



Általános
INFORMATIKAI
Tanszék

Számítógépek, számítógép rendszerek

4. A hálózatok, az Internet

Dr. Vadász Dénes

Miskolc, 2005. február

TARTALOM

TARTALOM.....	a
4. Hálózatok, az Internet	1
4.1 Az Internet története	1
4.1.1. Az ARPANet	1
4.1.2. Az RFC-k (Request for Comments)	1
4.1.3. A korai protokollok	2
4.1.4. A történet folytatódik	3
4.1.5. További protokollok	3
4.2. Az Internet Magyarországon	4
4.3. Csomópontok azonosítása az Internet-en	6
4.4. Az elektronikus levelezés alapfogalmai	9
4.5. Az Internet gopher	12
4.6. A File Transfer Protocol FTP	13
4.7. Az Archie szolgáltatás	13
4.8. Hogyan kereshetünk személyeket, számítógépeket?	14
4.9. A World Wide Web (WWW) és nézegetők	14
A WWW programozási nyelve: a Java	16

4. Hálózatok, az Internet

Az egyedülálló számítógép hasznos az élet minden területén. Hálózatba kapcsolva a számítógépek még hasznosabbak. Korábban a hálózatosság legfőbb oka az erőforrás-megosztás, az erőforrás-összevonás volt, manapság ezt kiegészíti a számítógépes kommunikáció (Computer Mediated Communication). A Internet világ kiteljesíti ezt a paradigmát, hiszen azt mondhatjuk az Internet a hálózatok hálózata, ahol egy hálózat egy csomópontjának felhasználója földrajzi és politikai határokat figyelmen kívül hagyva kommunikálhat bármelyik hálózat felhasználójával, bármelyik csomópont kérhet vagy biztosíthat szolgáltatásokat másik csomóponttól, csomópontnak (hacsak biztonsági okokból nem tiltják külön a hozzáféréseket).

A kapcsolatban lévő hálózatokat sem lehet felsorolni, nemhogy az Internetre csatlakozó gépeket. Azt sem lehet megállapítani, hogy hány hálózat, hány csomópont tartozik az Internethez, hány Internet felhasználó van a világon, csak az mondható biztosan, hogy állandóan növekszik. Illetve az is biztosan állítható, hogy a kapcsolatban lévő csomópontok, illetve a felhasználók számát tekintve az Internet a legnagyobb hálózat a világon.

4.1 Az Internet története

4.1.1. Az ARPANet

A 60-as években az USA-ban a Defense Advanced Research Projects Agency (DARPA) támogatásával kutatás indult, melynek kettős célja volt. Az egyik cél az volt, hogy telefonhálózaton keresztül csomagkapcsolással lehetne-e számítógépeket hálózatba kapcsolni, hogy egy vonalat több felhasználó is használhasson "egyidőben". A másik cél - nem kevésbé fontos - pedig, olyan hálózatot készítsenek, amelyik működőképes marad akkor is, ha háborús események miatt a hálózat egy része el is pusztulna: ha kiesnek bizonyos csomópontok, vonalak, automatikus átirányítással más vonalakon, csomópontokon továbbíthatók legyenek az üzenetek.

1969-re a kutatás-fejlesztések eredményeként - néhány (University of California at Los Angeles, Stanford Research Institute, University of California at Santa Barbara, University of Utah) helyszín összeköttetésével - kialakult az ARPANet. 1971-re 15, 1972-re 37 növekedett ez a szám, ami az ARPANet sikerét jelezte. Bár a támogató hadügyminisztériumának (Department of Defense, DoD) eredeti célja a távoli helyszínek közötti megbízható, robusztus összeköttetések megvalósítása volt, a kutatás-fejlesztésben résztvevők - mivel nagyon kényelmesnek találták - elkezdték a hálózatot személyes üzenetváltásokra is használni, és például az elektronikus levelezés az ARPANet segítségével hallatlanul népszerűvé vált.

Az az elv, hogy nincs a hálózatnak külön központi üzemeltető gépe (vagy gépei) valóban hozta a tervezett előnyt: nem omlik össze a hálózat részeinek meghibásodásával. Hátrány is következett persze ebből: meglehetősen nehéz a hálózaton történő navigálás. A különböző gépek különböző operációs rendszerei és felhasználói felületei, parancsértelmezői gondot jelentettek. Ebben a korai időszakban még messze nem voltak meg a ma jól ismert egységesített navigációs eszközök, böngészők, keresők.

4.1.2. Az RFC-k (Request for Comments)

Az ARPANet kezdeti fejlesztési időszakában nemigen voltak még hálózati szabványok. Ezért a kutató-fejlesztők kitaláltak egy meglehetősen informális módszert a "szabványosításra", az

RFC-k módszerét. Ha valakinek volt valamilyen javaslata valamilyen megoldásra, akkor közzétette ezt egy ún. előzetes RFC-ben (draft RFC), az ARPANet társadalom megvitatta, kommentálta, javította a javaslatot, és végül megegyezéssel elfogadta az RFC-t. Ekkor az RFC sorszámot kapott és ezzel szabvánnyá (Internet Standard) vált: a számával lehet hivatkozni rá. Az első RFC-t 1969-ben S. Crocker publikálta. Manapság az RFC-k száma meghaladja a 2000-et. A módszer sikerét az akadémiai kutatók együttműködésre való hajlama segítette: bárki tehetett előzetes javaslatot, a közzétett javaslatot bárki kritizálhatta, javíthatta, széleskörű volt e megegyezés a végleges szabványról. Maga az ARPANet pedig a "szabványosítás" folyamatát gyorsította, a vita és megegyezés a hálózat segítségével történhetett.

4.1.3. A korai protokollok

A korai 70-es években az ARPANet-re meglehetősen különböző számítógépeket csatlakoztattak. Minden helyszínen volt egy *interface message processor* (IMP) csomópont, egy kiskapacitású miniszámítógép, a helyszínek közötti összeköttetéshez, de a helyszín bármelyik számítógépét rákapcsolhatták a hálózatra. Eleinte a TCP/IP protokollszöveget még nem volt kialakítva, egy korlátozott lehetőségeket biztosító hálózatvezérlő programcsomagot használtak az ARPANet-ben. A szállítási réteghez tartozó protokoll az Network Control Protocol (NCP) volt.

1974 májusában V. Cerf és R. Cahn cikke az IREE Transactions on Communication-ban, A Protocol for Packet Network Interconnecting címmel az első javaslat a TCP protokollra. Ezután következett az RFC szabványosítás: a TCP szabvány az RFC-793-ban található. Az *átvitelvezérlő protokoll* (Transmission Control Protocol, TCP) megbízható csatlakozásorientált protokoll.

A *hálózatok közötti protokoll* (Internet Protocol, IP) a üzenetsomagok továbbításra szolgáló protokoll, az RFC-791-ben rögzített. A TCP és az IP az Internet legismertebb protokolljai, összefoglaló betűszavuk a TCP/IP. Függetlenek a hálózat alsóbb rétegeitől, a fizikai közegtől.

A TCP/IP protokollszöveget annyira összekapcsolódott az Internettel, hogy néha el is felejtkezünk arról, hogy más protokollok is tartoznak az Internet protokoll szöveghoz. Nem részletezve tekintsük át a legfontosabbakat, a "hagyományos" szolgáltatásokra való pillantással.

Egyik korai szolgáltatás az már az ARPANet korában a *számítógépes levelezés*. A SMTP (Simple Mail Transfer Protocol, RFC-821) és a TMP (Text Message Protocol, RFC-822) a legalapvetőbb dokumentumok (1982-ben keletkeztek). Az RFC-937 már a mikroszámítógépes személyi levelező ügynök és a kiszolgáló gép közötti kapcsolatot szabványosítja, az RFC-1521 pedig a MIME (Multipurpose Internet Mail Extension) leírása.

Szintén korai szolgáltatás a *távoli bejelentkezés*. A *telnet* hálózati terminál protokoll az RFC-854--860 szabványokban rögzített. Általában egy ismert terminál emuláció és egy kapcsolatépítő a legfontosabb alkalmazási szoftver elemek. A kapcsolatépítéshez a telnet-nek a távoli csomópont azonosítóját (címét vagy nevét) megadva a kiépülő kapcsolat nagyjából egy közvetlen vagy modemes vonali kapcsolathoz hasonlít: minden begépelte karakter a távoli rendszerhez kerül. A távoli rendszer a viszony (ülés, session) létesítéshez rendszerint a bejelentkezési eljárást (login procedure) biztosítja, ha van számlaszámunk a távoli gépen, akkor azonosítónk és jelszavunk megadásával létesíthetünk viszonyt, ülést a kapcsolaton.

További alapvető szolgáltatás az *állománytovábbítás*. Protokollja az állománytovábbító protokoll (File Transfer Protocol, FTP, RFC-959). A szolgáltatásokat az *ftp ügynök* programok segítségével igénybe vehetjük.

4.1.4. A történet folytatódik

1975-ben az ARPANet fejlesztés a Defense Communication Agency (DCA) felügyelete alá kerül. 1982-ben elkezdve, 1983 január 1-jére az ARPANet teljesen áttért a TCP/IP protokollszövegre. Ekkor kezdték használni az *Internet* kifejezést - az IP protokoll nevéből származtatva a szót - azt is jelezve ezzel, hogy az Internet hálózatok hálózata, hiszen az ARPANet-hez egyre több hálózat kapcsolódott, most már átjárók nélkül, mert azok is a TCP/IP szöveget használták (pl. a Bitnet). Megkezdődött az Internet rohamos, megállíthatatlan növekedése.

1983-ban az ARPANet-Internet két hálózatra vált szét: MILNET-re és ARPANET-re. Talán ez az esemény az Internet igazi születésnapja, hiszen a szétválás lehetővé tette az USA hadügyminisztérium által eddig nem támogatott intézmények Internetre való csatlakozását is.

1984-ben állították be az első névszolgáltatót (domain name server). Kialakult a japán JUNET, az Egyesült Királyságban a JANET. 1986-ban az NFSNET is elindult. 1988 az Internet *féreg* éve: az Internet mintegy 6000 számítógépét "támadta meg" a *worm* nevű számítógép program. A hatására alakították ki az RFC-1087 azonosítójú Ethics and the Internet dokumentumot.

1989-ben alakult az Internet Engineering Task Force (IETF) az Internet Activities Board (IAB) alá rendelve. 1990-ben az ARPANet hivatalosan megszűnt. Ez az év azért is jelentős, mert megjelent az Archie: az ftp-ző felhasználók ennek segítségével kereshették, hol találják meg az érdeklődésükre számot tartó állományokat. 1991-ben alakult a Commercial Internet Exchange Association (CIX): tagjainak a saját hálózataikon ingyenes adattovábbításokat biztosítottak, az USA kormányzat által támogatott NFSNET korlátozásokat elkerülhették. 1991 a *gopher* megjelenésének is az éve. Említsük meg szerzői nevét: Paul Linder és Mark McCahill, University of Minnesota. Ugyanekkor jelenik meg az Interneten a WAIS (Wide Area Information Service)

1992-ben az Internetre kapcsolódó gépek száma meghaladta az 1 milliót. Megalakult az Internet Society (ISOC), azzal a céllal, hogy az Internet technológiák fejlesztését és használatát, a szabványosításokat kézben tarthassák. Ez az év az első World Wide Web (WWW) böngésző kibocsátásának éve. A CERN European Laboratory for Particle Physics, Genf, Svájc kutatója Tim Berners-Lee a WWW koncepció szülője.

1993-ban alakult az Internet Network Information Center (InterNIC), feladata a regisztráció, a szabványok RFC-k gondozása, információszolgáltatás az Internetről. Megjelenik az első grafikus WWW böngésző, a Mosaic. A képmegejelenítés, a hanglejátszás élménye miatt 1994 áprilisára a WWW forgalom felülmúlja a gopher forgalmat.

4.1.5. További protokollok

A már eddig említett alapvető szolgáltatásokhoz tartozó protokollok mellett további protokollok alkotják az Internet protokoll szöveget. A fontosabbakat megemlítjük ezek közül, megjegyezve, hogy ezek ügynök-szolgáltató (klient-server) jellegű szolgáltatási körbe tartoznak.

A *hálózati állományrendszer* (network file system) szolgáltatás segítségével az FTP módszert meghaladva lehet fájl-szolgáltatást biztosítani. Segítségével egy számítógép számára virtuális meghajtókat biztosítanak más rendszerek állományaiból. Takarékosági előnyök, közös állomány hozzáférés, könnyebb rendszerkarbantartás, archiválás lehetősége adott így, hogy csak a legfontosabbakat említsük az előnyeiből. A TCP alapú PC-kre való NetBIOS leírását az RFC-1001-1002 közli. Unix-os munkaállomásokhoz, szolgáltató gépekhez ma leggyakoribb a Sun Network File System (SUN NFS) hálózat fájlrendszere. A protokollokat a Sun Microsystems fejlesztette, szolgáltatja.

A *távoli nyomtatás* legszélesebb körben használt protokollja a BSD távoli sornymotató protokollja. Sajnos, dokumentált leírása nem létezik, de C nyelvű forráskódokhoz a megvalósításhoz hozzá lehet jutni.

A *távoli futtatás* hasznossága nyilvánvaló ha adott munka erőforrás-igényes részét erőforrás-gazdagabb gépen akarjuk elvégezni. A leggyakoribb megoldások a BSD *rsh* és *rexec* kiszolgálói, a man lapokon leírásuk megtalálható. A távoli eljárásívás legelterjedtebb protokolljai a Xerox cég Curier, ill. a Sun RPC. Leírásukat az adott cégektől kell megszerezni, de megemlítjük, hogy a BSD 4.3-tól kezdve ezek megvalósítása megvan (a Sun RPC csak részben).

A *névszolgáltatás*. Miután rengeteg nevet kell kezelni a nagykiterjedésű rendszerekben, ma minden TCP/IP megvalósításnak része kell legyen ez a szolgáltatás. Az érintett protokollokat az RFC-822-823 írja le. A Sun Yellow Pages rendszere - ma már Network Information Service (NIS) néven a felhasználók neveinek számlaszámainak, csoportjainak menedzselése mellett további szintet ad a névfeloldásra, Unix rendszerek által használt adatbázis kezelésre, szolgáltatás nevek kezelésére.

Terminál szerver szolgáltatás. A terminálszerverek kisebb teljesítményű speciális célú operációs rendszer alatt működő számítógépek. A már említett telnet protokoll mellett a névszolgáltatási protokollokat is kezelik, gyakran más protokollokat is ismernek (pl. távoli nyomtatási protokollokat, nem TCP/IP szöveget stb.). A terminálokat sokszor nem közvetlenül kapcsolják a gazdagépekre, hanem a terminálszerver portjaira, a terminálszerver végzi a kapcsolást, és miután rendszerint lehetséges egyszerre több aktív kapcsolat létesítése is, a szerver az aktív kapcsolatok közötti váltásokat is segítheti.

A *hálózati alapú grafikus megjelenítés* jelentősége azért nagy, mert így nem a gazdagéphez közvetlenül kapcsolt bit-térképes grafikus képernyőn is lehetséges a grafikus megjelenítés. Legszélesebb körben elfogadott szabvány az X11. Leírásukat több helyről is beszerezhetjük, az X11 ma a UNIX rendszerek természetes része. Azt is meg kell említeni, hogy az X11 a HTML terjedésével veszít jelentőségéből.

És végül megemlítjük, hogy az Internet szabványok aktív listáját megkaphatjuk az RFC-1011-ből.

4.2. Az Internet Magyarországon

Az Internet magyarországi története teljesen összefonódott az Információs Infrastruktúra Program és az azt követő Nemzeti Információs Infrastruktúra program történetével.

A program gondolata 1985-ben született, Vámos Tibor akadémikus merész kezdeményezésére. Gondoljuk meg, abban az időben, amikor minden hálózati termék szigorú embargó alá

esett, egy országos kutatói számítógépes hálózat létrehozását javasolta Vámos Tibor. Szerencsére, az OMFB, a Magyar Tudományos Akadémia vezetői felismerve a kezdeményezés jelentőségét biztosítottak megfelelő szervezeti és pénzügyi feltételeket: elindulhatott az IIF program, és első szakasza 1986 és 1990 között sikeresnek bizonyult.

1988-ra a Magyar Postánál üzembe helyezték a hazai fejlesztésű 80 vonalas csomagkapcsoló központot, és elkészült az ELLA elektronikus levelező program. 1989-ben az adathálózaton hazai és nemzetközi szolgáltatásokat biztosítanak, mintegy 100 végrendszer kapcsolódik a hálózatra. A kiépülő rendszer illeszkedett nemzetközi szabványokhoz (OSI, X.25). A szolgáltatások hazai (ELLA) és nemzetközi (EUnet) elektronikus levelezésre és távoli számítógép használatra, ezen keresztül adatbázisok lekérdezésére terjedtek ki. Más nemzetközi kapcsolatokat, szolgáltatásokat az embargó miatt nem lehetett igénybe venni.

1990-ben a politikai változások lehetővé tették a nemzetközi kapcsolatok bővülését, csatlakozhattunk az EARN-hoz (European Academic and Research Network). A Magyar Posta az adathálózat kapacitását növelte, megnyitotta a nyilvános csomagkapcsolt adatszolgáltatást. Az embargó a hálózati elemek behozatalát még mindig akadályozza, a helyi hálózatok összekapcsolásának eszközei hazai fejlesztéssel alakultak ki. Mintegy 200 intézmény, felsőoktatási intézmények, kutatóhelyek, könyvtárak és múzeumok, alkotják az IIF intézeteket. Tulajdonképpen sikerrel lezárul a program első szakasza.

A sikerre való tekintettel a program tovább erősödött, 1991-ben kibővült a támogatók köre, az OMFB-hez és az MTA-hoz támogatóként csatlakozott a Művelődésügyi és Köznevelési Minisztérium és az Országos Tudományos Kutatási Alap (OTKA). Az IIF második fázisa indul, kissé módosuló fejlesztési céllal. Most már nemcsak hazai fejlesztésű hálózati elemekre lehet építeni, és most már az egész országot lefedő hálózatra is lehet gondolni. Mivel 1988-tól az USA engedélyezte Európa számára is az Internet technológiákat, a meglévő európai felsőoktatási és kutatói hálózatok Internetre való csatlakozását, az európai infrastruktúrán keresztül nekünk is lehetőségünk nyílt az Internet használat. Megkezdődhetett a kísérleti Internet kapcsolatok felépítése, a névszolgáltatás biztosítása, az elektronikus levelezésben az Internet címzések használata. A MATÁV korszerűbb, import csomagkapcsoló központokat szerezhetett be, melyekkel az adathálózat minőségét, teljesítményét és a szolgáltatások körét bővíthették.

1992-re Unix-os gazdagépek kerültek a rendszerbe. Ezekhez "alapértelmezés" szerint tartozik a TCP/IP protokollszöveg és a névszolgáltató szoftverrendszer. 1993-ra Világbanki támogatással szinte az összes felsőoktatási intézményben elterjedtek a UNIX konfigurációk, korszerű helyi hálózatok alakultak ki. Elindult a HBONE, a hazai IP protokollú gerinchálózat kialakítása. Az IIF program finanszírozásából nem csak a fejlesztések folytak, hanem az intézetek ingyenesen vehették igénybe a hálózati szolgáltatásokat. Az IIF programban ekkor mintegy 450 intézmény vett részt. Sikerét elismerték mind a hazai, mind külföldi körökben.

1993-ban az IIF - ismét megújulva - a HBONE (országos bérelt vonalas, IP technológiájú gerinchálózat) megerősítését, az IP technológiákra épülő szolgáltatások (telnet, ftp, smtp, gopher, wais, archie, news stb.) hozzáférhetővé tételét, terítését tűzte ki célul. Az elgondolás az volt, hogy a szigetszerűen már kialakult vagy kialakuló helyi, nagy-forgalmú IP hálózatokat közvetlenül a gerinchálózatra kapcsolják, míg a többi intézmény, a kisebb elszórt felhasználók a nyilvános X.25. hálózat, legrosszabb esetben a nyilvános telefonhálózat közvetítésével érhesse el a gerinchálózatot. Nem elhanyagolható cél továbbá: a HBONE és a nagy nemzetközi hálózatok megbízható, nagy kapacitású vonalakkal való összekapcsolása is.

1995 folyamán a HBONE kialakult. A nagy-megbízhatóságú gerinchálózati mag Budapesten a KFKI-ban, az IIF központban és a BME-én elhelyezett router-ekből, az azokat összekötő 2Mbps sebességű mikrohullámú kapcsolatokból, a MATÁV Városház utcai központjában lévő router-ekből és kapcsolatrendszerükből állt. Utóbbinál kapcsolódnak a HBONE nemzetközi vonalai. Ugyancsak a MATÁV-nál hozták létre a Budapest Internet eXchange (BIX) csomópontot, melyet már tízegynéhány profitorientált Internet szolgáltató is finanszírozott. Ennek célja az volt, hogy a profitorientált szolgáltatók által menedzselt hazai felhasználók közötti forgalom a BIX-en keresztül cserélődjön ki, és ne terhelje egyik szolgáltató nemzetközi vonalait. Ez egy lényeges mozzanat: az IIF intézmények mellett a profitorientált intézmények is kacsintgatnak az Internetre. A HBONE a budapesti magból és mintegy 20 regionális központból állt 1995-96 fordulóján, és csak néhány regionális központ bekapcsolása volt hátra, hogy az összes megyeszékhelyen csomóponttal rendelkezzen. A vonalszélességekben is volt fejlődés, az öt legnagyobb forgalmú regionális központhoz (Veszprém, Pécs, Szeged, Debrecen, Miskolc) az 512 Kbps kapcsolatot 1996 őszén biztosították. A fejlődést mutatja, hogy 1996 áprilisára a HBONE névszolgáltatóinak száma meghaladta az 500-at, ugyanekkor a bejelentett elérhető gazdagépek száma meghaladta a 22000-et [Martos-Tétényi96].

Az 1996-os év több szempontból is jelentős. Egyrészt az IIF program átalakult NIIF programmá: Nemzeti Információs Infrastruktúra Programmá. Másrészt ez az év, amikor a profitorientált Internet szolgáltatók színre léptek és nagy sikereket értek el. (Feltétlenül meg kell említeni, hogy az IIF nélkül sikerük nem lett volna ilyen átütő, fejlődésük nem lett volna ilyen rohamos. Az IIF és a HBONE fejlesztés Magyarországon kialakította azt a szakembergárdát, amelyik az Internet technológiákat ismeri, a rendszereket működtetni tudja. Másrészt e rövid idő alatt is kialakult az Internet felhasználók széles köre: a mai főiskolai és egyetemi hallgatók képzésének eleme az Internet használat, egyre többen úgy lépnek ki az iskolákból, hogy ismerik és igénylik a hálózat szolgáltatásait. Sőt, ma már a középiskolák vannak a soron, megkezdődött ezek rákapcsolása is a Hálózatra.

4.3. Csomópontok azonosítása az Internet-en

A forgalomirányítók az IP címeket használják a csomagok továbbítása során. Az IP címek egyediek a csomópontokra (node-kra).

Példa:

193.6.10.1	gold	IBM R/6000 970	AIX
193.6.5.33	zeus	SGI Power Series	Irix

Hasznos feljegyezni fontos csomópontok IP címéit. A csomópontok neveit a rendszergazdák választják, nem egyediek.

Példa:

kuka az IIT-n:	VAX 2000,	VMS (már nincs!)
kuka a BME-n:	?	? (152.66.81.30)

A címek megjegyzése kellemetlen feladat. Jobban szeretjük a neveket használni parancsainkban. A megoldás tehát: név - IP cím feloldást kell biztosítani!

A névfeloldás alap gondolata: egy táblázatban rendeljük össze az IP címeket és a neveket. Ez a táblázat lehet az /etc/hosts fájl, vagy NIS rendszerben a megfelelő adatbázis. Felmerülnek azonban gondok! Ha van több mint 1 millió csomópont, mekkora lesz ez a tábla? És mi lesz az ismétlődő nevekkal (name collision)? És ha cserélődnek a nevek, címek, hogy biztosítható a naprakész állapot (consistency)?

A továbbfejlesztett megoldás célja (kissé leegyszerűsítve most) a következő: a helyi adminisztráció engedélyezett legyen (azaz a neveket lehessen viszonylag szabadon megválasztani), mégis biztosítsanak globális eléréseket.

1984-ben javasolták (RFC 822, 883; később RFC 1535,1536,1537) a **tartomány-név rendszer** (Domain Name System, **DNS**) hierarchikus osztott adatbázist, melynek helyi adminisztráció biztosított egy-egy szegmensén, de globális az elérése. Kliens-szerver koncepciójú. Ehhez bevezetették a *névszolgáltató* (name server) és a *névfeloldó* (resolver) fogalmakat.

A koncepció szerint felosztották a világot *tartományokra* (domain), *nevet* is adva a tartományoknak. A tartományok szervezésében az összefogó erő lehet a földrajzi elhelyezkedés, lehet az politikai összetartozás, gazdasági, kulturális, társadalmi, szervezeti hasonlóság is. *Csúcstartomány*nak nevezzük a legfelső szintű tartományokat (ezeknél az összefogó erő legtöbbször a közös államiság). Egy tartományon belül lehetnek

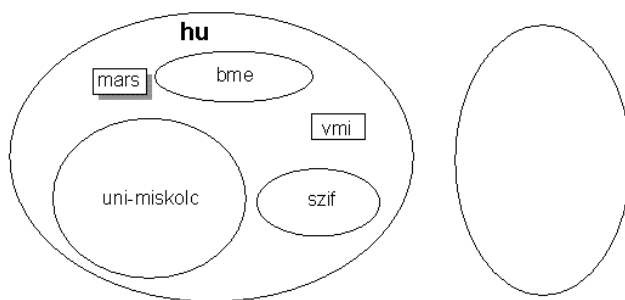
- *altartományok* (subdomain) a nevükkel,
- *csomópontok* (node-k) a nevükkel. Ezeknek IP címük is van.

Fontos, hogy a nevek a tartományon belül egyediek legyenek.

Az 4.1. ábrán láthatjuk a *hu* tartományt, benne *uni-miskolc*, és *bme* altartományokat, illetve a *mars* nevű csomópontot (természetesen a *hu* tartományban sokkal több altartomány és csomópont van, mint amit az ábrán láthatunk). A *hu* csúcstartomány.

A nevek szintaktikája ezután:

node-név.altartomány-név.csúcs-tartomány-név
 node-név.csúcs-tartomány-név
 altartomány-név.csúcs-tartomány-név



4.1. ábra. Példa tartományra

Látható, ez a névtér (domain name space) hierarchikus (max. 127 mélységig mehet). Egy egyszerű név (simple name) max. 63 karakteres lehet. A teljes tartománynév, csomópontnév pedig egyszerű nevek listája, a pont (dot) elválasztóval, egy tartománytól, csomóponttól a gyökér tartományig (csúcsig). (Ebben az absztrakcióban egy csomópont egy tartománynak felel meg!)

A tartománynév indexeli a DNS adatbázist! Az adatbázis pedig információ-

kat tartalmaz! Pl. egy gazdagépet indexelve a gazdagép (host) a hálózati címét (IP cím), HW jellemzőit, elektronikus levelezéshez útvonal-irányítási információit (e-mail routing) tartalmazza. Egy tartományt indexelve az adatbázis strukturált információkat tartalmaz a gyermekeiről!

Az információk hierarchikusan leosztottak és decentralizáltak: a decentralizálás eszköze a felelősség leosztás (delegation).

Az információkat a névszolgáltatók biztosítják, melyek a delegált szervezet felügyelete alatti gazdagépen futó programok (továbbiakban gazdagépet említünk névszolgáltatóként). Egy névszolgáltató teljes információkat szolgáltat a névtartomány egy részéről (zónájáról). Egy szolgáltató több zónáról is szolgáltat adatokat. Bár a zóna sokszor egybeesik egy tartománnyal, nem egy tartomány! A zóna lehet nagyobb, mint egy tartomány (ez azt jelenti a névszolgáltató nemcsak egy tartományért felel), lehet kisebb is (Pl. egy tartomány altartományaiért való felelősséget delegálták, akkor a delegált altartományért más névszolgáltató felel, de az altartomány megmaradt a felettes tartomány részének). Úgy is mondhatjuk, hogy a zónába beletartozik a delegáltakon kívül minden (gazdagépek is, altartományok is).

Válasszunk ki egy tartományon belül egy csomópontot, ami a *névszolgáltató* gép (name server) lesz! Határozzuk meg ennek a zónáját!

Példa: legyen az uni-miskolc tartomány névszolgáltatója a gold és legyen a tartományban az 4.2. ábra szerint néhány altartomány, néhány csomópont. Legyen a zónája a uni-miskolc tartomány minden gépe és altartománya, kivéve az iit altartományt (azt delegáltuk az Informatikai Intézethez).

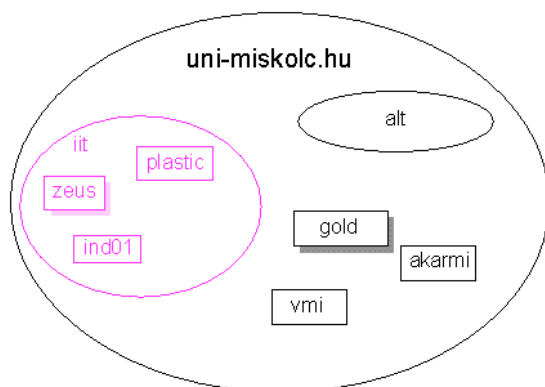
Tételezzük fel, hogy akarmi-ról keresem vmi-t (egy kliens program hív névfeloldó - resolver - rutint), és az nincs az akarmi saját névfeloldó táblájában.

Ekkor *akarmi* megszólítja a névszolgáltatót (name-server), itt a *gold*-ot, és kéri a név feloldását.

gold köteles ismerni a zóna összes nevét, és azt is, hogy az egyes nevekhez tartomány tartozik-e, vagy csomópont. Utóbbi esetben tudja az IP címét is: ezt tehát feloldhatja.

- A névszolgáltató köteles ismerni a delegált tartomány névszolgáltatóját is., hogy feloldáshoz segítséget kérjen.

- A névszolgáltató köteles ismerni a csúcstartomány névszolgáltatóját is, onnan is kérhet segítséget a feloldáshoz. A csúcstól lefelé haladva a hierarchián, előbb-utóbb feloldható a név.



4.2. ábra. Az uni-miskolc tartomány (csak képzeletben!)

A névszolgáltatók ideiglenesen tárolnak (cache-elnek) információkat (akár másik névszolgáltatóról, akár csomóponttól); ez gyorsítja a névfeloldást, csökkenti a hálózati forgalmat. Az ideiglenes tárolás "ideje" beállítható (konzisztencia így biztosítható változások esetén is).

Összefoglalva:

Vannak csúcs-tartományok. (pl. *.hu*, *.de*, *.com*, *.edu* stb.), ezeknek névszolgáltatóik.

A tartományok névszolgáltatói ismerik "felé" a csúcs névszolgáltatót (root name server), "lefelé" az zónájuk altartomány neveit és a delegált tartományok névszolgáltatóit, valamint a saját zónájuk csomópontjainak név-IP cím páryait.

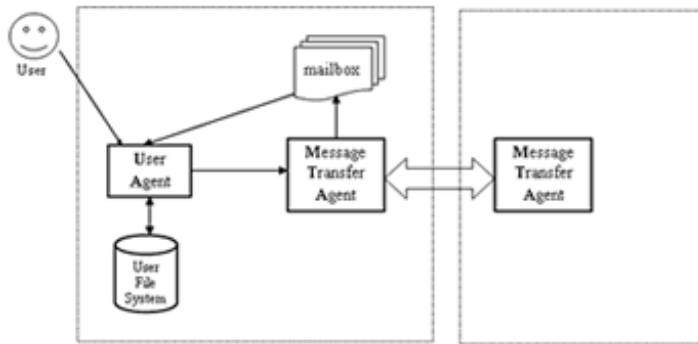
Mindig csak lefelé kell *névegyeztetést* biztosítani: ez még kézben tartható organizáció!

A tartományon belül a névkiosztás elég szabad, csak a névszolgáltató menedzserével kell egyeztetni. Ő ugyanekkor az IP címek kiosztását is adminisztrálja (neki kiadhatnak egy cím-készletet, ebből adhat címeket).

4.4. Az elektronikus levelezés alapfogalmai

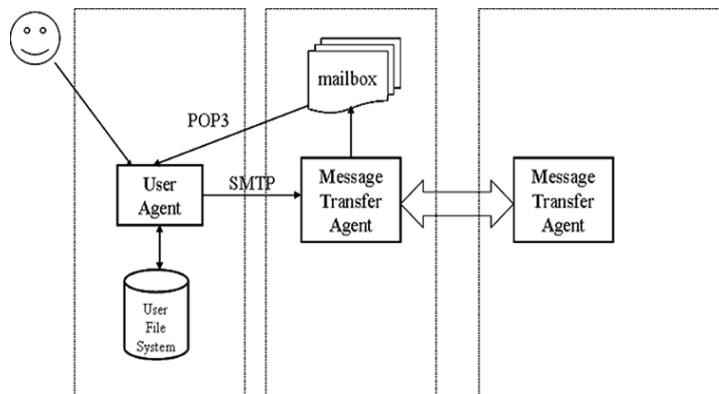
A hagyományos postai küldeményekhez továbbításához hasonlíthatjuk. Ott vannak postaládák (piros színű szekrények), amelyekbe bedobhatjuk a megcímezett, felbélyegezett leveleinket. Van postaszolgálat, ami kiveszi, osztályozza, továbbítja a leveleket, és végül kikézbésíti: bedobja a címzett postafiókjába (leveles szekrényébe). Innen a címzett kiemelheti és olvashatja az üzenetet.

Az elektronikus levelezés egyszerűsített sémája az 4.3. ábrán látható. A *személyek* a helyi gazdagépen (host) *levelező ügynök* (User Agent) programot futtathatnak, és van a gazdagépen egy *postafiókjuk* is. A postafiók az *elektronikus levelezési címmel* (e-mail address) azonosított. A hálózat csomópontjain futnak *levél továbbító programok* (MTA: Message Transfer Agent) is.



4.3. ábra. Elektronikus levelezés sémája

- levél feladását.
- (Némely ügynök a helyi leveleket képes letenni a postafiókba!)



4.4. ábra. A levelező kliens PC-n fut

- átfutó levelek továbbítását (relay funkció),
- bejövő levél elhelyezését a megfelelő postafiókba. (Némely ügynök erre külön processzt hív, nem maga végzi a lehelyezést).

Egy közbelső csomóponton futó MTA ideiglenesen sorokban (queue) tárolja a leveleket, amíg nem tudja azokat továbbítani, vagy postafiókba letenni. Sokszor gondot jelent a nagyméretű levelek ideiglenes tárolása: diszkapacitást köt le, vagy gond lehet egy-egy olyan csomópont kiesése, ahol nagy számban vannak postafiókok, és nem lehet letenni ezekbe a leveleket. Gondot jelent egy rossz címzés is: sokszor ez csak a cél közelében derül ki, illetve, ha kiderült, értesíteni kell a feladót a kézbesítés lehetetlenségéről, indulnak tehát visszafelé is a levelek (amiket az MTA-k generáltak). Ügyeljünk az etikus viselkedésre!

Egy egyszerű elektronikus levél két részből áll: a *fejrészből* és a levél *testéből* (tartalmából).

A *fejrész* sorai megmondják *kitől*, *kinek*, milyen *tárgykorú* levelet kell továbbítani, milyen ügynökök mikor, hová továbbították (*bélyegzők*) stb. Szerepelhet a fejben *kinek még* jellegű cím is. Egyes levelező ügynökök *csatolt* részeket is képesek a levélhez fűzni, ekkor ez is sze-

UA - a helyi gazdagépen fut. Itt van a "postafiókunk" is. Ez biztosítja:

- a postafiók vizsgálatát,
- a levél megtekintését (view funkció),
- a levél törlését, áthelyezését *irat-tartóba* (jegyzékbe), kinyomtatását, lementését egy fájlba stb.
- Biztosítja továbbá feladandó levél szerkesztését (edit funkció),
- levél megcímzését ("fej" készítését),

A User Agent csak akkor fut, ha a felhasználó elindítja! A felhasználói ügynök futhat személyi számítógépen is, ekkor a modell a 4.4. ábrán látható

MTA - a helyi gazdagépen és további csomópontokon is fut. Űn. "daemon", tehát aktív, a rendszermenedzser indítja., menedzseli.

Biztosítja:

- a feladott levél továbbítását (sending funkció),

reper a fejen. Az egyszerű levelezési szabványban a levél *teste* kizárólag ASCII karakterekből álló szövegsorokat tartalmaz.

Az induló UA állítja elő az induló *fejrészt*, többnyire képes a *testet* is előállítani: felolvassa azt egy fájlból, meghív egy szövegszerkesztőt a tartalom szerkesztésére. A nálunk működő *pine* UA pl. alapértelmezés szerint a *pico* nevű szövegszerkesztőt hívja meg levéltartalom szerkesztésre, de lehet neki más szövegszerkesztőt is adni. A UA képes feladni (send) a megcímezett levelet: ekkor tulajdonképpen átadja egy MTA-nak a levelet.

Az MTA-k a továbbiakban a fejrészt vizsgálják, az kiegészítik és továbbpasszolják a levelet, az utolsó MTA (vagy egy általa hívott processz) leteszi a címzett postafiókjába. (Ha úgy tetszik, az MTA a fejrészből borítékot, borítékokat készít. A borítékban már csak a címzett információi vannak, ha több címzett is van, több boríték is "készül". Az MTA beállításától függően felveszi a kapcsolatot a címzett MTA-val, egyeztet, létezik-e a címzett egyáltalán (igenlő válasza továbbítja a levelet), vagy csak továbbítja a borítékolt levelet (benne a fejet és testet), és más MTA-ra bízta a címzett létezésének ellenőrzését. Vegyük észre a boríték és a fejrész közti különbséget! Leegyszerűsítve: a fejrészben lehet több címzett, mindegyiknek külön boríték készül!

E-mail címek az Internet-en

(1) Adott egy *gazdagép* (host, cluster) a saját *felhasználói azonosítóival*. Pl.:

gold.uni-miskolc.hu gazdagépen

iitvd bejegyzett felhasználó név (számlaszám).

ekkor:

`iitvd@gold.uni-miskolc.hu`

egy *e-mail* cím.

(2) Adott egy *altartomány* (subdomain), a *névszolgáltatója*, (esetleg cluster), és adott egy *felhasználói név* a az *altartományban* (clusterben):

iit.uni-miskolc.hu egy subdomain,

zeus a névszolgáltatója,

vadasz egy felhasználói név a tartományban,

ekkor

`vadasz@iit.uni-miskolc.hu`

egy *e-mail* cím. (Ekkor a névszolgáltató levelezési irányító információt is biztosít, ami megmondja, hogy melyik gazdagépen is van a címzett postafiókjja.)

Vagyis

user-name@host.sub-domain...

illetve

user-name@sub-domain...

alakú lehet egy e-mail cím!

A protokollok

SMTP (Simple Mail Transfer Protocol) RFC-821, 1982, aug.

MIME (Multipurpose Internet Mail Extension) RFC-1521, 1993, szept.

4.5. Az Internet gopher

Minnesotai Egyetem, 1991, nyilvános adatbázis és lekérdező rendszer.

Kliens-szerver filozófiájú.

Gopher szerver: többfeladatos (multi tasking, rendszerint multi user) operációs rendszer alatt futó információ gyűjtő, szolgáltató. Daemon.

Képes:

- hierarchikus struktúrában információkat (lapokat) tárolni,
- kliensektől kiinduló kapcsolatkérsre kapcsolatokat létesíteni,
- kliens kérésére "lépni" föl/le a hierarchikus struktúrán,
- kliens kérésére lapot (fájlt) letölteni a kliens számára,
- kliens kérésére "szolgáltatást" biztosítani.

Gopher kliens: szinte minden operációs rendszer alatt futhat. Felhasználó indíthatja.

Képes:

- kapcsolatot létesíteni adott szerverrel,
- egy-egy "lapot" fogadni és azt "kezelní".

Mi lehet egy "lap"?

- egy "menü" ami nem más, mint az adott szinten egy jegyzéklista. Föl/le léphetünk rajta. Kiválaszthatjuk egy elemét. Egy eleme: további "lap".
- egy *szolgáltatás*: Pl. egy keresés, egy átkapcsolás másik gopher szerverre, egy processz elindítása stb.
- egy *fájl*, ami véglegvél a hierarchián. Lehet szöveg: ekkor a kliens megjelenítheti (viewer), lehet egy kép, ekkor is megjelenítheti, egy hangfájl, ekkor "lejátszhatja" egy lejátszóval (ha a kliens képes rá), egy video stb. A fájlokat le is töltheti a helyi fájl-rendszerbe, esetleg postázhatja stb.

Ismerjük meg a *gopher világot*! Ismerd meg a tanszék laboratóriumaiban a gopher lehetőségeket!

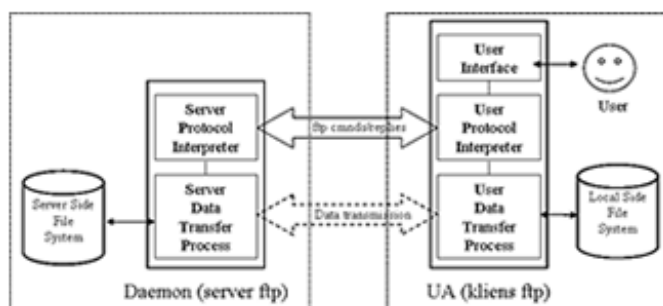
Jegyezzük meg: a kapcsolat a kliens és a szerver között csak addig él, amíg a "lap lejön". A letöltött lapot a kliens tárolja, megjeleníti. Nem terheljük a hálózatot, ha egy-egy lapot sokáig nézegetünk, csakis a kliens gépe erőforrásait használjuk ez alatt. Ez nagy különbség a telenet-es kapcsolatépítéshez képest, hiszen ott a kapcsolat addig él, amíg az ülésünk él. Persze, megtehetjük, hogy telnet-es kapcsolatot építünk arra a csomópontra, amelyiken a gopher szerver fut, és e kapcsolaton futtatunk egy gopher klienst (vannak csomópontok, ahol a *gopher* login névre jelszó nélküli bejelentkezés a gopher klienst indítja). Ekkor a kliens egyszerű processzek közti kommunikációs mechanizmusokkal építi fel a gopher kapcsolatot, a telnetes kapcsolat végig terheli a hálózatot.

4.6. A File Transfer Protocol FTP

Egy ftp kliens indítása:

```
> ftp
```

Kliens-szerver alapú fájltoábbítási (letöltés, felvitel) célú szolgáltatás. A helyi gépen indított ftp kliens kapcsolat létesít a távoli csomóponton az *ftp daemon*-nal (lásd 4.5. ábra).



4.5. ábra. Az ftp kliens-szerver modellje

Kell hozzá számlaszám (login név: uid) a távoli gépen.

Szokásos az *: anonymous* login név az ún. anonymous ftp szervereken. Ezekhez rendszerint nincs jelszó, vagy ha van, az a kezdeményező azonosítását szolgálja.

Az ftp kliens

- "lépegethet" a távoli csomópont fájlrendszerén, lekérdezheti a jegyzékek listáját stb.,
- fájl transzfer valósít meg ide (get) vagy oda (put). Anonimus szervereken rendszerint nem engedélyezik az oda továbbítást .

Szinte minden operációs rendszerben, ahol TCP/IP protokoll működik, van ftp kliens. A Unixok alatt rendszerint van ftp szerver is. Anonimus szerver persze nem mindenütt van.

Nézz utána az on-line kézikönyvben, milyen parancsai vannak az ftp kliensnek.

4.7. Az Archie szolgáltatás

Segít, hogy találjunk *anonymous ftp szervereket* az Internet-en.

Nagyon sok ftp szolgáltató név van a különböző archie szervereken. Tudni kell az archie szerverek (hostok) neveit. (Pl. archie.th-darmstadt.de).

Azarchie-t 3 módon érhetjük el:

helyi klienssel: >archie

telnet-üléssel >telnet host + login:archie

e-mail-lel: mail toarchie@archie-server, és a levél testében a parancsok (pl. help).

4.8. Hogyan kereshetünk személyeket, számítógépeket?

- 1) a finger segítségével (lásd: man finger).
- 2) a WHOIS szolgáltatással.
- 3) az X.500 ISO szabványú directory szolgáltató lekérdezésével.

4.9. A World Wide Web (WWW) és nézegetők

A WWW (W³) *hipermédia* jellegű, az Internet-re kiterjedő szolgáltatás.

A WWW az Internet világban forradalmi változást hozott. Hatására az Internet akadémiai, kutatói hálózatból üzleti és hobby hálózattá vált, szerepet kapott a szórakoztatás világában, a tájékoztató médiák körében, a pénzforgalom és kereskedelem, a reklám világában, az üzleti alkalmazások motorjává vált. Hatása akkora, hogy sokan, mikor az Internet kifejezést meghallják, csakis a WWW világra gondolnak.

A WWW koncepciójában a már jól ismert kliens-szerver koncepció mellett három - tulajdonképpen eddig szintén ismert - paradigma fonódik össze. Ezek a hypertext paradigmája, a hypertext utalások kiterjesztése IP hálózatokra gondolat és a multimédia paradigmája.

A *hypertext paradigma* lényege olyan szövegmegjelenítés, melyben a lineáris vagy a hierarchikus rendszerű, rendezett szöveg olvasás korlátja megszűnik. Elektronikus szövegek lineáris olvasásához elegendő egy egyszerű szövegnézegető (viewer). Már a legegyszerűbb szövegszerkesztő is megfelel, melynek segítségével előre, hátra lapozhatunk a szövegben, sőt, egy esetleges kereső (search) funkcióval már-már átléphetünk egy szinttel feljebb, közelíthetjük a rendezett szövegek olvasásához. A rendezett olvasást biztosítanak a szótárprogramok, adatbázis lekérdezők. A hypertext jellegű rendszerekben a szövegdokumentumokban valamilyen szövegrészekhez rögzítettek kapcsolódó dokumentumaik is. A megjelenítő valamilyen módon kiemelten jeleníti meg ezeket a szövegrészeket. Ezek a kiemelt részek utalások (kapcsolatok, linkek) más dokumentumokra, más szövegekre, szövegrészekre. A hypertext böngésző nem csak kiemelten jeleníti meg a szövegrészeket, hanem lehetőséget ad azok kiválasztására is (pl. mutatóval rákattinthatunk). A kiemelt rész kiválasztásával az utalt, a hivatkozott (linked) dokumentum betöltődik a nézegetőbe, folytatható az olvasás, természetesen itt ugyancsak lehetnek utalások, akár közvetlenül, akár közvetetten már előzőleg nézegetett dokumentumra is. Az így biztosított információs rendszer jellegzetesen hálós szerkezetű. Léteznek hypertext szövegeket létrehozó, azokat kezelni tudó információs rendszerek, bár jelentőségük a WWW terjedésével egyre szűkebb.

A *hypertext IP hálózatra való kiterjesztése* megszünteti azt a korlátozást, hogy az utalások csak ugyanarra a helyszínre, számítógéprendszerre vonatkozhatnak. Egy-egy kapcsolódó

dokumentum helye a hálózaton "akárhol" lehet, ha az utalások megfelelnek az Uniform Resource Locator (URL) szabványnak.

Végül a *multimédia* paradigma megszünteti a szövegekre való korlátozást: nemcsak hypertext háló, hanem hypermédia háló alakulhat ki. Hivatkozott dokumentum lehet kép, hanganyag, mozgóképfájl, adatfájl, szolgáltatás stb. is. Ráadásul a kép dokumentumokban könnyű elhelyezni további utalásokat is, onnan tovább folytatható a láncolás.

Végül szóljunk a WWW jellegzetes kliens-szerver koncepciójáról is. A WWW kliensek a böngészőprogramok, a tallózók. Képesek a Hyper Text Markup Language (HTML) direktíváival kiegészített szövegek megjelenítésére, bennük az utalásokhoz rendelt szövegrészek kiemelt kezelésére, a kiemelt szövegek kiválasztására. Képesek bizonyos kép dokumentumok megjelenítésére, ezekben kiemelések kiválasztására, hangfájlok, videók lejátszására, vagy közvetlenül, vagy valamilyen segédprogram aktiválásával. A szerverek pedig képesek szöveg-, kép-, hang- és videó-fájlokat megkeresni saját fájlrendszerükben, és azokat elküldeni a kliensnek megjelenítésre. A kliens és szerver között üzenetváltások jellegzetesen négy lépéses forgatókönyv szerint történnek a Hyper Text Transport Protocol (HTTP) szabályozása alatt.

Az első lépés a kapcsolat-létesítés (connection): ezt a kliens kezdeményezi, hozzá legfontosabb információ a szerver azonosítója. A második lépésben a kliens kérelmet (request) küld a kapcsolaton a szervernek, ebben közli, hogy milyen protokollal, melyik dokumentumot kéri (nem részletezzük, de az átviteli eljárás, a *method* is paramétere a kérelemnek). Ezután a szerver megkeresi a kért dokumentumot és válaszol (response): a kapcsolaton leküldi a kért dokumentumot. Végül a kapcsolat lezárul (close). Mindezek után a kliens felelőssége, hogy mit is csinál a leküldött dokumentummal. Mindenesetre ideiglenesen tárolja a saját memóriájában és/vagy fájl-rendszerén, és a dokumentum fajtájától függően megjeleníti azt, esetleg elindítva külső lejátszót, annak átadva dokumentumot közvetve jeleníti meg, lehetőséget ad a felhasználónak végleges lementésre stb. Már a programozás kérdéskörébe tartozik, hogy ha olyan dokumentumot kap a böngésző, melyet közvetlenül nem tud megjeleníteni, lejátszani (futtatni), milyen segédprogramot hívjon meg a megjelenítésre. A felhasználó a MIME szabványoknak megfelelő lejátszókat beállíthat, rendszerint a böngésző konfigurációs menüjében a segítőprogramok (helpers) almenüben.

A manapság legismertebb WWW böngészők nem csak a HTTP protokollt ismerik, hanem más protokollok segítségével nemcsak WWW szolgáltatókkal tudnak kapcsolatot létesíteni, azoktól szolgáltatásokat kérni. Hogy csak a legfontosabbakat említsük, rendszerint képesek ftp protokollon keresztül állomány átvitel szolgáltatások igénylésére (ekkor a kapcsolat végigéli az ftp ülést), telnet protokollal távoli elérésre (ugyancsak végig van kapcsolat az ülés alatt), gopher protokollal gopher szolgáltatás és böngészés végzésére, POP3 protokollal levelezéskrények vizsgálatára, letöltésére, SMTP vagy MIME protokollal levelek feladására (kapcsolat levéltovábbító ügynök szolgáltatóhoz), a USENET news levelek olvasására. Mindezekhez viszonylag egységes felhasználói felületet biztosítanak, innen adódik tehát az a téveszme, hogy az Internet az a WWW, vagy fordítva: hiszen egy jó WWW tallózó szinte minden szolgáltatást biztosít, amit az Interneten elérhetünk.

Amit eddig elmondtunk a WWW világról, az még mindig nem biztosítja igazán a programozhatóságot. A WWW szolgáltatóknak rendszerint van még további szolgáltatásuk is. A legegyszerűbb "programozási" lehetőség az, hogy bizonyos szolgáltatók megengedik, hogy különben kommentárnak számító HTML direktíva a szolgáltató parancs-értelmezőjének szó-

ló burok parancs legyen. A szerver elindítja a parancsértelmezőt, végrehajtatja a parancsot, az eredményeit pedig szövegfájl válaszként elküldi a kliensnek megjelenítésre. A Common Gateway Interface (CGI) protokoll szerint akár paramétereket is küldhetünk a kliensből a CGI programnak, a CGI program akár bele is írhat az utoljára megjelenített dokumentumba. Maga a CGI program pedig akármilyen nyelvű is lehet, gyakran egyszerű burokprogramok (shell script), többnyire lefordított és szerkesztett futtatható fájlok. (Ne feledjük: a CGI nem egy programnyelv, hanem egy interfész, azt szabályozza, hogy kap és ad információkat, paramétereket a CGI program.) Leggyakoribb alkalmazási területük a számlálók és vendégkönyvek elhelyezése a WWW nyitólapokon, pontos idő szolgáltatás, keresések a helyi vagy akár távoli WWW rendszerekben, átjárók adatbázis lekérdező rendszerekhez, kérdőívek, szavazólapok kitöltése, de egyéb programozási megoldásokra is alkalmasak.

A WWW programozási nyelve: a Java

Mint említettük, a WWW böngészőkkel egységes, felhasználóbarát felületet kapott a WWW, ezzel részben az Internet is. A programozás eszközeit - korlátozottan - igénybe lehet venni. A CGI programokkal, melyek a szerver oldalon futnak, bizonyos feladatokat megoldhatunk, bizonyos alkalmazásokat készíthetünk, vagy készíthetnek számunkra. A Sun Microsystem fejlesztői felismerve az eddigi programnyelvek korlátozásait egy teljesen új programnyelvet dolgoztak ki a WWW programozáshoz, a Java nyelvet. Ezzel párhuzamosan a WWW tallózók fejlesztői olyan böngészőt készítettek, amelyik a Java nyelven írt programokat képes értelmezni és futtatni. Az ilyen tallózók *Java virtuális gépként* viselkednek. A HTML dokumentumokban a Java programokra való hivatkozások ugyanúgy találhatóak meg, mint a más, pl. kép hivatkozások, és a dokumentum letöltése során akár ezek is letöltődnek. Az a tény, hogy a program nem a szerver oldalon fut (mint a CGI programoknál történik), hanem letöltődik a böngészőhöz és a böngésző hajtja azt végre több előnyt is eredményezett. Egyik előny az, hogy tehermentesítik a szervert, esetlegesen a hálózatot. Másik, talán még nagyobb előny, hogy a nem kell a különböző operációs rendszerekhez, géptípusokhoz illeszteni az alkalmazást, a "szabványos" Java kódot a Java virtuális gép, a böngésző végre tudja hajtani, a böngésző feladata az adott hardver, operációs rendszer adottságaihoz való illesztés. Hátrány is jelentkezik azonban, elsősorban biztonsági kérdések merülnek fel a Java alkalmazások (applet) futtatásánál. Miután a helyi gépen futtatunk, akár bizonytalan eredetű programokat, külön gondot kellett fordítani arra, hogy ne legyen lehetséges vírus- vagy féregprogramokat készíteni a Java nyelv segítségével. Ennek következtében a Java programcskák nem képesek a számukra kijelölt területen túllépni, maguk a böngészők pedig külön kérésünkre további biztonsági szintként nem fogadnak Java alkalmazásokat (amivel el is veszítjük a programozhatóságot). A Java nyelv könnyen megtanulható, különösen C++ ismeretek birtokában.

Tessék használni, felfedezni a WWW világot!

IRODALOM

Barry M. Leiner, Vinton G. Cerf, David D. Clark, Robert E. Kahn, Leonard Kleinrock, Daniel C. Lynch, Jon Postel, Larry G. Roberts, Stephen Wolff : [A Brief History of the Internet](http://www.isoc.org/internet/history/brief.shtml),
<http://www.isoc.org/internet/history/brief.shtml>