

Számítógép hálózatok (6. rész)

A távbeszélő rendszerek, modemek, az adatkapcsolati réteg és protokolljaik.

A távbeszélő rendszer hasznos dolog, főként azért, mert jó a kiépítettsége. Ha azonban számítógépeinkkel ezen keresztül szeretnénk kapcsolatot tartani egymással, modem is szükséges hozzá. Miután lerántottuk a leplet e misztikus szerkezetek működéséről, befejezzük a fizikai réteggel való foglalkozást, és egy szinttel feljebb lépünk. A telefonhálózatot annak idején az emberi beszéd továbbítására találták ki. Mivel 1876-ban még nem volt számítógép, de digitális technika sem volt, ezért a telefonhálózatok analóg rendszerként működtek, azaz a hangot a feszültség (vagy egyéb más fizikai jellemző) időbeli változtatásával továbbították. A számítógépek sajnos nem tudnak ilyen módon kapcsolatot tartani egymással, ugyanis az átviteli közegek korántsem tökéletesek. Ez azt jelenti, hogy az analóg jel útközben megváltozik, azaz nem fog megegyezni azzal, amit a forrás küldött. Digitális adatátvitelkor csupán két szintet használunk, amelyeket akkor is meg lehet különböztetni, ha a jelet viszonylag nagy torzulás éri.

A digitális hálózatok sok szempontból előnyösebbek az analógokénál (lásd előző rész), ezért a telefontársaságok is kezdtek áttérni a digitális átvitelre. Ma már a legtöbb helyen a tönkők között digitális vonalakat építenek ki.

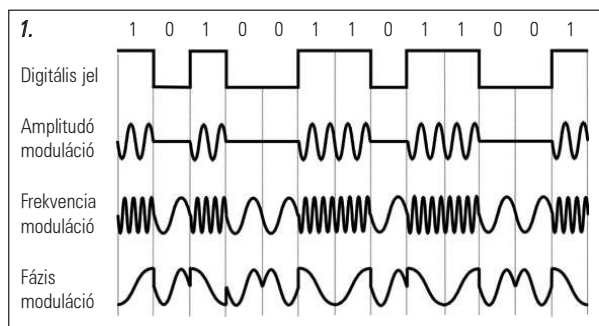
A távbeszélő hálózat gerince tehát hiába digitális, ha rajta keresztül két számítógép szeretne kapcsolatot tartani, szükség lesz egy olyan szerkezetre, amely a számítógépről érkező digitális adatokat analóg jellé alakítja (és ugyanezt fordítja).

A modem

Ez az eszköz a modulátor–demodulátor, vagy közismertebb nevén a modem. Fontos, hogy a modem nem egy analóg–digitális, illetve digitális–analóg átalakító. Valójában a bemenetre érkező digitális jelek alapján modulál egy azaz megfelelő analóg jelet, illetve a beérkező analóg jeleket demodulálja.

Vajon mit is csinál a moduláló? Az analóg jeleket nehéz kezelni, mivel frekvenciájuk, amplitúdójuk és nem utolsósorban fázisszögeik is vannak, és mindezeket az adatok továbbítására is felhasználhatjuk. Digitális jelet úgy továbbíthatunk, ha az adott digitális jel függvényében megváltoztatjuk az analóg jel valamelyik jellemzőjét. Ezt a tevékenységet modulációnak nevezzük.

Az 1. ábrán példát láthatunk különböző modulációs eljárásokra. A felső sorban a négyzöghullámot, azaz magát a digitális jelet látjuk. A második sorban láthatjuk, hogy a jel „átesett” egy amplitúdó-moduláción, ennek során az analóg



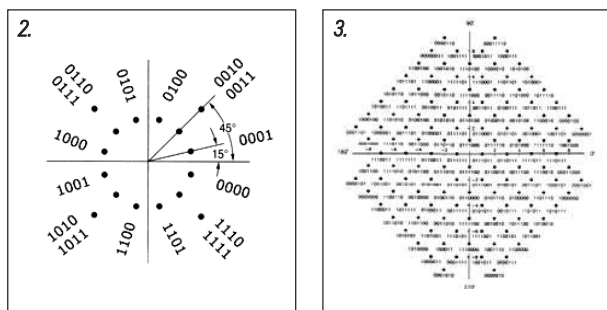
jel amplitúdóját változtatgatták a digitális jelnek megfelelően. Ebben az adat az amplitúdó nagyságában rejlik. Külön értéket rendeltek a 0-s, és külön értéket az 1-es digitális jelhez.

A frekvencia-modulációnál a frekvenciát, a fázisszög-modulációnál pedig értelem szerűen a fázisszöget változtatgattuk. A modemek többféle modulációs eljárást alkalmaznak, így egyszerre több bitet képesek átküldeni. Például úgy, hogy a 0, 90, 180, 270 fokos fázisokhoz két különböző amplitúdót rendelnek (amplitúdó alatt most az origótól vett távolságot értjük). Ez azt jelenti, hogy 8 kombinációban modulálhatunk hullámokat, azaz egy jelváltással 3 bitet vihetünk át.

A 2. ábrán egy kicsit jobb modulációs sémát vehetünk szemügyre. Itt 16 különböző amplitúdójú és fázisú értéket használhatunk, azaz már 4 bitet is átvihetünk jelváltásonként. Ezt kvadratúra amplitúdó-modulációnak nevezzük, és ezt alkalmazták a 9600 bit/s-os sebességű modemek is.

A 2. ábrához hasonlókat amplitúdófázis-diagramoknak vagy csillagkép mintázatoknak nevezzük. Minden modem rendelkezik egy ilyen mintázattal, és általában csak a megegyező mintázatú modemek képesek egymással kapcsolatba lépni. (Habár a gyorsabb modemek tudják emulálni a náluk lassabbakat). A 3. ábrán a V.32 bis-nek nevezett szabványt láthatjuk, ezt használják a 14 400 bit/s-os modemek is.

A 3. ábrán szemmel láthatóan sokkalta több pont helyezkedik el, mint a másodikon. Amikor ilyen sok pontot tartalmaz a csillagkép mintázat, akkor elég egy nagyon kicsi zaj, és máris fennáll az átviteli hiba veszélye. Ráadásul egyszerre 6 bitet kapunk hibásan, mivel ennyi bitet tudunk átvinni jelváltásonként. A gyorsabb modemeknél a helyzet sokkal súlyosabb lehet, ezért a zajszint csökkentése érdekében a rendelkezésre álló 3000 Hz-es sáv szélességet felbontják 512



keskeny csatornára. Ha egy csatorna túlságosan zajossá lesz, akkor egyszerűen ki kell iktatni. Az ilyen megoldások azt feltételezik, hogy a modem maga is processzorral rendelkezik, így nemcsak a sebességüket, de árukat tekintve is magasabb kategóriába tartoznak.

A mai modemek már maguk is tartalmaznak hibajavító és tömörítő szolgáltatásokat, ezáltal nem kell módosítani a meglévő programokat nagyobb sebesség elérése érdekében. A tömörítés alapfeltétele a hibajavítás, ezért általában ezt a két feladatot szokás foglalni. A legnépszerűbb ilyen szabványok a Microcom által kifejlesztett MNP és a V42bis.

Érdekes, ám a modemek életét megkeserítő jelenség a visszhanghatás. Az elektromágneses hullámok is visszaverődnek. Ilyesmi történik például akkor is, mikor telefonálás közben kis késleltetéssel saját hangukat halljuk vissza.

A hullámok ugyanis visszaverődnek az előfizetői hurok lezárásánál. A visszhanghatás igazából csak akkor érzékelhető igazán, ha nagy távolságra van tőlünk, akivel beszélgetünk. Ha ugyanis kicsi a távolság, akkor a visszaverődő jel hamarabb visszaér hozzánk, mint észrevennénk.

A gond valójában nem a jelenséggel, hanem az őket csillapítani próbáló visszhang-elynyomókkal van. Ugyanis sok embert zavar, ha telefonálás közben saját hangját hallja vissza, ezért a telefontársaságok a 2000 km-nél hosszabb vonalakra visszhang-elynyomókat telepítettek. Ez egy olyan szerkezet, amely képes érzékelni, ha az egyik irányból emberi hang érkezik. Ilyenkor nem engedi át a másik irányból érkező jeleket, ezáltal megszűnik a visszhanghatás. Amikor az egyik fél befejezte a mondandóját, és a másik fél veszi át a szót, akkor a visszhang-elynyomó irányt vált.

Az lehet, hogy magunkat már nem halljuk a telefonban, de ezzel duplex átviteli kísérleteinket is keresztülhúzták.

A duplex átvitel azt jelenti, hogy adatokat mindkét irányba egyszerre továbbíthatunk, például olyan módon, hogy a két irány különböző frekvenciatartományt használ. Ha visszhang-elynyomót telepítettek a rendszerbe, akkor csak fél-duplex átvitel oldható meg, ami olyan mint az egypályás vasúti sín: egy időben csak az egyik irányba mehetnek rajta az adatok.

Sok országban ezért úgy módosították a visszhang-elynyomókat, hogyha tisztán 2100 Hz-es hangot hallanak, akkor maguktól kikapcsolnak. De jelenleg nagyon kevés helyen találunk ilyen, ugyanis a telefontársaságok attól félnek, hogy a felhasználók így megzavarhatják a rendszer működését.

A visszhang-elynyomók helyett inkább visszhangtörőket használnak, amelyek megbecsülik a visszhang nagyságát, és az átmenő jelből „levonják”.

Külső vagy belső?

A modemeket két alapvető csoportba foglalhatjuk. Az első a külső modemek családja, ezeket a szabványos soros kárpun keresztül kapcsolhatjuk a számítógéphez. A modem és a számítógép között az átvitel digitális módon zajlik. Ennek módját az RS-232-C nevű szabvány rögzíti, mely egy 25 tűs csatlakozót ad meg. Az összes tűhöz tartozik egy szolgáltatás, de valójában mindenhol csak 9 kerül megvalósításra. A legtöbb esetben a többi tűt el is hagyják. A logikai 0-t a -3 V - 15 V tartományba eső feszültségek, míg a logikai 1-et a +3 V és +15 V közötti feszültségek jelentik. Egy modemet és egy számítógépet összekötő RS-232-es kábel legnagyobb hossza 15 méter lehet, és nem igazán érhetünk el rajta 20 kb/s-nál nagyobb sebességet.

Nézzük, miként is zajlik az adatátvitel az RS-232 szabvány szerint. Mindegyik tű egy-egy feladatot lát el, például valamit jelez. A 20-as tű azt jelzi, hogy a számítógép be van kapcsolva. A modem készenlétét a 6-os tű jelzi. Például, ha a számítógép adatot kíván küldeni a modemnek, akkor aktívulódik az adáskérés jel, erre a modem az adásra kész jellel válaszol. Ezután megkezdődik a 2-es tűn az adatok vétele. Látható, hogy ez nem egy túlzottan bonyolult protokoll. Két számítógépet összeköthetünk az RS-232-C csatlakozóján, azaz a soros kapujukon keresztül. Ilyenkor egy nullmodem-kábelre van szükség.

A külső modemek külön lyukat igényelnek a konnektorbán, ráadásul az asztalon is helyet foglalnak, ezért idővel megjelentek az úgynevezett belső modemek, amelyek ISA vagy PCI csatlakozófelülettel bíró kártyák. Az ilyen modemek vezérlői sokkalta bonyolultabbak, mivel nemcsak a hagyományos modemfeladatokat kell megvalósítaniuk, hanem például a csatlakozófelület kezelését is.

Az ilyen vezérlő nagyon drága, a piacon pedig éles verseny folyt, így a gyártók arra törekedtek, hogy az ő termékük legyen a legolcsóbb a piacon. A költségeket úgy lehetett csökkenteni, hogy a vezérlőkre csak a legalapvetőbb feladatokat bízta, a többi a processzorral, azaz programból megvalósított módon valósították meg. Nagy hátránya az ilyen eszközöknek, hogy a megfelelő kezelőprogram nélkül semmit sem érnek. Ráadásul ezeket a programokat először csak Windowsra készítették el, így más rendszer (például Linux) alatt használhatatlanok voltak. Ezért rájuk aggatták a WinModem nevet.

Adatkapcsolati réteg

Legyen most már elég a fizikai rétegből és a műszaki nehézségek körbejárásából. Lépünk egy szinttel feljebb és nézzük meg, miként is működik az adatkapcsolati réteg, amelynek nincs más feladata, mint a megbízható és hatékony kapcsolat létrehozása két szomszédos számítógép között. A „szomszédos” szó azt sejteti, hogy a két masina egymáshoz elég közel, például egy közös szobában helyezkedik el és vezetékkel össze vannak kötve. Valójában nem az a fontos, hogy egy szobában vannak-e, illetve hány vezetékkel kapcsolódnak egymáshoz. Az sem baj, ha két különböző földrészen található és telefonhálózaton keresztül, modemek segítségével tartják a kapcsolatot. A hangsúly azon van, hogy a kapcsolattartás csatornája „vezetékyszerű”, azaz minden bit a küldés sorrendjében érkezik meg.

Az egyik gép egymás után küldi a biteket, a másik fogadja őket, és nem fordulhat elő olyan eset, hogy egy bit megelőzzön egy másikat. Ez a világ legegyszerűbb protokollja. Vajon miért kell foglalkoznunk ezzel? Igazából nem lenne semmi gond, és egyből ugorhatnánk is a hálózati rétegre, ha minden tökéletes lenne: az adatok nem vesznének el, nem lenne az adatátviteli sebességnek korlátja, minden bitet késleltetés nélkül tudnánk küldeni, és végül de nem utolsó sorban minden számítógép megegyező sebességgel dolgozna. De sajnos a dogok nem ilyen egyszerűek. Mivel az adatokat csak véges sebességgel tudjuk továbbítani, és nem nulla késleltetéssel lehet küldeni biteket, komoly meggondolásra kényszerít bennünket, ha valóban hatékony hálózatot szeretnénk létrehozni. Arról nem is beszélve, hogy rendkívül kellemetlen, ha a forrás gyorsabban küldi az adatokat, mint ahogy azt a vevő gép feldolgoni képes. Az átviteli hibák is gondot okozhatnak, főleg, ha nem tudjuk felismerni és kijavítani őket. Ezeket a felsőbb rétegekre is bízhatjuk, de ha túl sok van belőlük (mert mondjuk, rendkívül zajos a csatorna), teljesen használhatatlanná válna hálózatunk, hiszen lehet, hogy csupán egy bajt kellene újra küldeni, és nem az egész 1 KB-os csomagot.

Az adatkapcsolati réteg szolgáltatásai

Sorozatunk legeslegelső részében már volt róla szó, hogy minden réteg a felette levőnek végez el bizonyos feladatokat. Ezért minden réteg rendelkezik egy jól meghatározott csatolófelülettel, a felsőbb réteg ezen keresztül tartja a kapcsolatot vele.

Sok hálózati megvalósítás nem tesz különbséget bizonyos rétegek között, például az első három réteg gyakran egy helyen, a gép belsejében található. Mi azonban azt mondtuk, olyan módon képzeljük el a dolgokat, hogy minden réteg egy teljesen különálló részt alkot, például mindegyik egy-egy folyamat, amelyek egymással párhuzamosan futnak. Ez a szemlélet sokkal könnyebben átlálhatóvá teszi a hálózatok működését.

Azt se felejtjük el, hogy minden réteg csak egymással áll kapcsolatban, vagyis az A gépen futó adatkapcsolati rétegfolyamat nem fog soha kapcsolatba lépni a B gép hálózati rétegével. Másik fontos dolog, hogy a rétegek sosem foglalkoznak azzal, hogy az alattuk levő réteg mit csinál. Az adatkapcsolati protokoll tehát nem fog azzal törődni, hogy az adatok milyen csatornán is jutnak át a másik géphez. Ő csak annyit tud, hogy az általa elküldött bitsorozat a másik gépen – a küldés sorrendjében – meg fog jelenni.

Ennek fényében nézzük meg, milyen szolgáltatásokat is nyújt a hálózati rétegnek. Az, hogy melyek ezek, rendszerrel változik, de három általános szolgáltatást mindenhol megtalálhatunk: nyugtázatlan összeköttetés nélküli szolgálat, nyugtázott összeköttetés nélküli szolgálat, nyugtázott összeköttetés alapú szolgálat. Ezek közül az első a legegyszerűbb szolgáltatás. Itt tulajdonképpen nincs másról szó, mint két gép úgy küld egymásnak kereteket, hogy nem vár semmiféle visszajelzést. Nem foglalkoznak így sem a kapcsolat felállításával, sem annak lebontásával. Ennek egyszerű a következménye, hogyha egy keret a vonalon fellépő zaj miatt megsérül, esetleg el is vész, a forrás erről sosem fog értesülni, és a sérült rész ismételt küldésére sincs mód. Egyértelmű, hogy ez

a fajta összeköttetés csak nagyon jó minőségű csatorna esetén használható, amelyen viszonylag kevés a hibaarány. Ha kicsi a hibaarány, elegendő a felső rétegekben elvégezni a hibakezelést.

Ezt a fajta összeköttetést érdemes használnunk akkor is, amikor inkább az a fontos, hogy az adatok ne késsenek, minthogy hibátlanul érkezzenek, például a beszéd- vagy videóátvitelnél. Ha akadozik az átvitel, a műsor élvezhetlenebb, mintha egy-egy képkocka hibásan érkezik át. Ennél kicsit megbízhatóbb szolgálat a nyugtázott összeköttetés nélküli szolgálat. Itt már a megérkezett kereteket nyugtázni lehet, így a forrás értesülhet róla, hogy sikeresen célba ért-e az általa küldött adatok. Ha egy nyugta nem érkezik meg adott időn belül, akkor a kérdéses keretet újra el kell küldeni.

Nagyon-nagyon fontos, hogy az adatkapcsolati rétegnek valójában nem lenne feladata a nyugtázás. Igazából nem követeli meg tőle sem az OSI, sem egyéb más szabvány. Az egész csak a hatékonyság szempontjából hasznos. A helyi hálózatokon a gépek üvegszál vagy koaxiális kábelekkel vannak összekötve egymással, ezért nagyrészt a nyugtázatlan összeköttetés nélküli szolgálatot választják. Ha viszont vezeték nélküli eszközöket használunk, a nyugtázott összeköttetés nélküli szolgálatot érdemes alkalmazni. Létezik egy harmadik szolgálat is, mégpedig az összeköttetés alapú. Az összeköttetés alapú azt jelenti, hogy a forrás és célgép az adatcsere megkezdésekor felépítenek egy kapcsolatot. Amikor ez a kapcsolat létrejön, az adatkapcsolati réteg felelősséget vállal azért, hogy minden elküldött keret megérkezik, de pontosan csak egyszer. A nyugtázatlan összeköttetés nélküli szolgálatnál elképzelhető, hogy egy-egy keret elvész. A nyugtázott, ám összeköttetés nélküli szolgálatnál az elveszett keret ugyan pótlódik, de a nyugta nem, így ha az vész el, akkor egy keret akár kétszer vagy többször is megérkezhet. Az összeköttetés alapú szolgálat esetében minden keret számot kap, és az összes keret a küldés sorrendjében fog megérkezni.

Látni fogjuk, hogy az összeköttetés alapú kapcsolat megvalósítása nem egyszerű. Az ilyen átvitelek mindig három részre bonthatók: a kapcsolat felépítése, a tényleges adattovábbítás és a kapcsolat lebontása. Az elsőben a kezdő értékek inicializálódnak, ezek szükségesek hozzá, hogy a fent leírtakat be- tarthassuk. Ehhez viszonylag sok változót és számlálót szükséges alkalmaznunk, amelyek erőforrásokat igényelnek, és a kapcsolat lebontásakor fel kell szabadítanunk őket.

Ezt a szolgálatot gyakran használják a forgalomirányításért felelős kódok, ha nem akarnak foglalkozni azzal a lehetőséggel, hogy esetleg az általuk küldött csomag valahol elveszhet. Írásomban röviden összefoglaltam, milyen szolgáltatásokat kell nyújtania az adatkapcsolati rétegnek a hálózati réteg felé. A következők részben a megvalósítás kérdéseit taglaljuk, valamint szót ejtünk a keretezésről, a hibajavításról, illetve a forgalomszabályozásról is.

Garzó András (garzoand@interware.hu)

Körülbelül három éve foglalkozik Linux- és más Unix-rendszerekkel. Legjobban az operációs rendszerek lelkivilága érdekli, de nyitott egyéniség. Kedvenc étele a palacsinta, és van egy Richard nevű macskája. Minden észrevételt, megjegyzést, levelet szívesen fogad.