópont kétféle táblát kezel. A hálózati forgalomirányító tábla azt mutatja meg, hogy az adott csomóponton a hálózathoz kapcsolódó DTE-k felé melyik kimeneti adatkapcsolatot (linket) kell választani. Az ún. link forgalomirányító tábla pedig azt mutatja meg, hogy egy adott kimeneti adatkapcsolathoz milyen logikai csatorna azonosító (LCI) tartozik egy bizonyos bemeneti logikai csatornaszám esetén.

Az összeköttetést kezdeményező Call Request (CR) csomag fejrészében lévő cél-DTE címe alapján a csomópont a hálózati forgalomirányító táblából meghatározza a kimeneti adatkapcsolat számát, amerre a csomagot továbbítani kell. Ezután megkeresi az azon az adatkapcsolaton kezelt első szabad logikai csatorna számát a szabad LCI-k listájáról és bejegyzi a bemeneti és kimeneti oldal link forgalomirányító táblájába. A fentiek bemutatására az F.10. ábra szolgál. A CR csomag az 1. link felől érkezik 12 logikai csatorna számmal, a cél DTE címe 18. A csomópont a 2. link felé irányítja és logikai csatornaszámot rendel hozzá, amelynek értéke 14 lesz. Egyidőben a 2. link táblájába is bevezeti a 14-hez az 1. linket és a 12. logikai csatorna számot. Ezután el-



F.10. ábra. Forgalomirányítás

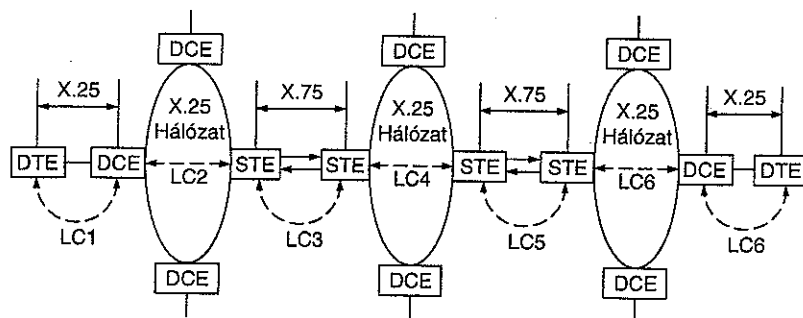
kezdí továbbítani a *CR csomagot* a 2. linken a csomag fejrészében 14 logikai csatorna számmal. ez a folyamat zajlik le a soronkövetkező összes csomópontban, amíg a cél DTE-hez kapcsolódó csomópontot a *CR csomag* el nem éri. Mivel a virtuális áramkör kialakult, a válasz ugyanezen az útvonalon halad vissza, majd az adatcsomagok is ezt az útvonalat követik.

F.3. Terminál hozzáférés

Sok esetben előfordul, hogy a DTE nem rendelkezik akkora intelligenciával (feldolgozó képességgel), amely lehetővé tenné a csomagszintű működést. Ez fordul elő egyszerűbb terminálok csatlakoztatása esetén is. Ilyenkor a korrek működés érdekében *kiegészítő ún. PAD (Packet Assembly Disassembly) eszközöket* kell a DTE és a DCE közé helyezni. Egy karakter módú terminál és a PAD között egy RS-232C/V.24 interfész lehet, míg a PAD és a csomópont az X.25 interfészen kommunikál. A PAD arra szolgál, hogy a terminál felől jövő karaktereket összerakja csomaggá, és fordítva. Az ilyen aszinkron terminálról történő hálózati hozzáférést az X.28 ajánlás szabályozza. Mivel a PAD berendezést rendszerint a szolgáltató felügyeli, normális esetben a DSE-vel azonos helyen található. A PAD és a távoli csomag-DTE közötti kapcsolat szabályozása az X.29 ajánlás alapján történik, míg magának a PAD-nek a működését az X.3 ajánlás írja le.

F.4. X.25 hálózatok összekapcsolása

Két különálló hálózathoz kapcsolódó DTE közötti adatforgalom biztosítása érdekében az X.25-ös hálózatokat össze kell kapcsolni. Mivel egy hálózatban a DTE-k között virtuális áramkört kell kialakítani, a két különböző hálózathoz kapcsolódó DTE-k között is virtuális áramkör kialakítása szükséges, ahogy az az F.11. ábrán látható. Az



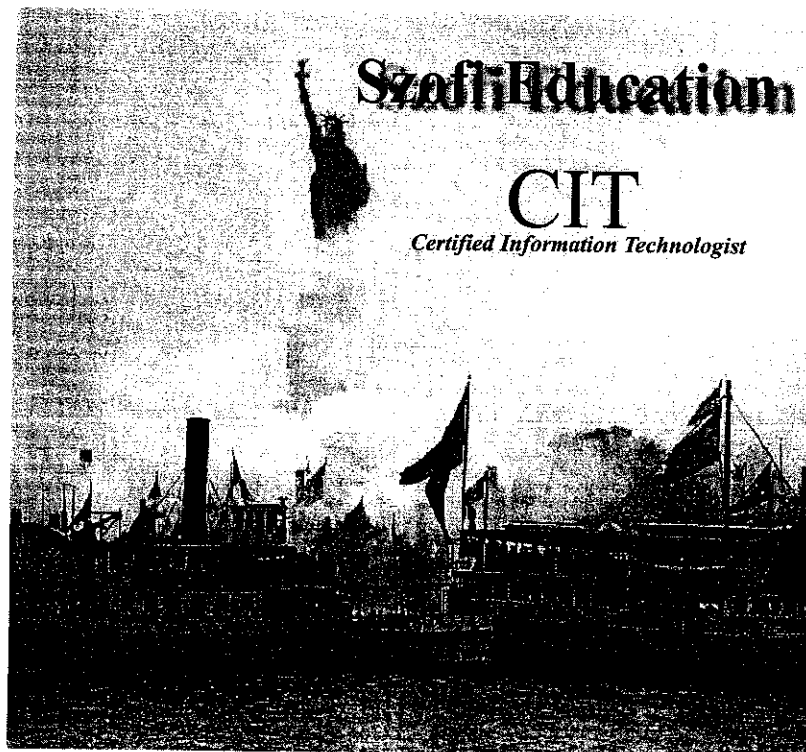
F.11. ábra. X.25 hálózatok összekapcsolása

X.25 alapú hálózatok virtuális áramkörökkel történő összekapcsolásához speciális DCE csatoló eszközre van szükség, amelyet **jelzési terminálnak (Signalling Terminal Equipment, STE)** neveznek. Az STE működését és az általa használt protokollokat a CCITT X.75 ajánlása specifikálja.

Az X.75, az X.25-höz hasonlóan, három protokoll réteget definiál: a csomag réteg protokollt (PLP), az adatkapcsolati réteg protokollt (DLP) és a fizikai réteg protokollt (PHY). Az *adatkapcsolati rétegben* a nagyobb hatékonyság érdekében a *több adatkapcsolatot kezelő eljárás (Multilink Procedure, MLP)*³ használatos. Ez a LAPB protokoll egyfajta kiterjesztése. Az adatkapcsolati rétegbe, a LAPB fölé egy *többkapcsolati-vezérlő alréteget (Multilink Control Sublayer, MLC)* helyeztek lehetővé téve több, LAPB protokoll alapján működő kapcsolat egyidejű létezését is. Mivel a keret fejrészében lévő újonnan bevezetett MLC vezérlő mező két bájt hosszúságú, lehetővé válik 12 bites sorszámok használata. Ez azért jelentős, mert így 4096 egymástól különböző adatkapcsolati összeköttetés (link) kezelhető, vagyis ugyanannyi, amennyi virtuális áramkör haladhat át a két eszköz között.

Az X.75 csomag rétege egyszerűbb, mint az X.25 ugyanezen rétege. Ez azért lehetséges, mert itt két közvetlenül összekötött STE között kell a kapcsolatot felépíteni és fenntartani. A csomagformátum nagyon hasonló az X.25 csomagformátumhoz az a különbséggel, hogy a *hálózati szolgáltatások* nevű mező hosszát és magát a *szolgáltatásokat* tartalmazó mező bekerült a *Call Request (CR)* csomagformátumba. Ez teszi lehetővé az STE számára, hogy jelezze a másik STE-nek az ablak és csomag méretét, a becsült áthaladási késleltetés mértékét, valamint a gyors kiválasztást. Az STE miközben átjárást biztosít a két hálózat között, csomagformátum konverziót is végez. Minden STE-nek speciális X.121 címe van. A cím hierarchikus szerkezete miatt a vevő STE a vett csomagban lévő cél-cím legfelső helyiértékén lévő biteket veszi figyelembe a következő STE vagy a címzett DCE/DTE meghatározásához. Egy *CR csomag* vételekor elvégzi a virtuális áramkör (logikai csatorna) kezelésével kapcsolatos adminisztrációt, majd a csomag fejrészébe beírja az új logikai csatorna számot és a megfelelő hálózati szolgáltatási paramétereket, azután elkezdi átvinni az csomagot az STE-STE adatkapcsolaton a megfelelő protokollok segítségével. A vevő STE a csomagot ezután a következő hálózat megfelelő csomópontja irányába továbbítja a szükséges adminisztráció, forgalomirányítás és csomagátalakítás elvégzése után. Az adatcsomagok a kialakult virtuális áramkörtön ezután már könnyűszerrel eljutnak a címzethez.

³ A lektor megjegyzése: Régebben használatos protokoll a sávszélesség növelésére. Újabban a multimodemes PPP vagy a SLIP használatos.



Informatikusok folyamatos továbbképzése
 Gyakorlati kiegészítő képzések informatikus hallgatók számára
 Számítógép-hálózati rendszeradminisztrátor
 Számítógép-hálózati rendszergazda
 Rendszerprogramozó, Programozó
 Rendszerelemző, rendszerszervező képzések

ORACLE, Windows NT, Novell Netware, Hálózatmenedzsment
 C, C++, SSADM, SQL, Internetworking gyakorlati képzések

**Szofi Magyar-Amerikai Informatikai
 Oktató- és Továbbképző Központ**

WWW.SZOFI.HU
 New York, Szeged, Budapest

Tárgymutató

- 1-perzisztens CSMA, 279
- 100Base-FX, 357
- 100Base-T, 357
- 100Base-T4, 356
- 100Base-TX, 356
- 10Base-F, 309
- 10Base-T, 307
- 10Base2, 307
- 10Base5, 307
- 4B5B kódolás, 356
- 5-ből 4 kódolás, 353
- 8 bit leképezve 6 tercre, 356
- 802.3u, 355
- 8B6T, 356
- A-interfész, 299
- A-oldali szolgáltató, 187
- AAL, 86, 582
- absztrakt ablak eszközkészlet, 758
- absztrakt szintaxis jelölés, 673
- ACR, 507
- ACTS, 364
- adaptív algoritmusok, 382
- adási ablak, 232
- adat-végberendezés, 138
- adatáramkört lezáró berendezés, 138
- adatelelem, 821
- adatfolyam gép, 26
- adatkapcsolat-vezérlő protokoll, 259
- adatkapcsolati elérési eljárás, 254
- adatkapcsolati réteg, 204–270
 - bitbeszúrás, 210
 - csúszóablakos protokoll, 230–247
 - elemi protokollok, 219–230
 - forgalomszabályozás, 211–212
 - HDLC protokoll, 253–256
 - hibavédelem, 210–211, 212–219
 - karakterbeszúrás, 209
 - keretezés, 207–210
 - nyújtott szolgálatok, 205–207
 - példa protokollok, 253–266
 - tervezési szempontok, 204–212
- adatkapcsolati réteg, 49
- adatkapcsolati titkosítás, 618
- adatkapcsoló berendezés, 30
- adatkeret, 49
- adatszórás, 26, 311, 405
- adatszóró hálózatok, 25
- adattitkosítási szabvány, 627–637
 - láncolás, 629–632
 - támadás, feltörési kísérletezés, 632–637
 - vitatás, 632
- adattömörítés, 770–783
 - diszkrét koszinusz-transzformáció, 773
 - entrópiakódolás, 771–772
 - forráskódolás, 772–774
 - futamhossz kódolás, 771
 - különbégi kódolás, 772
 - statisztikai kódolás, 772
 - transzformáció kódolás, 773
 - vektorkvantálás, 773
 - veszteséges, 771
 - veszteségmentes, 771
- adattovábbító folyamat, 820

ADC, 765
 ADCCP, 253
 adó-vevő kábel, 308
 adó-vevő, 308
 ADSL, 791
 aktív ismétlő, 115
 alagút típusú átvitel, 439
 alap kódolási szabályok, 676
 alapsávi koaxiális kábel, 107
 alapvető bit-térkép eljárás, 283
 alcím, 469
 alhálózat, 29
 csomagkapcsolt, 31
 két pont közötti/kétpontos, 31
 tárol-és-továbbít, 31
 alhálózati maszk, 455
 alhálózatok, 454
 alkalmazási átjáró, 447
 alkalmazási átjárók, 434
 alkalmazási folyamat, 808
 alkalmazási réteg, 53, 57
 alkalmazási réteg, 53, 57, 616–807
 elektronikus levél, 684–710
 hálózatfelügyelet, 670–684
 hálózati biztonság, 616–661
 hálózati hírek, 710–721
 körzeti névkezelő rendszer, 661–669
 multimédia, 764–800
 Világháló, 721–764
 alkalmazási szolgáltató, 808
 állandó bitsebesség, 495
 állandó lakhely, 403
 állandó virtuális áramkör, 81, 171
 állapot-átmenet, 248
 állomás modell, 273
 állomás, 273
 ALOHA, 275–279
 egyszerű, 275–278
 műholdas, 362
 réselt, 278–279
 alosztály, 754
 alsómeteszű rendszer, 109
 általános erőforrás-azonosító, 735
 Amerikai Országos Szabványügyi
 Intézet, 92
 amplitúdómoduláció, 134
 AMPS, 185
 analóg celluláris telefon, 184
 analóg-digitális átalakító, 765
 ANS, 72
 ANSI, 92
 ANSNET, 73
 áramkör kapcsolás, 155
 árnyékolatlan csavart érpár, 107
 ARP, 457
 ARQ, 229
 ASCII vértzet, 694
 ASN.1, 673
 aszimmetrikus átvitel, 139
 aszimmetrikus digitális előfizetői vonal,
 791
 aszinkron átviteli mód, 83–88
 adatkapcsolati réteg, 262–266
 cella formátum, 487–489
 CS alréteg, 87
 felhasználói sík, 86
 forgalom formálás, 500–504
 forgalomirányítás és kapcsolás, 491–
 494
 jövője, 87
 lyukas vödör, 501
 összeköttetés létesítés, 489–491
 PMD alréteg, 87
 SAR alréteg, 87
 szolgálati osztályok, 495–497
 szolgálat minősége, 497–500
 TC alréteg, 87
 torlódásvédelem, 504–507
 vezérlési sík, 86
 virtuális csatorna, 486
 virtuális útvonal, 487
 aszinkron transzfer mód, 83
 áthallás, 133
 átjáró, 34
 átkapcsolásos rendszer, 184
 ATM adaptációs réteg, 583
 ATM adaptációs réteg, 86
 ATM Forum, 87
 ATM kapcsoló, 174
 ATM réteg, 85

ATM, 83
 ATMARP, 509
 átmenet, 252
 átmenő hálózatok, 465
 attribútum, 820
 átvitel
 duplex, 40, 137
 fél-duplex, 39, 137
 szimplex, 39
 átvitel, 105–125
 infravörös, 123
 látható fényhullámú, 124
 vezetékes, 105–117
 vezeték nélküli, 117–125
 átviteli konvergencia alréteg, 87
 felhasználói sík, 86
 fizikai közegtől függő alréteg, 173
 fizikai réteg, 85
 fizikai rétegtől függő alréteg, 87
 kapcsológép, 174
 konvergencia alréteg, 87
 vezérlési sík, 86
 átviteli szintaxis, 676
 átvitelvezérlő protokoll, 56
 átvitelvezérlő protokoll, 56–57, 560–582,
 699, 718, 725
 átviteli politika, 570–574
 buta ablak jelenség, 573–574
 időzítő kezelése, 577–579
 Karn-féle algoritmus, 579
 Nagle-féle algoritmus, 572
 összeköttetés kezelése, 566–570
 szolgálati modell, 560–562
 torlódásvédelem, 574–577
 vezeték nélküli hálózat, 580–582
 Aurora, 77
 automatikus ismétléskérés, 229
 autonóm rendszerek, 441, 447
 az Internet Társaság, 94
 B-ISDN, 83, 170
 B-oldali szolgáltató, 187
 bárhova küldés, 478
 base64 kódolás, 694
 Batcher–banyan kapcsoló, 178
 baud, 103
 beágyazás, 753
 becsatlakozó adatfolyam, 153
 belépés ellenőrzése, 421
 Bell Üzemeltető Vállalat, 130
 Bellman–Ford forgalomirányítási
 algoritmus, 390
 belső átjáró protokoll, 441, 460
 bemeneti él, 252
 BER, 676
 betűhármások, 622
 betűkettősök, 622
 BGP, 465
 bináris visszaszámlálás, 284
 birodalom, 651
 bitbeszúrás, 210
 bizottsági javaslat, 92
 biztonságos hash algoritmus, 658
 Blanca, 77
 BOC, 130
 BOOTP, 459
 bor politika, 427
 boríték, 687
 böngésző, 722
 BUS, 509
 buta ablak jelenség, 573
 Caesar-titkosító, 621
 CASA, 77
 CCITT, 90
 CDMA, 301
 CDPD, 33, 298
 CDV, 499
 CDVT, 498
 cella, 31, 83, 742
 cellaátviteli késleltetés, 499
 cellahibaarány, 499
 cellakapcsolás, 84
 cellakésleltetés szórása, 499
 cellakésleltetés szórásának türése, 498
 cellavesztési arány, 499
 celluláris datagram szolgálat, 33, 298
 celluláris rádió, 182
 CER, 499

CGI, 746
 CIDR, 471
 ciklikus redundancia, 215
 címfeloldási protokoll, 457
 címfeloldó, 662
 címkék, 737
 CLR, 499
 CLUT, 772
 CMR, 499
 CRC kód, 215
 crossbar kapcsoló, 161
 CS, 87
 csak titkosított szöveg, 621
 csapda irányított lekérdezés, 672
 csapda, 672
 csatorna átadás, 187
 csatornafigyelő protokollok, 279
 csatornafüggő jelzés, 147
 csavart érpár, 106
 3-as kategóriájú, 107
 5-ös kategóriájú, 107
 csíkozás, 789
 csillag alakú gyűrű, 327
 csillagkép mintázat, 135
 csillapítás, 133
 CSMA/CD, 281
 CSNET, 71
 csomag, 25, 753
 csomag-összerakó/szétzabdalo, 82
 csomag-végberendezés, 855
 csomagformátum, 858–861
 csomagkapcsolás, 158
 csomagkapcsoló csomópont, 30
 csomagolt elemi folyam, 784
 csomagszűrő, 446
 csomóponti gép, 68
 csomóponti számítógép, 69
 csonka hálózatok, 465
 csoport, 144
 csővezetékezés, 237
 csúcs cellasebesség, 498
 csúszóablakos protokoll, 231
 csúszómászók, 760
 CTD, 499
 darabok, 443
 datagram szolgálat, 43
 datagramok, 377
 dB, 765
 DCE, 138
 DCS 1800, 296
 DCT, 774
 de facto szabvány, 89
 de jure szabvány, 89
 decibel, 105
 dekódoló, 771
 deltamoduláció, 148
 DES, 627
 differenciális kriptanalízis, 635
 differenciális Manchester-kódolás, 310
 Diffie–Hellman-féle kulcscsere, 645
 digitális adatszóró hálózat, 797
 digitális aláírás szabvány, 656
 digitális aláírás, 653–660
 nyilvános kulcsú, 654–656
 titkos kulcsú, 653–654
 üzenet pecsét, 656–658
 digitális biteső, 166
 digitális celluláris telefon, 189
 díjszabási könyv, 90
 direktívák, 737
 Diszkrét idő, 274
 diszkrét koszinusz-transzformáció, 774
 DMSP, 702
 DNS, 662
 domain, 663
 DQDB, 29, 333
 DSMA, 300
 DTE, 138
 duplex átvitel, 40
 duplex, 137
 DVMRP, 799
 dzsitter, 421
 E-interfész, 299
 e-levél átjáró, 701
 e-levél, 23, 74, 684–710
 architektúra és szolgálatok, 685–687
 átjáró, 701
 boríték, 687

fejrész, 687
 feladatok, funkciók, 685–687
 felhasználói ügynök, 685, 687–691
 felhasználói parancs, 689–691
 korai rendszerek, 684
 küldés, 688
 MIME formátum, 693–698
 olvasás, 688–691
 RFC 822 formátum, 691–693
 személyiségi jogok, 703–710
 szűrő, 702
 utolsó kézbesítés, 702–703
 üzenetformátum, 691–698
 üzenet kézbesítés, 698–703
 üzenetkézbesítő ügynök, 685
 törzs, 687
 E1, 147
 egybetű-helyettesítéses titkosítás, 622
 egyéni kulcsgyűrű, 706
 egyenlő esélyű sorba állítás, 423
 egyetlen csatorna feltételezése, 273
 egyeztetés, 45
 egyhelyben-tartózkodási időnek, 361
 egymódusú szál, 111
 egységes erőforrás-meghatározó, 732
 egyszer használatos bitminta, 624
 egyszer-elküldött, egyedi üzenetszám, 647
 egyszerű ALOHA, 278
 egyszerű cím, 530
 egyszerű hálózatfelügyelő protokoll, 670
 egyszerű hatékony adaptációs réteg, 590
 egyszerű levéltovábbító protokoll, 699
 elárasztás, 387
 elég jól biztosított személyiségi jog, 704
 elektronikus kódkönyv mód, 629
 elemi művelet, 44
 elérhetőség vizsgálat, 248
 életben tartó időzítő, 579
 e-levél, 74
 elhalkulás, 122
 eljárás, 731
 elosztó, 307
 elosztott levelezési rendszer protokoll, 702
 elosztott rendszer, 20
 élő-lánc támadás, 646
 előfizetői hurok, 128, 132
 élőkapocs, 722
 előzetes szabvány, 94
 elrendezés, 685
 emoticon, 714
 engedélyezett átmenet, 252
 entitás, 41
 entrópiakódolás, 772
 ER, 507
 eredeti cellasebesség algoritmus, 500
 erőforrás-bejegyzés, 664
 erőforrás-foglalási protokoll, 429
 erőforrás-kezelő, 507
 erőforrás-megosztás, 21
 érzelmet kifejező jelzések, 714
 ESMTP, 700
 Ethernet, 28, 306
 explicit sebesség, 507
 extended SMTP, 700
 fájl transzfer, 75
 fájlátviteli protokoll, 818
 fájllelési adategység, 821
 fájlművelet, 820
 fájlstruktúra, 820
 FAQ, 703, 714
 fázismoduláció, 135
 FDDI, 351
 FDM, 142
 fejlett adatkommunikáció-vezérlő
 eljárás, 254
 fejlett kommunikáció-technológiai
 műhold, 364
 fejrész hibavédelem, 263
 fejrész, 38
 e-levél, 687, 691
 keret, 221
 fejrész, 38, 687
 fejrész-előrejelzés, 605
 fél-duplex átvitel, 39
 fél-duplex, 137
 félátjáró, 434
 felfelé nyalábolás, 544
 felhasználó-hálózat interfész, 487
 felhasználói datagram protokoll, 57

felhasználói profil, 689
 felhasználói ügynök, 811
 felhasználói ügynök, 823
 felhívó csatorna, 298
 felkutató keret, 347
 felügyeleti adatbázis, 672
 felügyeleti adatok struktúrája, 679
 felügyeleti állomás, 671
 felügyelő állomás, 329
 felügyelő keret, 254
 felügyelt hírcsoport, 714
 felületi hozzáférés, 257
 fénysebesség, 118
 fényvezető szál, 110
 fényvezető szál, 110–117
 egymódusú, 111
 többmódusú, 111
 fényvezető szál, üvegszál, 110–117
 alapelvek, 110–112
 csillapítás, csillapodás, 112
 egymódusú, 111
 szóródás, 113
 többmódusú, 111
 fényvezető szálcsatorna, 359
 fényvezető szálcsatorna osztott adat interfész, 351
 férgék, 760
 feszítőfa, 406
 feszítőfás híd, 342
 fizikai közeg, 36
 fizikai réteg, 49
 fizikai réteg, 49, 100–203
 átviteli közeg, 105–117
 celluláris rádió, 182
 távközlési műhold, 191
 telefon rendszer, 125–165
 vezeték nélküli átvitel, 117–125
 flamewar, 713
 foglalásos protokoll, 283
 folyamalapú forgalomirányítás, 388
 folyamatszolgáltatás, 529
 folyammeghatározás, 420
 folytatódó időzítő, 579
 folytonos idő, 274
 folytonos média, 764
 Ford–Fulkerson-algoritmus, 390
 fordított címfeloldási protokoll, 459
 forgalmi politika, 415
 forgalomformálás, 415
 forgalomirányítás, 865–866
 forgalomirányító algoritmus, 381
 forgalomirányító algoritmus, 381–409
 adaptív, 382
 adatszórásos, 405–407
 elárasztásos, 387
 folyam alapú, 388–390
 hálózatok közötti, 441–442
 hierarchikus, 400–402
 kapcsolat-állapot alapú, 394–400
 legrövidebb út alapú, 384–387
 mobil hoszt, 402–405
 nemadaptív, 382–387
 távolságvektor alapú, 390–394
 többbesküldés, 407–409
 visszirányú továbbítással, 407
 forgalomlefró, 497
 forgalomszabályozás, 52, 212
 forgalomszabályozás, 862–864
 forrás általi forgalomirányítás, 346
 forráskódolás, 773
 Fourier-sor, 101
 főállomás, 109
 főcsoport, 144
 főszereplők, 641
 frame relay, 82
 frekvencia billentyűzés, 134
 frekvencia, 118
 frekvenciamoduláció, 134
 frekvenciaosztásos multiplexelés, 142
 FTTC, 141, 792
 FTTH, 141, 792
 futamhosszkódolás, 772
 fuzball, 71
 generátor polinom, 216
 geoszinkron műhold, 192
 gerinchálózat, 462
 globális mobilkommunikációs rendszer, 190
 GSM, 190, 295
 gyakran feltett kérdések, 703, 714

gyors Ethernet, 355
 gyors kiválasztás, 861
 hagyományos telefonszolgáltatás, 169
 hálózat elérési pont, 73
 hálózat-hálózat interfész, 487
 hálózati architektúra, 36
 hálózati hírek, 710
 hálózati mag protokoll, 66
 hálózati rendszer architektúra, 61
 hálózati réteg, 50
 hálózati réteg, 50, 374–516
 belső szervezés, 377–378
 forgalomirányító algoritmusok, 381–409
 hálózatok összekapcsolása, 431–447
 nyújtott szolgálatok, 374–377
 tervezési kérdések, 374–381
 torlódásvédelem, 409–431
 hálózati vezérlő protokoll, 259
 hálózati videózás, 784–796
 elosztó hálózat, 790–793
 készülék doboz, 794–795
 szolgáltató, szerver, 785–790
 hálózati virtuális terminál, 813
 hálózatközi protokoll, 448
 hálózatok összekapcsolása, 431–447
 összeköttetés alapú, 436–437
 összeköttetés nélküli, 437–439
 Hamming távolság, 213
 hangminőségű vonal, 103
 hangszerek digitális interfésze, 766
 harmonikus, 101
 három medve problémája, 470
 háromutas kézfogás, 534
 határátjáró protokoll, 465
 hátrafelé tanulás, 343
 hazai ügynök, 403
 HDLC, 254
 HDTV, 770
 HEC, 263
 hely, 251
 helyettesítő ARP, 458
 helyettesítő kiszolgáló, 729
 helyettesítő kódoló, 621
 helyettesítő ügynök, 672
 helyi körzet, 131
 helyi központ, 127
 helyi szolgáltató, 131
 helyközi trónk, 128
 HFC, 792
 hibajavító kód, 213
 hibajelző kód, 213
 hibás cellákat tartalmazó blokkok
 aránya, 499
 hibavédelem, 864
 hibrid üvegszál-coax, 792
 híd, 336–351, 433
 feszítőfás, 345–346
 forrásirányított, 346–348
 távolsági, 350–351
 transzparens, 342–346
 hidak, 336, 433
 hierarchikus címek, 529
 hipermedia, 724
 hipertext átviteli protokoll, 730
 hipertext jelölő nyelv, 732
 hipertext jelölő nyelv, 732–746
 űrlapja, 742–746
 verzija, 732–742
 hipertext, 722
 HIPPI, 358
 hírcsoport átviteli protokoll, 717
 hirdetés, 469
 hírek, 74
 hírforrás, 715
 hitel-üzenet, 556
 hiteles bejegyzés, 668
 hitelességvizsgálat, 641
 hitelességvizsgáló protokoll, 641–652
 nyilvános kulcsú, 651–652
 hivatkozási modell
 ISO OSI, 48
 TCP/IP, 55
 holtpon, 250
 hosszú idejű cellasebesség, 498
 hoszt, 29
 hozzáférés engedélyező csatorna, 298
 HTML, 732
 HTTP, 730

hub, 193
 hullámhossz, 118
 hullámhosszosztásos multiplexelés, 144
 hullámhosszosztásos többszörös hozzáférés, 289
 huzalközpont, 327
 hyperlink, 722

 I-interfész, 299
 IAB, 93
 ICMP, 455
 IDEA, 635
 idegen ügynök, 403
 idézett nyomtatható karakteres kódolás, 694
 időbeli reflektometria, 307
 időkerék, 606
 időosztásos kapcsoló, 164
 időosztásos multiplexelés, 142
 időrészcsere, 164
 IDU, 41
 IEEE 802.3, 306–318
 gyors Ethernet, 354–357
 kapcsolt, 316–318
 kábelezés, 307–310
 keretformátum, 311
 kódolás, 310–311
 teljesítménye, 314–316
 protokoll, 311–314
 IEEE 802.4, 316–318
 gyűrű karbantartás, 322–323
 protokoll, 319–323
 IEEE 802.5, 324–331
 gyűrű karbantartás, 329–331
 protokoll, 327–331
 IEEE, 93
 IETF, 94
 igazoló hatóságok, 708
 IGMP, 467
 IMP, 68
 impulzuskód-moduláció, 146
 IMTS, 184
 információs keret, 254
 inicializáló vektort, 630

 integrált szolgáltatást nyújtó digitális hálózat, 165
 interfész adategység, 41
 interfész, 36
 internet alkalmazások, 484–800
 e-levél, 684–710
 hálózati hírek, 710–721
 multimédia, 763–800
 Világháló, Web 721–763
 Internet Architektúra Testület, 93
 internet csoportfelügyeleti protokoll, 467
 Internet Koordinációs Testület, 93
 Internet protokoll, 56
 internet réteg, 55
 Internet Szabvány, 94
 Internet Társaság, 74
 internet vezérlőüzenet protokoll, 455
 internet, 34, 431
 internet, 34, 55, 73–75, 256, 447
 adatkapcsolati réteg, 256–262
 forgalomirányító protokoll, 460–473
 internet rétege, 55, 447–485
 mobil IP, 468–470
 összeköttetések kezelése, 566–570
 többesküldés, 467–468
 történet, 73–75
 Internet-szolgáltató, 257
 internetes irányvonal nyilvántartó hatóság, 708
 internetwork, 34
 IP, 56, 448
 ipari/tudományos/orsvosi sáv, 123
 IPRA, 708
 IPv6, 473–485
 cím, 477
 ellentmondások, 483–485
 fő fejrész, 475–480
 jumbogram, 481
 kiegészítő fejrész, 480–483
 IPv6, 474
 IPX, 66
 irányelvet igazoló hatóság, 708
 IRTF, 94
 IS-IS, 400
 ISDN, 165

alapsebesség, 168
 alközpont, 168
 hibrid sebesség, 168
 keskenysávú, 169
 primer sebesség, 168
 szélessávú, 170
 ismeretlen továbbküldő, 714
 ismert nyílt szöveg alapú, 621
 ismétlők, 310, 433
 ISO, 92
 ITU, 90
 IXC, 131

 Java, 746–760
 absztrakt ablak eszközkészlet, 756–758
 biztonság, 758–760
 nyelv leírás, 749–752
 objektum alapúság, 752–756
 osztály, 753–756
 polimorfizmus, 756
 javított mobiltelefon-szolgáltatás, 184
 jel-zaj viszony, 105
 jeltorzulás, 133
 jelzési sebesség, 103
 jelzési terminál, 867
 jelző bájt, 210
 jól ismert portok, 561
 jól ismert cím, 812
 JPEG, 775

 kapcsolás, 155–161
 csomag, 158–159
 tárol-és-továbbít, 158
 telefon, 160–165
 üzenet, 158
 vonál/áramkör, 155–157
 kapcsolatállapot alapú forgalomirányítás, 395
 kapcsoló, 161–165
 időosztásos, 164–165
 keresztrudas, 161–162
 térosztásos, 162–164
 kapcsolt virtuális áramkör, 81, 171
 kapcsolt, többmegabites adatátviteli szolgáltatás, 78

 karakterbeszúrás, 209
 Karn-féle algoritmus, 579
 katalógusszolgáltató, 529
 kérdés-válasz szolgálat, 44
 keresőgépek, 760
 keresztbezáró protokoll, 652
 keresztvezetés, 161
 keresztvezetési kapcsoló, 161
 keresztütküldés, 713
 keret fejrésze, 221
 keret, 49, 207–210, 767–768
 adat, 49
 nyugtató, 49
 keret, 768
 készülék objektum, 817
 készülékek, 786
 két-hadsereg probléma, 536
 kétpontos hálózatok, 26
 kettes exponenciális visszalépés, 313
 kettős sín osztott várakozási sorral, 333
 keverő kódolók, 623
 kézbesítés, 685
 kezdeti állapot, 248
 kezdeti összeköttetést létesítő protokoll, 528
 kiegészítő fejrész, 480
 kihívás-válasz, 642
 kijelölt router, 464
 kijelző objektum, 817
 killfile, 713
 kimenet visszacsatolási mód, 631
 kimeneti él, 252
 Kínai Lottó elmélet, 633
 kisalkalmazás, 747
 kiegészítő alkalmazás, 724
 kiterjesztés, 754
 kiűtő kapcsoló, 177
 kliens, 22
 kliens-szerver modell, 22
 koaxiális kábel, 107
 koaxiális kábel, 107–110
 alapsávú, 107–108
 szélessávú, 108–110
 kodek, 146
 kézikönyv, 774

kódolás, 770
 entrópia, 771–772
 forrás, 772–774
 kódoló, 771
 kódoló, 771
 blokk, 627–637
 Caesar-féle, 621
 helyettesítő, 621–623
 keverő, 623–624
 kódosztásos többszörös hozzáférés, 301
 kódszó, 213
 kommunikációs alhálózat, 29
 kommunikációs eszköz, 22
 kompozit, 769
 konvergencia alréteg, 584
 korlátozott verseny, 286
 korszerű mobiltelefon-rendszer, 185
 körzet, 663
 körzeti névkezelő rendszer, 662
 közbeékelődéses támadás, 646
 közbülső rendszer, 30
 közelítő hálózati videózás, 785
 középben találkozó típusú feltörési kísérlet, 633
 középmetésű rendszer, 109
 közérdekű vezérlőcsatorna, 297
 közös átjáró interfész, 746
 közös csatormás jelzés, 147
 közös vezérlőcsatorna, 298
 központközi trónk, 128
 közszolgáltató, 90
 közvetett TCP, 581
 közvetítő távbeszélő szolgáltató, 131
 közvetlen sorozatú szórt spektrum, 120
 közvetlen trónk, 161
 kriptóanalízis, 620
 kriptológia, 620
 kritikus tartományok, 752
 krominancia, 769
 kulcs, 619
 kulcsszóosztó központ, 641
 különbségi impulzuskód-moduláció, 148
 különbségi kódolás, 773
 külső átjáró protokoll, 441, 460
 külső megjelenítő, 724
 kvadratúra amplitúdómoduláció, 135
 kvantálás, 777
 kvantálási zaj, 766
 LAN, 27
 lángtenger, 713
 LAP, 254
 lassú kezdést biztosító algoritmus, 576
 LATA, 131
 LCP, 259
 LEC, 131
 lefelé nyálábolás, 544
 lefojtó csomag, 423
 leghosszabb átvihető adategység, 562
 legrövidebb út, 384
 lemezfarm, 788
 lemeztömb, 789
 LES, 509
 levelezési listák, 687
 levélirtó fájl, 713
 lineáris kriptóanalízis, 635
 LLC, 306, 336
 Local Area Network, 27
 logikai csatorna, 857
 logikai IP alhálózatot, 509
 logikai kapcsolatvezérlés, 306, 336
 lokális hálózat, 27
 lokális hálózat, 27–28
 csatormakiosztás, 272–275
 gyors Ethernet, 354–357
 nagysebességű, 351–360
 vezérjeles gyűrű, 324–331
 vezérjeles sín, 318–323
 luminancia, 769
 lyukas vödör algoritmus, 415
 MAC, 271
 MACA, 293
 MACAW, 294
 magalapú fák, 409
 magas szintű adatkapcsolat-vezérlés, 254
 makroblokkok, 781
 MAN, 29
 Manchester-kódolás, 310
 MBone, 797

MCR, 498
 MCS, 364
 megáll-és-vár protokoll, 225
 megerősítetlen szolgálat, 45
 megerősített szolgálat, 45
 megjelenítés, 685
 megjelenítési réteg, 52
 megkülönböztetett vezérlőcsatorna, 298
 megnövelt személyiségi jogokat biztosító levél, 707
 megosztott láthatár, 393
 megvilágított állomás problémája, 293
 mester vezérlőállomás, 364
 mestercsoport, 144
 módszer, 753
 mező, 769
 MIB, 672
 MIDI, 766
 MIME, 693
 minimális cellasebesség, 498
 MNP 5, 136
 mobil kapcsolóközpont, 186
 mobiltelefon-központ, 186
 modem, 135
 moduláció, 134–135
 amplitúdó, 134
 fázis, 135
 frekvencia, 134
 módus, 111
 MOSPF, 800
 mozgó felhasználó, 402
 mozgó objektumok kommunikációjára alkalmas globális rendszer, 296
 MPEG szabvány, 778–784, 790–793
 B kép, 782
 D kép, 782
 folyam, 783
 I kép, 781
 makroblokk, 781
 P kép, 781
 profil, 783
 mrouter, 798
 MSC, 186
 MTSO, 186
 multimédia, 764–800
 adattömörítés, 770–783
 audió, 764–767
 hálózati videózás, 784–796
 video, 767–770
 munkatényező, 621
 műholdas hálózat, 360–366
 alacsonypályás, 195–196
 geoszinkron, 192–195
 távközlési, 191–192
 műszaki jelentések, 94
 N-ISDN, 169
 Nagle-féle algoritmus, 572
 nagy teljesítményű párhuzamos interfész, 358
 nagyfelbontású TV, 770
 nagyterjedésű hálózat, 29
 nagyobb megbízhatóság, 21
 nagyon kis aperturájú terminál, 193
 nagyvárosi hálózat, 29
 NAP, 73
 NCP, 66, 259
 Nectar, 77
 nem adaptív algoritmusok, 382
 nemperisztens CSMA, 280
 nemzetközi adat kódoló algoritmus, 636
 nemzetközi szabvány, 93
 Nemzetközi Szabványügyi Szervezet, 92
 Nemzetközi Távközlési Egyesülés, 90
 névszolgáltató, 529
 NIC, 452
 nincs vivőjel-érzékelés, 274
 NIST, 93
 NNI, 487
 NNTP, 717
 Novell NetWare, 65
 NREN, 73
 NSFNET, 72
 NT1, 166
 NT2, 168
 NTSC, 769
 null-modem, 139
 nyálábolás/multiplexelés, 142–155, 543–545
 felfelé, 544

lefelé, 544
nyelőfa, 383
nyílt rendszer, 810
nyilvános kapcsolt telefonhálózat, 126
nyilvános kulcsgyűrű, 707
nyilvános kulcsú titkosítás, 638
nyugtázó keret, 49
nyugtázott datagram szolgálat, 43

OAM, 264
objektum, 672, 753, 754
objektumorientált nyelvek, 753
okos robotok, 761
olcsó lemezek redundáns tömbje, 789
oldalak, 722
ONU, 792
opcióegyeztetés, 521
optimalitási elv, 383
Országos Szabványügyi és Technológiai Intézet, 93
ortogonális, 303
OSI hivatkozási modell, 47–55
kritikája, 60–63
TCP/IP modellhez hasonlítva, 58–60
OSI környezet, 809
OSPF (Open Shortest Path First), 460
osztály, 753
osztálynélküli körzetek közötti forgalomirányítás, 471
osztott adatszerkezet, 816

önkényes ARP, 469
öröklődés, 754
összefüggő, 464
összeköttetés-alapú szolgálat, 42
összeköttetés-mentes szolgálat, 42

P-doboz, 626
p-perzisztens CSMA, 280
P1 protokoll, 824
P2 protokoll, 824
P3 protokoll, 824
P7 protokoll, 825
PAD, 82
PAL, 769

PAR, 229
paritásbit, 213
passzív csillag, 116
PBX, 168
PCA, 708
PCM, 146
PCN, 190
PCR, 498
PCS, 190
PDU, 42
PEM, 707
PES, 783
Petri-gráf, 251
PGP, 704
PIM, 800
PMD, 87, 173
pókok, 760
polimorfizmus, 756
polinom-kód, 215
pont-pont protokoll, 259
pontnyaláb, 193
pontokkal elválasztott decimális jelölésrendszer, 452
POP, 131
POP3, 702
port, 561
postahivatal protokoll, 702
POTS, 169
pozitív nyugtázás újraküldéssel, 229
PPP, 259
prediktív kódolás, 148
primitív, 44
profil, 783
progresszív, 769
protokoll adategység, 42
protokoll automata, 247
protokoll, 35
belső átjáró, 441–442
bináris visszaszámlálás, 284–285
bit-térkép, 283–284
csúszóablakos, 230–247
elemi adakapcsolati, 219–230
fabejárás, 287–288
gigabites hálózati, 606–610
hitelességvizsgáló, 641–652

kihívás-válasz, 642–643
korlátozás nélküli szimplex, 223–224
korlátozott versenyes, 285–288
külső átjáró, 441–442
szelektív ismétléses, 241–247
többszörös hozzáférésű, 289–291
ütközésmentes, 282–285
vezeték nélküli LAN, 291–295
visszalépés n-nel, 236–241
zajos csatorna, 227–230

protokoll, 35
fájlviteli, 818
P1, 824
P2, 824
P3, 824
P7, 825
virtuális terminál, 815
protokollfüggetlen többesküldés, 800
protokollkészlet, 37
PSTN, 126
pszeudo terminál, 813
PTT, 90

Q.2931, 489
QAM, 135
QoS, 497, 519

rácsos kódolás, 136
RAID, 789
RARP, 459
ráültetés, 231
referenciaállomás, 362
rejtett állomás problémája, 293
rejtett csatornák, 760
rekurzív lekérdezés, 669
rendelkezésre álló sebesség, 496
réselt ALOHA, 278
réteg, 35
réteg, 35
adatkapcsolati, 49, 204–270
alkalmazási, 53, 57, 616–800
fizikai, 49, 100–203
hálózati, 50, 374–516
megjelenítési, 53
szállítási, 51, 517–615

viszony, 52
rezsim, 822
RM, 507
router, 30
RS-232-C, 138
RS-232, 138
RS-422-A, 139
RS-449, 139
RS-423-A, 139
RSA, 638
RSVP, 429

S-dobozok, 626
SAP, 41
SAR, 584
sávon belüli jelzés, 137
sáv szélesség-késleltetés szorlat, 595
SBR, 499
SCR, 498
SDH, 150
SDLC, 253
SEAL, 590
SECAM, 769
sejt, cella, 31, 83, 742
ATM cella, 83
celluláris rádió, 182
SHA, 658
SIPP, 474
skalázhatóság, 22
SMDS, 78
SMI, 679
smiley, 714
SMTP, 699
SNA, 61
SNMP ügynök, 671
SNMP, 670
SONET, 150
elérési út, 151
foton alréteg, 155
szakasz, 151
vonal, 151
soreleji blokkolás, 176
sorugrásos megjelenítés, 769
SPADE, 363
SPE, 152

specifikálatlan bitsebesség, 496
 SPX, 66
 SSCOP, 592
 statikus forgalomirányítás, 382
 statisztikai kódolás, 772
 stíluslap, 738
 STS-1, 152
 súlyozott egyenlő esélyű sorba állítás, 424
 sürgős adat, 562
 szabályos átvitel időzítő, 354
 szabványos általánosított jelölő nyelv, 735
 szabványosítás, 89–95
 Internet, 93–95
 hálózati, 89–95
 távközlési, 92–93
 szabványtervezet, 94
 szállítási átjárók, 434
 szállítási entitás, 517
 szállítási folyamat, 784
 szállítási funkcionális elem, 517
 szállítási protokoll adatelem, 522
 szállítási protokoll, 525
 szállítási protokoll, 525–547
 címzés, 526–530
 elemei, 525–547
 forgalomszabályozás, 539–543
 multiplexelés/nyalábolás, 543–545
 szállítási réteg, 51, 517–615
 nyújtott szolgáltatások, 517–519
 protokoll elemek, 525–547
 teljesítőképesség, 592–610
 szállítási réteg, 51, 56
 szállítási szolgáltatás elérési pont, 526
 szállítási szolgáltatás felhasználója, 519
 szállítási szolgáltató, 519
 számítógép-hálózat, 19
 számozatlan keret, 254
 szegmens, 562
 szelektív elárasztás, 387
 szelektív ismétlés, 237
 szélessávú ISDN, 83
 szélessávú koaxiális kábel, 108
 személyhívó rendszer, 182

személyi hírközlő hálózat, 190
 személyi hírközlő szolgáltatás, 190
 szerkesztés, 685
 szerver, 21
 szimmetrikus átvitel, 139
 szimmetrikus kriptográfia, 638
 szimplex átvitel, 39
 szinkódtáblázat, 772
 szinkron adatkapcsolat-vezérlés, 253
 szinkron adatokat tartalmazó boríték, 152
 szinkron digitális hierarchia, 150
 szinkron optikai hálózat, 150
 szinkron szállító jel, 152
 szinkronizálás, 52
 szint, 35
 szinuszos vivőjel, 134
 szolgálat minősége, 497
 szolgálat
 összeköttetés-alapú, 42
 összeköttetés-mentes, 42
 szolgálat, 41
 datagram, 43
 kérés-válasz, 44
 összeköttetés alapú, 42
 összeköttetés nélküli, 42
 szolgálatelérési pont, 41
 szolgálatfelhasználó, 41
 szolgálati adategység, 42
 szolgálati minőség, 42
 szolgálatminőség, 519
 szolgáltató, 41
 szoliton, 113
 szóródás, 113
 szórt spektrum, 120
 szorzat típusú titkosító, 627
 szöveg, 619
 szövegrész, 687
 szuperosztály, 754
 szűrők, 702
 takarékoság, 21
 támadó, 620
 tandem központ, 128
 tárol-és-továbbít, 158

társentitás, 36, 41
 tartományok, 400
 távhívó központ, 128
 távoli bejelentkezés, 75
 távollételet jelző démon, 703
 távolságvektor alapú forgalomirányítás, 390
 távolságvektor alapú többesküldéses forgalomirányító protokoll, 799
 TC, 87
 TCP, 56, 560
 TDM, 142
 tej politika, 427
 telefon rendszer, 125–165
 helyi hurok, előfizetői szakasz, 132–133
 kapcsolat, 160–165
 trónkók és nyalábolás/multiplexelés, 142–155
 telepont, 191
 televízió, 767
 analóg, 767–769
 digitális, 769–770
 tépőszalagos központ, 158
 terhelés eltávolítása, 426
 terminál hozzáférés, 866
 térosztásos kapcsoló, 162
 területek, 462
 téves helyre kézbesített cellák aránya, 499
 tiltott tartomány, 533
 TIP, 69
 titkosítás, kriptográfia, 616–661
 hagyományos, 619–625
 nyilvános kulcsú, 637–641
 titkos kulcsú, 626–637
 titkosított blokkok láncolása, 630
 titkosított szöveg, 620
 token (vastag pont), 252
 torlódási ablak, 575
 torlódási küszöb, 576
 torlódásvédelmi algoritmus, 409–431, 504
 általános elvek, 411–413
 lefojtó csomagok, 422–426

lyukas vödör, 415–417
 sebesség alapú, 505–507
 súlyozott egyenlő esélyű sorbaállítás, 423–425
 többesküldés, 428–431
 vezérjeles vödör, 418–420
 többcélú forgalomirányítás, 406
 többcélú hálózati levelezéskiterjesztés, 693
 többes küldés, 311
 többesküldés, 26, 408
 többesküldéses forgalomirányítás, 408
 többesküldéses router, 798
 többmódusú szál, 111
 többprotokollós routerek, 433
 többszámítógépes rendszerek, 27
 többszörös elérési csatorna, 271
 többszörös hozzáférés digitális érzékeléssel, 300
 többszörös hozzáférés ütközések elkerülésével, 293
 többszörös hozzáférés, 461
 többszörösen beiktott hálózatok, 465
 töredék, 302
 töredéksorozat, 302
 törzs, 687
 TPDU, 522
 transzformációk, 773
 transzparens híd, 342
 transzponder, 191
 tréfás közleményre utaló jelzések, 714
 TSAP, 526
 túltöltés, 756
 tüzel, 252
 tűzfal, 446
 UBR, 496
 UDP, 57, 579
 UNI, 487
 URI, 735
 URL, 732
 USENET, 710
 UTP, 107
 uucp, 710

üresjáratú cellák, 264
 űrlapok, 742
 ütközés feltételezése, 274
 ütközés, 274
 ütközésérzékeléses CSMA, 281
 ütköztetési tartomány, 317
 üvegszál
 az elosztó dobozig, 792
 otthonra, 792
 üzenetátviteli rendszer, 823
 üzenetkapcsolás, 158
 üzenetkézbesítő ügynök, 685
 üzenetkezelő rendszer, 823
 üzenetszórás vihar, 594
 üzenettovábbító ügynök, 824

V.24, 138
 V.32 bis, 136
 V.34, 136
 V.42 bis, 136
 válaszablak, 322
 választott nyílt szöveg típusú, 621
 válogatás nélküli üzemmód, 338
 valós környezet, 809
 változó bitebesség, 495
 vámpír csatlakozó, 307
 vastag Ethernet, 307
 véges állapotú automata, 247
 végközpont, 127
 végrendszer, 29
 végtelenig számolás problémája, 393
 vékony Ethernet, 307
 vektorkvantálás, 774
 véletlen elérésű/hozzáférésű csatorna,
 271
 véletlen hozzáférésű csatorna, 298
 versenyhelyzetes rendszer, 275
 veszteséges, 772
 veszteségmentes, 772
 vételi ablak, 232
 vezérjel, 318, 325
 vezérjel-körbefutási időzítő, 354
 vezérjeles gyűrű, 324
 vezérjeles sín, 318
 vezérjeles vödör algoritmus, 418

vezérjelkezelés, 52
 vezérjeltartási idő, 328
 vezérjeltartási időzítő, 354
 vezérlő és karbantartó, 264
 vezérlő folyamat, 820
 vezérlő objektum, 817
 vezeték nélküli telefon, 184
 vezetékes szolgáltató, 187
 vezeték nélküli hálózat, 32–34, 117–125
 analóg rádió, 184–189
 digitális rádió, 189–190
 elektromágneses hullámok, 118–120
 mobil hoszt, 402–405, 468–470
 vezeték nélküli LAN, 291–295
 vezeték nélküli TCP, 580–582
 video, mozgókép, 767–770
 analóg, 767–769
 digitális, 769–770
 progresszív, nem sorugrásos, 768
 sorugrásos megjelenítés, 768
 videokiszolgáló, 786
 videokonferencia, 23
 Világháló, 721–763
 böngésző, 722
 élőkapocs, 722
 hipermédia, 724
 keresőgép, 760–763
 külső megjelenítő, 724
 HTML, 732, 735–742
 HTTP, 730–731
 Világháló, 75
 Villamos- és Elektronikai Mérnökök
 Szervezete, 93
 virtuális áramkör, 857
 kapcsolt, 857
 állandó, 857
 felépítése, 861–862
 lebontása, 861–862
 virtuális áramkörök, 377
 virtuális csatorna, 486
 virtuális eszköz, 810
 virtuális fájl tároló modell, 821
 virtuális hálózati terminál, 53
 virtuális terminál környezet, 816
 virtuális terminál protokoll, 815

virtuális ütemező algoritmus, 501
 virtuális útvonal, 487
 visszacsatolásos kódolóval, 630
 visszahajlított cső, 361
 visszairányú továbbítás, 407
 visszajátszásos támadás, 647
 visszajelzés, 685
 „visszalépés n-nel”, 237
 visszatükrözéses támadás, 643
 visszhangelnyomó, 136
 visszhangtörölő, 138
 VISTAnet, 77
 viszony kulcs, 641
 viszony réteg, 52
 viszony-forgalomirányítás, 381
 vivőjel-érzékelés, 274
 vivőjel-érzékeléses többszörös
 hozzáférés, 279

vonalkapcsolás, 155
 VSAT, 193

WAN, 29
 WDM, 144
 WDMA, 289
 WWW, 75, 721

X.21, 81
 X.25, 81
 X.28, 82
 X.29, 82
 X.3, 82

zaj, 133
 Zipf-féle törvény, 787
 zónák, 668

