

7

NIVELUL APLICAȚIE

După ce am epuizat toate preliminariile putem aborda nivelul unde pot fi găsite toate aplicațiile. Nivelurile de sub nivelul aplicație servesc la asigurarea unui transport sigur, dar nu îndeplinesc nici o funcție concretă pentru utilizatori. În acest capitol vom studia câteva aplicații reale.

Totuși, chiar și la nivelul aplicație, apare necesitatea unor protocoale-suport care să permită funcționarea aplicațiilor reale. Înainte de a începe studiul aplicațiilor, ne vom ocupa de unul dintre acestea. Subiectul în discuție este DNS, care se ocupă de convențiile de nume în Internet. După aceea vom examina trei aplicații reale: poșta electronică, World Wide Web (rom.: rețea de întindere planetară) și, în final, multimedia.

7.1 DNS - SISTEMUL NUMELOR DE DOMENII

Cu toate că teoretic programele ar putea să se refere la sistemele gazdă, la cutiile poștale și la alte resurse prin adresa lor de rețea (de exemplu prin adresa IP), aceste adrese sunt greu de memorat de către oameni. De asemenea, în trimiterea de poștă electronică la *tana@128.111.24.41* ar însemna că dacă furnizorul de servicii Internet sau organizația Tanei mută serverul de poștă pe o mașină diferită, cu o adresă IP diferită, adresa ei de e-mail se va schimba. De aceea au fost introduse nume ASCII pentru a separa numele mașinilor de adresele mașinilor. În acest fel, adresa Tanei ar putea fi ceva de genul *tana@art.ucsb.edu*. Cu toate acestea, rețeaua înțelege numai adrese numerice, deci este necesar un mecanism care să convertească șirurile ASCII în adrese de rețea. În secțiunile următoare se va studia cum este realizată această conversie în Internet.

Încă de la ARPANET exista un fișier *host.txt* care cuprindea toate sistemele gazdă și adresele lor IP. În fiecare noapte, toate gazdele îl preluau de la situl unde era păstrat. Pentru o rețea formată din câteva sute de mașini mari, cu divizarea timpului, această abordare era destul de rezonabilă.

Totuși, atunci când la rețea au fost conectate mii de stații de lucru, toți și-au dat seama că această abordare nu putea să funcționeze la nesfârșit. În primul rând dimensiunea fișierului ar deveni prea mare. Cu toate acestea și chiar mai important, conflictele de nume de sisteme gazdă ar apărea în permanență dacă nu ar fi administrate centralizat, ceva de negândit într-o rețea internațională de dimensiuni uriașe din cauza încărcării și a latenței. Pentru a rezolva aceste probleme, a fost inventat **DNS (Domain Name System - Sistemul numelor de domenii)**.

Esența DNS-ului constă într-o schemă ierarhică de nume de domenii și a unui sistem de baze de date distribuite pentru implementarea acestei scheme de nume. În principal este utilizat pentru a pune în corespondență numele sistemelor gazdă și adresele destinațiilor de e-mail cu adresele IP, dar poate fi utilizat și pentru alte scopuri. DNS este definit în RFC-urile 1034 și 1035.

Foarte pe scurt, DNS este utilizat după cum urmează. Pentru a stabili corespondența dintre un nume și o adresă IP, programul de aplicație apelează o procedură de bibliotecă numită **resolver**, transferându-i numele ca parametru. Putem vedea un exemplu de resolver, *gethostbyname*, în fig. 6-6. Resolver-ul trimite un pachet UDP la serverul DNS local, care caută numele și returnează adresa IP către resolver, care o returnează apelantului. Înarmat cu adresa IP, programul poate stabili o conexiune TCP cu destinația sau îi poate trimite pachete UDP.

7.1.1 Spațiul de nume DNS

Administrarea unui volum mare de nume în permanentă schimbare nu este o problemă prea ușoară. În sistemul poștal, administrarea numelor este realizată impunând ca pe o scrisoare să se specifice (implicit sau explicit) țara, statul sau provincia, orașul, strada și restul adresei destinatarului. Utilizând o astfel de adresare ierarhică, nu există nici o confuzie între Marvin Anderson de pe Main St. din White Plains, N.Y. și Marvin Anderson de pe Main St. din Austin, Texas. DNS lucrează în același mod.

Conceptual, Internetul este divizat în peste 200 **domenii** de nivel superior, fiecare domeniu cuprinzând mai multe sisteme gazdă. Fiecare domeniu este partiționat în subdomenii și acestea sunt, la rândul lor, partiționate ș.a.m.d. Toate aceste domenii pot fi reprezentate ca un arbore, așa cum se arată în fig. 7-1. Frunzele arborelui reprezintă domenii care nu au subdomenii (dar, bineînțeles, conțin sisteme). Un domeniu frunză poate conține un singur sistem gazdă sau poate reprezenta o firmă, deci să conțină mii de sisteme gazdă.

Domeniile de pe primul nivel se împart în două categorii: generice și de țări. Domeniile generice sunt *com* (comercial), *edu* (instituții educaționale), *gov* (guvernul federal al SUA), *int* (organizații internaționale), *mil* (forțele armate ale SUA), *net* (furnizori Internet) și *org* (organizații nonprofit). Domeniile de țări includ o intrare pentru fiecare țară, cum se definește în ISO 3166.

În noiembrie 2000, ICANN a aprobat patru domenii de nivel superior noi, de interes general, și anume, *biz* (afaceri), *info* (informații), *name* (nume de persoane), și *pro* (profesii, ca de exemplu doctori sau avocați). În plus, au fost introduse trei domenii de nivel superior cu caracter specializat, cerute de către anumite industrii. Acestea sunt *aero* (industria aerospațială), *coop* (cooperative), și *museum* (muzee). În viitor vor fi adăugate alte domenii superioare.

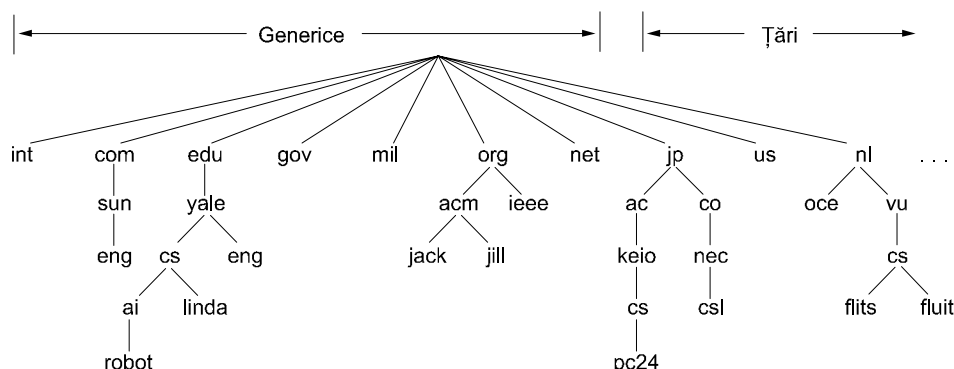


Fig. 7-1. O porțiune a spațiului numelor de domenii din Internet.

Pe de altă parte, pe măsură ce Internetul devine mai comercial, el devine și mai discutabil. Să luăm domeniul *pro*, de exemplu, care a fost proiectat pentru profesioniștii atestați. Dar cine este un profesionist? Și de cine este atestat? Dar cum rămâne cu fotografi, profesorii de pian, magicienii, instalatorii, frizerii, exterminatorii, artiștii de tatuaje, mercenarii și prostituatele? Sunt acestea meserii și sunt acceptabile pentru domeniile *pro*? Și dacă da, cine atestă diverși practicieni?

În general, obținerea unui domeniu de nivel secundar, ca de exemplu *nume-al-companiei.com*, este ușoară. Pur și simplu este necesară doar consultarea serviciului de înregistrare al nivelului superior corespunzător (*com* în acest caz) pentru a vedea dacă numele dorit este disponibil și nu aparține altcuiva. Dacă nu sunt probleme, solicitantul plătește o mică taxă anuală și primește numele. Până acum, cam toate cuvintele comune (din engleză) au fost luate în domeniul *com*. Încercați nume de articole casnice, animale, plante, părți ale corpului etc. Aproape toate sunt luate.

Fiecare domeniu este identificat prin calea în arbore de la el la domeniul (fără nume) rădăcină. Componentele sunt separate prin puncte (pronunțat „dot”). Astfel, departamentul tehnic de la Sun Microsystems ar putea fi *eng.sun.com*, în loc de numele în stil UNIX */com/sun/eng*. De notat că această numire ierarhică face ca *eng.sun.com* să nu intre în conflict cu posibila utilizare a lui *eng* din *eng.yale.edu*, care ar putea fi folosit pentru departamentul de limba engleză de la Yale.

Numele de domenii pot fi absolute sau relative. Un nume absolut de domeniu se termină cu un punct (de exemplu, *eng.sun.com.*), în timp ce unul relativ nu. Numele relative trebuie interpretate în context pentru a le determina înțelesul adevărat. În ambele cazuri, un nume de domeniu se referă la un anumit nod din arbore și la toate nodurile de sub el.

Numele de domenii nu fac distincție între litere mici și litere mari, astfel *edu*, *Edu*, sau *EDU* înseamnă același lucru. Componentele numelor pot avea o lungime de cel mult 63 caractere, iar întreaga cale de nume nu trebuie să depășească 255 de caractere.

În principiu, domeniile pot fi inserate în arbore în două moduri diferite. De exemplu, *cs.yale.edu* ar putea la fel de bine să fie inclus în domeniul țării *us* ca *cs.yale.ct.us*. În practică, totuși, aproape toate organizațiile din Statele Unite sunt repartizate după criteriul generic, iar aproape toate din afara Statelor Unite fac parte din domeniul țării lor. Nu există nici o regulă împotriva înregistrării sub două domenii de nivel superior, însă puține organizații în afară de cele multinaționale o fac (de exemplu, *sony.com* și *sony.nl*).

Fiecare domeniu controlează cum sunt alocate domeniile de sub el. De exemplu, Japonia are domeniile *ac.jp* și *co.jp* echivalente cu *edu* și *com*. Olanda nu face nici o distincție și pune toate orga-

nizațiile direct sub *nl*. Astfel următoarele trei nume sunt toate departamente de calculatoare (computer science) din universități:

1. *cs.yale.edu* (Universitatea Yale din Statele Unite).
2. *cs.vn.nl* (Vrije Universiteit în Olanda).
3. *cs.keio.ac.jp* (Universitatea Keio din Japonia).

Pentru a crea un nou domeniu, se cere permisiunea domeniului în care va fi inclus. De exemplu, dacă un grup VLSI de la Yale dorește să fie cunoscut ca *vlsi.cs.yale.edu*, acesta are nevoie de permisiunea celui care administrează *cs.yale.edu*. Similar, dacă este acreditată o nouă universitate, să zicem Universitatea din Northern South Dakota, ea trebuie să ceară administratorului domeniului *edu* să-i atribuiască *unsd.edu*. În acest mod sunt evitate conflictele de nume și fiecare domeniu poate ține evidența tuturor subdomeniilor sale. Odată ce un nou domeniu a fost creat și înregistrat, el poate crea subdomenii, cum ar fi *cs.unsd.edu*, fără a cere permisiunea de la cineva din partea superioară a arborelui.

Atribuirea de nume respectă granițele organizaționale, nu pe cele ale rețelelor fizice. De exemplu, dacă departamentele de știința calculatoarelor și de inginerie electrică sunt localizate în aceeași clădire și folosesc aceeași rețea locală, ele pot avea totuși domenii distincte. Similar, dacă departamentul de știința calculatoarelor este împărțit în două clădiri (Babbage Hall și Turing Hall), toate sistemele gazdă din ambele clădiri aparțin aceluiași domeniu.

7.1.2 Înregistrări de resurse

Fiecărui domeniu, fie că este un singur calculator gazdă, fie un domeniu de nivel superior, îi poate fi asociată o mulțime de înregistrări de resurse (**resource records**). Pentru un singur sistem gazdă, cea mai obișnuită înregistrare de resursă este chiar adresa IP, dar există multe alte tipuri de înregistrări de resurse. Atunci când un resolver trimite un nume de domeniu către un DNS, ceea ce va primi ca răspuns sunt înregistrările de resurse asociate acelui nume. Astfel, adevărata funcție a DNS este să realizeze corespondența dintre numele de domenii și înregistrările de resurse.

O înregistrare de resursă este un 5-tuplu. Cu toate că, din rațiuni de eficiență, înregistrările de resurse sunt codificate binar, în majoritatea expunerilor ele sunt prezentate ca text ASCII, câte o înregistrare de resursă pe linie. Formatul pe care îl vom utiliza este următorul:

Nume_domeniu Timp_de_viață Clasă Tip Valoare

Nume_domeniu (domain_name) precizează domeniul căruia i se aplică această înregistrare. În mod normal există mai multe înregistrări pentru fiecare domeniu și fiecare copie a bazei de date păstrează informații despre mai multe domenii. Acest câmp este utilizat ca cheie de căutare primară pentru a satisface cererile. Ordinea înregistrărilor în baza de date nu este semnificativă.

Câmpul *Timp_de_viață* (time_to_live) dă o indicație despre cât de stabilă este înregistrarea. Informația care este foarte stabilă are asigurată o valoare mare, cum ar fi 86400 (numărul de secunde dintr-o zi). Informației instabile îi este atribuită o valoare mică, cum ar fi 60 (1 minut). Vom reveni la acest punct mai târziu, când vom discuta despre utilizarea memoriei ascunse.

Al treilea câmp dintr-o înregistrare de resursă este *Clasă* (class). Pentru informațiile legate de Internet este tot timpul *IN*. Pentru alte informații pot fi folosite alte coduri, însă în practică acestea se întâlnesc rar.

Câmpul *Tip* (type) precizează tipul înregistrării. Cele mai importante tipuri sunt prezentate în fig. 7-2.

Tip	Semnificație	Valoare
SOA	Start autoritate	Parametrii pentru această zonă
A	Adresa IP a unui sistem gazdă	Întreg pe 32 de biți
MX	Schimb de poștă	Prioritate, domeniu dispus să accepte poștă electronică
NS	Server de Nume	Numele serverului pentru acest domeniu
CNAME	Nume canonic	Numele domeniului
PTR	Pointer	Pseudonim pentru adresa IP
HINFO	Descriere sistem gazdă	Unitate centrală și sistem de operare în ASCII
TXT	Text	Text ASCII neinterpretat

Fig. 7-2. Principalele tipuri de înregistrări de resurse DNS.

O înregistrare *SOA* furnizează numele sursei primare de informații despre zona serverului de nume (descrisă mai jos), adresa de e-mail a administratorului, un identificator unic și diverși indicatori și contoare de timp.

Cel mai important tip de înregistrare este înregistrarea *A* (adresă). Ea păstrează adresa IP de 32 de biți a unui sistem gazdă. Fiecare sistem gazdă Internet trebuie să aibă cel puțin o adresă IP, astfel încât alte mașini să poată comunica cu el. Unele sisteme gazdă au două sau mai multe conexiuni în rețea, caz în care vor avea câte o înregistrare de tip *A* pentru fiecare conexiune (și astfel pentru fiecare adresă IP).

Următoarea ca importanță este înregistrarea *MX*. Aceasta precizează numele sistemului gazdă pregătit să accepte poșta electronică pentru domeniul specificat. El este folosit deoarece nu toate mașinile sunt pregătite să accepte poșta electronică pentru domeniul specificat. Dacă cineva vrea să-i trimită un e-mail, de exemplu, lui *bill@microsoft.com*, sistemul care trimite trebuie să găsească un server la *microsoft.com* dispus să accepte e-mail. Înregistrarea *MX* poate să furnizeze această informație.

Înregistrările *NS* specifică serverele de nume. De exemplu, fiecare bază de date DNS are în mod normal o înregistrare *NS* pentru fiecare domeniu de pe primul nivel, astfel încât, de exemplu, poșta electronică să poată fi trimisă în zone îndepărtate ale arborelui de nume. Vom reveni la acest aspect mai târziu.

Înregistrările *CNAME* permit crearea pseudonimelor. De exemplu, o persoană familiarizată cu atribuirea numelor în Internet, care dorește să trimită un mesaj unei persoane al cărei nume de conectare la un sistem de calcul din departamentul de calculatoare de la M.I.T. este *paul* poate presupune că adresa *paul@cs.mit.edu* este corectă. De fapt această adresă nu este corectă, deoarece domeniul departamentului de calculatoare de la M.I.T. este *lcs.mit.edu*. Totuși, ca un serviciu pentru cei care nu știu acest lucru, M.I.T. poate crea o intrare *CNAME*, pentru a dirija persoanele și programele în direcția corectă. O astfel de intrare poate fi:

```
cs.mit.edu 86400 IN CNAME lcs.mit.edu
```

Ca și *CNAME*, *PTR* se referă la un alt nume. Totuși, spre deosebire de *CNAME*, care este în realitate numai o macro-definiție, *PTR* este un tip de date DNS a cărui interpretare depinde de contextul în care este utilizat. În practică este aproape întotdeauna utilizat pentru asocierea unui nume cu o adresă IP, pentru a permite căutarea adresei IP și obținerea numelui mașinii corespunzătoare. Acestea se numesc **căutări inverse (reverse lookups)**.

Înregistrările *HINFO* permit aflarea tipului de mașină și de sistem de operare cărora le corespunde domeniul. În sfârșit, înregistrările *TXT* permit domeniilor să se autoidentifice într-un mod arbitrar. Aceste două tipuri de înregistrări sunt introduse pentru ușurința utilizatorului. Nici una

dintre ele nu este necesară, astfel încât programele nu pot conta pe obținerea lor (și probabil că dacă le obțin nu le pot trata).

În final ajungem la câmpul *Valoare*. Acest câmp poate fi un număr, un nume de domeniu sau un șir ASCII. Semantica depinde de tipul de înregistrare. O scurtă descriere a câmpurilor *Valoare* pentru fiecare dintre principalele tipuri de înregistrări este dată în fig. 7-2.

Un exemplu de informație ce se poate găsi în baza de date DNS a unui domeniu este prezentat în fig. 7-3. Această figură prezintă o parte (semi-ipotetică) a bazei de date pentru domeniul *cs.vu.nl* prezentat în fig. 7-1. Baza de date conține șapte tipuri de înregistrări de resurse.

;Baza de date pentru cs.vu.nl

cs.vu.nl.	86400	IN	SOA	star boss (9527, 7200, 7200, 241920, 86400)
cs.vu.nl.	86400	IN	TXT	„Divisie Wiskunde en Informatica.”
cs.vu.nl.	86400	IN	TXT	„Vrije Universiteit Amsterdam.”
cs.vu.nl.	86400	IN	MX	1 zephyr.cs.vu.nl.
cs.vu.nl.	86400	IN	MX	2 top.cs.vu.nl.
flits.cs.vu.nl.	86400	IN	HINFO	Sun Unix
flits.cs.vu.nl.	86400	IN	A	130.37.16.112
flits.cs.vu.nl.	86400	IN	A	192.31.231.165
flits.cs.vu.nl.	86400	IN	MX	1 flits.cs.vu.nl.
flits.cs.vu.nl.	86400	IN	MX	2 zephyr.cs.vu.nl.
flits.cs.vu.nl.	86400	IN	MX	3 top.cs.vu.nl.
www.cs.vu.nl.	86400	IN	CNAME	star.cs.vu.nl.
ftp.cs.vu.nl.	86400	IN	CNAME	zephyr.cs.vu.nl.
rowboat		IN	A	130.37.56.201
		IN	MX	1 rowboat
		IN	MX	2 zephyr
		IN	HINFO	Sun Unix
little-sister		IN	A	130.37.62.23
		IN	HINFO	Mac MacOS
laserjet		IN	A	192.31.231.216
		IN	HINFO	„HP Laserjet IIISi” Proprietary

Fig. 7-3. O parte dintr-o posibilă bază de date DNS pentru *cs.vu.nl*.

Prima linie necomentată din fig. 7-3 dă câteva informații de bază despre domeniu, de care nu ne vom ocupa. Următoarele două linii furnizează informații textuale despre amplasarea domeniului. Urmează două intrări care specifică primul și al doilea loc unde se încearcă să se livreze poșta electronică trimisă pentru *persoana@cs.vu.nl*. Mai întâi se încearcă trimiterea la mașina *zephyr*. Dacă aceasta eșuează, atunci trebuie să se încerce la *top*.

După o linie liberă, adăugată numai pentru claritate, urmează linii care spun că *flits* este o stație de lucru Sun care lucrează sub UNIX și se specifică ambele sale adrese IP. Urmează trei variante de tratare a poștei electronice trimise la *flits.cs.vu.nl*. Prima alegere este, în mod natural, chiar *flits*, iar dacă nu se reușește, se încearcă la *zephyr* și apoi la *top*. Urmează un pseudonim, *www.cs.vu.nl*, astfel ca această adresă să poată fi utilizată fără a specifica o anumită mașină. Crearea acestui pseudonim permite ca *cs.vu.nl* să schimbe serverul *www* fără invalidarea adresei folosite în mod curent pentru adresarea lui. Un argument similar este valabil pentru *ftp.cs.vu.nl*.

Următoarele patru linii conțin o înregistrare tipică pentru o stație de lucru, în acest caz *rowboat.cs.vu.nl*. Informația furnizează adresa IP, destinația primară și secundară pentru poșta electronică și

informații despre mașină. Urmează o intrare pentru un sistem non-UNIX care nu este capabil să primească poșta el însuși, urmat de o intrare pentru o imprimantă laser conectată la Internet.

Ceea ce nu este arătat (și nu există în acest fișier) sunt adresele IP utilizate pentru a căuta adresele domeniilor de pe primul nivel. Acestea sunt necesare pentru a căuta sistemele gazdă aflate la distanță, dar, deoarece ele nu fac parte din domeniul *cs.vu.nl*, nu se găsesc în acest fișier. Ele sunt furnizate de serverele rădăcină ale căror adrese IP sunt prezentate în fișierul de configurare a sistemului și sunt încărcate în memoria ascunsă DNS atunci când este pornit serverul DNS. Există cam o duzină de servere rădăcină în lume și fiecare știe adresele IP ale tuturor celorlalte servere de domenii de nivel superior. Astfel, dacă o mașină știe adresa IP a cel puțin unuia din serverele rădăcină, el poate căuta orice nume DNS.

7.1.3 Servere de nume

Teoretic, un singur server de nume poate conține întreaga bază de date DNS și poate să răspundă tuturor cererilor. În practică, acest server poate fi atât de încărcat, încât să devină de neutilizat. În afară de aceasta, dacă se defectează, va fi afectat întregul Internet.

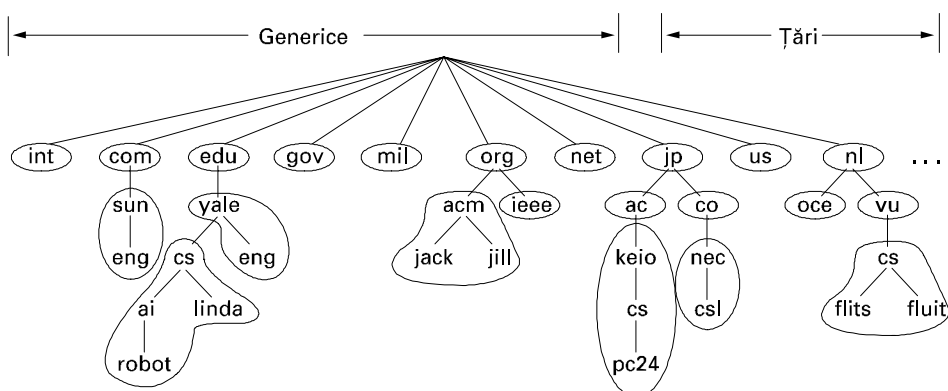


Fig. 7-4. O parte din spațiul numelor DNS prezentând împărțirea în zone.

Pentru a evita problemele asociate cu existența unei singure surse de informație, spațiul de nume DNS este împărțit în **zone** care nu se suprapun. O posibilă cale de împărțire a spațiului de nume din fig. 7-1 este arătată în fig. 7-4. Fiecare zonă conține câte o parte a arborelui precum și numele serverelor care păstrează informația autorizată despre acea zonă. În mod normal, o zonă va avea un server de nume primar, care preia informația dintr-un fișier de pe discul propriu și unul sau mai multe servere de nume secundare, care iau informațiile de pe serverul primar. Pentru a îmbunătăți fiabilitatea, unele servere pentru o zonă pot fi plasate în afara zonei.

Plasarea limitelor unei zone este la latitudinea administratorului ei. Această decizie este luată în mare parte bazându-se pe câte servere de nume sunt dorite și unde să fie plasate. De exemplu, în fig. 7-4, Yale are un server pentru *yale.edu* care administrează *eng.yale.edu*, dar nu și *cs.yale.edu*, care este o zonă separată cu propriile servere de nume. O astfel de decizie poate fi luată atunci când un departament ca cel de engleză nu dorește să aibă propriul server de nume, în schimb departamentul de calculatoare dorește. În consecință *cs.yale.edu* este o zonă separată, în timp ce zona *eng.yale.edu* nu este separată.

Atunci când un resolver are o cerere referitoare la un nume de domeniu, el transferă cererea unuia din serverele locale de nume. Dacă domeniul căutat este sub jurisdicția serverului de nume, cum ar fi *ai.cs.yale.edu*, care este sub *cs.yale.edu*, el reîntoarce înregistrări de resurse autorizate. O **înregistrare autorizată (authoritative record)** este cea care vine de la autoritatea care administrează înregistrarea și astfel este întotdeauna corectă. Înregistrările autorizate se deosebesc de înregistrările din memoria ascunsă, care pot fi expirate.

Dacă, totuși, domeniul se află la distanță, iar local nu este disponibilă nici o informație despre domeniul cerut, atunci serverul de nume trimite un mesaj de cerere la serverul de nume de pe primul nivel al domeniului solicitat. Pentru a clarifica acest proces să considerăm exemplul din fig. 7-5. Aici resolverul de pe *flits.cs.vu.nl* dorește să știe adresa IP a sistemului gazdă *linda.cs.yale.edu*. În pasul 1 trimite o cerere la serverul de nume local *cs.vu.nl*. Această cerere conține numele de domeniu căutat, tipul (A) și clasa (IN).

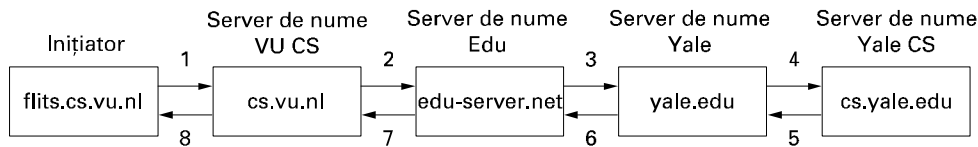


Fig. 7-5. Modul în care un resolver caută un nume aflat la distanță, în opt pași.

Să presupunem că serverul local de nume nu a avut niciodată o cerere pentru acest domeniu și nu știe nimic despre el. Poate solicita informații de la câteva servere de nume din apropiere, dar dacă nici unul dintre ele nu știe, va trimite un pachet UDP la serverul pentru *edu* specificat în baza de date (vezi fig. 7-5), *edu-server.net*. Este puțin probabil ca acest server să cunoască adresa *linda.cs.yale.edu* și probabil nu cunoaște nici adresa *cs.yale.edu*, dar trebuie să cunoască adresele fiilor din subarbore, astfel că va transmite cererea la serverul de nume *yale.edu* (pas 3). Acesta va transmite cererea mai departe către *cs.yale.edu* (pas 4), care trebuie să aibă înregistrările autorizate de resurse. Deoarece fiecare cerere este de la un client la un server, înregistrarea de resursă cerută parcurge pașii 5 până la 8.

Odată ce aceste înregistrări de resurse ajung înapoi la serverul de nume *cs.vu.nl*, ele vor fi depuse în memoria ascunsă, pentru a fi folosite ulterior. Totuși, această informație nu este autorizată, deoarece orice schimbare făcută la *cs.yale.edu* nu se va propaga spre toate serverele care au folosit-o. Din acest motiv intrările în memoria ascunsă nu ar trebui să aibă viață prea lungă. Acesta este motivul pentru care câmpul *Timp_de_viață* este inclus în fiecare înregistrare de resursă. El informează serverele de nume aflate la distanță cât timp să mențină înregistrările în memoria ascunsă. Dacă o anumită mașină are de ani de zile aceeași adresă IP, această informație ar putea fi păstrată timp de o zi. Pentru informații mai volatile este mai sigur ca înregistrările să fie eliminate după câteva secunde sau un minut.

De menționat că metoda de interogare descrisă aici este cunoscută ca metoda de **interogare recursivă (recursive query)**, deoarece fiecare server care nu are informația cerută o caută în altă parte și raportează. Este posibilă și o altă variantă. În acest caz, atunci când o cerere nu poate fi rezolvată local, cererea eșuează, dar este întors numele următorului server de pe calea ce trebuie încercată. Unele servere nu implementează interogarea recursivă și întorc întotdeauna numele următorului server la care să se încerce.

De asemenea merită menționat faptul că atunci când un client DNS nu reușește să primească un răspuns înainte de expirarea timpului de căutare, data viitoare va încerca un alt server. Se presupune că serverul este probabil nefuncțional, nu că cererea sau răspunsul s-au pierdut.

Deși DNS este foarte important pentru funcționarea corectă a Internetului, el nu face decât să pună în corespondență nume simbolice de mașini cu adresele lor IP. El nu ajută la localizarea oamenilor, resurselor, serviciilor sau obiectelor în general. Pentru localizarea acestora a fost definit un alt serviciu director, numit **LDAP (Lightweight Directory Access Protocol, rom.: Protocol de acces la cataloage simplificate)**. El este o versiune simplificată a serviciului de cataloage OSI X.500, descris în RFC 2251. El organizează informația sub formă de arbore și permite căutări pe diferite componente. Poate fi privit ca o carte de telefon obișnuită (de tipul „pagini albe”). Nu o să intrăm în amănunte referitoare la el în această carte, însă puteți găsi mai multe informații în (Weltman și Dahbura, 2000).

7.2 POȘTA ELECTRONICĂ

Poșta electronică, sau **e-mail**, cum este cunoscută de către numeroșii săi admiratori, există de peste două decenii. Înainte de 1990, era folosită în special în mediul academic. În timpul anilor 1990, a devenit cunoscută publicului larg și a crescut exponențial până la punctul în care numărul de mesaje electronice trimise pe zi este acum mult mai mare decât numărul de scrisori tradiționale (adică pe hârtie).

E-mail-ul, ca majoritatea celorlalte forme de comunicare, are convențiile și stilurile sale proprii. În particular, el este foarte neprotocolar și are un prag de folosire foarte scăzut. Oamenii care n-ar visa niciodată să sune la telefon sau chiar să scrie o scrisoare unei Persoane Foarte Importante nu ezită o secundă să trimită un e-mail neglijent.

E-mail-ul este plin de jargoane precum BTW (By The Way - apropo), ROTFL (Rolling On The Floor Laughing – a se tăvăli pe jos de râs) și IMHO (In My Humble Opinion - după umila mea părere). Mulți oameni folosesc în e-mail-urile lor câteva caractere ASCII numite **smileys** sau **emoticons** (fața zâmbitoare și fața tristă). Câteva din cele mai interesante sunt reproduse în fig. 7-6. Pentru cei mai mulți, rotirea cărții cu 90 de grade în sensul acelor de ceasornic, le va face mai clare. Pentru o cârtică cu peste 650 smileys, vedeți (Sanderson și Dougherty, 1993).

Smiley	Semnificație	Smiley	Semnificație	Smiley	Semnificație
:~)	Sunt fericit	=!:-)	Abe Lincoln	:+)	Nas mare
:-(Sunt trist, supărat	=)::-)	Unchiul Sam	:~))	Gușă
:	Sunt apatic	*<:-)	Moș Crăciun	:-(}	Mustață
:~)	Fac cu ochiul	<:-{	Dunce	#:-)	Păr încurcat
:-(0)	Tip	(:-)	Australian	8-)	Poartă ochelari
:-{*)	Vomit	:~)X	Om cu papion	C:-)	Cap mare

Fig. 7-6. Câteva smileys. Nu vor fi în examenul final :-)

Primele sisteme de poșta electronică constau pur și simplu din protocoale de transfer de fișiere, cu convenția ca prima linie a fiecărui mesaj (adică fișier) să conțină adresa receptorului. Cu timpul, limitările acestei abordări au devenit din ce în ce mai evidente. O parte dintre neajunsuri erau:

1. Trimiterea unui mesaj către un grup de persoane era incomodă. Managerii au nevoie adesea de această facilitare pentru a trimite note și rapoarte tuturor subordonaților.
2. Mesajele nu aveau structură internă, făcând astfel dificilă prelucrarea lor cu ajutorul calculatorului. De exemplu, dacă un mesaj trimis mai departe era inclus în corpul altui mesaj, extragerea părții incluse din mesajul primit era dificilă.
3. Inițiatorul (transmițătorul) nu știa niciodată dacă mesajul a ajuns sau nu.
4. Dacă cineva avea în plan să plece în călătorie de afaceri pentru mai multe săptămâni și dorea ca toată poșta primită în acest timp să fie preluată de către secretară, acest lucru nu era ușor de realizat.
5. Interfața cu utilizatorul era slab integrată cu sistemul de transmisie, cerând utilizatorilor ca întâi să editeze un fișier, apoi să părăsească editorul și să apeleze programul de transfer de fișiere.
6. Nu era posibilă transmiterea de mesaje care să conțină o combinație de text, desene, facsimil și voce.

Pe măsură ce s-a câștigat experiență, au fost propuse sisteme de poștă electronică mai complicate. În 1982 au fost publicate propunerile cu privire la e-mail ale ARPANET, sub numele de RFC 821 (protocolul de transmisie) și RFC 822 (formatul mesajelor). Revizii minore, RFC 2821 și RFC 2822, au devenit standarde Internet, totuși toată lumea se referă la e-mail gândindu-se la RFC 822.

În 1984, CCITT a emis recomandarea X.400. După două decenii de competiție, sistemele de poștă electronică bazate pe RFC 822 sunt larg răspândite, în timp ce acelea bazate pe X.400 au dispărut. Modul în care un sistem încropit de o mână de absolvenți de știința calculatoarelor a învins un standard internațional oficial, puternic susținut de către toate PTT-urile din lumea întreagă, de multe guverne și de o parte substanțială a industriei calculatoarelor, ne aduce în minte povestea biblică a lui David și Goliat.

Motivul succesului lui RFC822 nu este dat de faptul că ar fi atât de bun, ci acela că X.400 a fost atât de slab proiectat și atât de complex, încât nimeni nu l-ar putea implementa bine. Având de ales între un sistem nesofisticat, dar care funcționează, cum este cel bazat pe RFC822 și sistemul de e-mail X.400, presupus cu adevărat minunat, dar nefuncțional, majoritatea organizațiilor l-au ales pe primul. Poate că este și o lecție în spatele acestei povești. De aceea discuția noastră referitoare la e-mail se va concentra asupra sistemului de e-mail din Internet.

7.2.1 Arhitectură și servicii

În această secțiune vom furniza o prezentare de ansamblu a ceea ce pot face sistemele de poștă electronică și cum sunt ele organizate. Aceste sisteme constau de obicei din două subsisteme: **agenții-utilizator**, care permit utilizatorilor să citească și să trimită scrisori prin poșta electronică și **agenții de transfer de mesaje**, care transportă mesajele de la sursă la destinație. Agenții-utilizator sunt programe locale, care furnizează o metodă de a interacționa cu sistemul de e-mail bazată pe comenzi, meniuri sau grafică. Agenții de transfer de mesaje sunt, de regulă, **demoni** de sistem, adică procese care se execută în fundal. Sarcina lor este să transfere mesajele prin sistem.

În general, sistemele de poștă electronică pun la dispoziție cinci funcții de bază. Să aruncăm o privire asupra lor.

Compunerea se referă la procesul de creare a mesajelor și a răspunsurilor. Deși pentru corpul mesajului poate fi folosit orice editor de texte, sistemul însuși poate acorda asistență la adresare și la

completarea numeroaselor câmpuri antet atașate fiecărui mesaj. De exemplu, când se răspunde la un mesaj, sistemul poate extrage adresa inițiatorului din mesajul primit și o poate insera automat în locul potrivit din cadrul răspunsului.

Transferul se referă la deplasarea mesajului de la autor la receptor. În mare, aceasta necesită stabilirea unei conexiuni la destinație, sau la o mașină intermediară, emiterea mesajului și eliberarea conexiunii. Sistemul de poștă ar trebui să facă acest lucru singur, fără a deranja utilizatorul.

Raportarea se referă la informarea inițiatorului despre ce s-a întâmplat cu mesajul. A fost livrat? A fost respins? A fost pierdut? Există numeroase aplicații în care confirmarea livrării este importantă și poate avea chiar semnificație juridică. („Știți, domnule judecător, sistemul meu de poștă electronică nu e foarte de încredere, așa că presupun că citația electronică s-a pierdut pe undeva.”)

Afișarea mesajelor primite este necesară pentru ca utilizatorii să-și poată citi poșta. Uneori sunt necesare conversii sau trebuie apelat un program de vizualizare special; de exemplu, dacă mesajul este un fișier PostScript, sau voce digitizată. Se mai încearcă uneori și conversii simple și formatare.

Dispoziția este pasul final și se referă la ceea ce face receptorul cu mesajul, după ce l-a primit. Posibilitățile includ eliminarea sa înainte de a-l citi, aruncarea sa după citire, salvarea sa ș.a.m.d. Ar trebui de asemenea să fie posibilă regăsirea și recitirea de mesaje deja salvate, trimiterea lor mai departe, sau procesarea lor în alte moduri.

În plus față de aceste servicii de bază, unele sisteme de e-mail, în special cele interne companiilor, dispun de o gamă variată de facilități avansate. Să menționăm pe scurt câteva dintre ele. Când utilizatorii se deplasează sau când sunt plecați pentru o perioadă de timp, pot dori ca poșta lor să fie trimisă acolo unde se găsesc, așa că sistemul ar trebui să fie capabil să facă acest lucru automat.

Majoritatea sistemelor permit utilizatorilor să-și creeze **cutii poștale (mailboxes)** pentru a păstra mesajele sosite. Sunt necesare comenzi de creare și distrugere a cutiilor poștale, de inspectare a conținutului acestora, de inserare și de ștergere de mesaje din cutii poștale ș.a.m.d.

Managerii de companii au adesea nevoie să trimită un același mesaj fiecărui subordonat, client sau furnizor. Acest lucru dă naștere ideii de **listă de poștă (mailing list)**, care este o listă de adrese de poștă electronică. Când un mesaj este trimis la lista de poștă, copii identice ale sale sunt expediate fiecăruia dintre cei de pe listă.

Alte caracteristici evaluate sunt copii la indigo, poștă de prioritate mare, poștă secretă (criptată), receptori alternativi, dacă cel primar nu este disponibil, și posibilitatea de a permite secretarelor să se ocupe de poșta primită de șefii lor.

Poșta electronică este în prezent folosită pe scară largă în industrie, pentru comunicație în cadrul companiilor. Aceasta permite unor angajați, răspândiți la distanțe mari unii de ceilalți, chiar și peste mai multe fusuri orare, să coopereze la proiecte complexe. Eliminând majoritatea indiciilor cu privire la funcție, vârstă și gen, dezbaterile prin poșta electronică tind să se concentreze asupra ideilor și nu a statutului din cadrul organizației. Prin poșta electronică, o idee sclipitoare a unui student la cursurile de vară poate avea un impact mai mare decât una stupidă, venită de la un vicepreședinte executiv.

O idee fundamentală în toate sistemele moderne de e-mail este distincția dintre **plic** și conținutul său. Plicul încapsulează mesajul. Conține toată informația necesară pentru transportul mesajului, cum ar fi destinația, adresa, prioritatea, nivelul de securitate, toate acestea fiind distincte de mesajul în sine. Agenții de transfer de mesaje folosesc plicul pentru rutare (dirijare), așa cum face și oficiul poștal.

Mesajul din interiorul plicului conține două părți: **antetul** și **corpul**. Antetul conține informație de control pentru agenții utilizator. Corpul mesajului se adresează în întregime utilizatorului uman. Plicurile și mesajele sunt ilustrate în fig. 7-7.

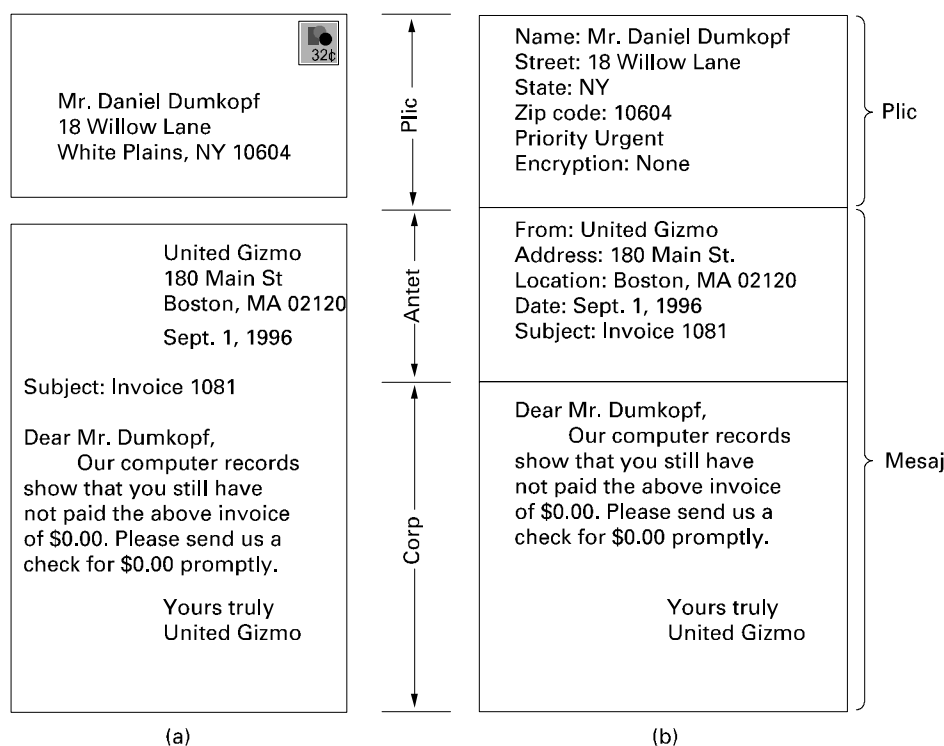


Fig. 7-7. Plicuri și mesaje. (a) poștă clasică (b) poștă electronică.

7.2.2 Agentul utilizator

Sistemele de poștă electronică au, așa cum am văzut, două părți esențiale: agenții-utilizator și agenții de transfer de mesaje. În această secțiune ne vom uita la agenții utilizator. Un agent utilizator este de obicei un program (uneori numit cititor de poștă) care acceptă o varietate de comenzi pentru compunerea, primirea și răspunsul la mesaje, cât și pentru manipularea cutiilor poștale. Unii agenți-utilizator au o interfață sofisticată, dirijată prin meniuri sau icoane, care necesită un maus, în timp ce altele acceptă comenzi de câte un caracter, date de la tastatură. Funcțional însă, toți aceștia sunt identici. Unele sisteme sunt dirijate prin meniuri sau icoane dau au și alternative mai „scurte” pe tastatură.

Trimiterea poștei electronice

Pentru a trimite un mesaj prin poșta electronică, un utilizator trebuie să furnizeze mesajul, adresa destinație, și eventual alți câțiva parametri. Mesajul poate fi produs cu un editor de texte de sine-stătător, cu un program de procesare de text sau, eventual, cu un editor de texte specializat, construit în interiorul agentului utilizator. Adresa de destinație trebuie să fie într-un format cu care agentul utilizator să poată lucra. Mulți agenți-utilizator solicită adrese de forma *utilizator@adresă-dns*. Deoarece aceste lucruri au fost studiate anterior în acest capitol, nu vom relua materialul respectiv aici.

Oricum, merită notat că există și alte forme de adresare. În particular, adresele X.400 arată radical diferit de cele DNS. Ele sunt compuse din perechi de forma *atribut = valoare*, separate de bare oblice. De exemplu:

/C=US/SP=MASSACHUSETTS/L=CAMBRIDGE/PA=360 MEMORIAL DR./CN=KEN SMITH/

Această adresă specifică o țară, un stat, o localitate, o adresă personală și un nume obișnuit (Ken Smith). Sunt posibile multe alte attribute, astfel încât poți trimite mesaje cuiva al cărui nume nu-l știi, atâta timp cât știi suficiente alte attribute (de exemplu, compania și funcția). Cu toate că adresele X.400 sunt mult mai puțin convenabile decât cele DNS, cele mai multe sisteme de poșta electronică permit folosirea de **pseudonime** (**aliases**, uneori numite și porecle) pentru obținerea numelor sau adreselor de e-mail corecte ale unei persoane. În consecință, chiar și cu adresele de tip X.400, de obicei nu este necesară scrierea în întregime a acelor șiruri ciudate.

Majoritatea sistemelor de e-mail acceptă liste de poștă, astfel că un utilizator poate trimite, cu o singură comandă, un același mesaj tuturor persoanelor dintr-o listă. Dacă lista de poștă este păstrată local, agentul-utilizator poate pur și simplu să trimită câte un mesaj separat fiecăruia dintre receptori doriți. Dacă lista este păstrată la distanță, atunci mesajele vor fi expandate acolo. De exemplu, dacă un grup de admiratori de păsări au o listă de poștă numită *birders*, instalată la *meadowlark.arizona.edu*, atunci orice mesaj trimis la *birders@meadowlark.arizona.edu* va fi dirijat către Universitatea din Arizona și expandat acolo în mesaje individuale pentru toți membrii listei de poștă, oriunde ar fi ei în lume. Utilizatorii acestei liste de poștă nu pot determina că aceasta este o listă de adrese. Ar putea fi la fel de bine cutia poștală personală a Prof. Gabriel O. Birders.

Citirea poștei electronice

În mod obișnuit, când este lansat un agent-utilizator, înainte de a afișa ceva pe ecran, el se va uita în cutia poștală a utilizatorului după mesajele care sosesc. Apoi poate anunța numărul de mesaje din cutie, sau poate afișa pentru fiecare mesaj câte un rezumat de o linie, pentru ca apoi să aștepte o comandă.

Ca exemplu despre cum lucrează un agent-utilizator, să aruncăm o privire asupra unui scenariu tipic pentru poșta electronică. După lansarea agentului-utilizator, utilizatorul cere un rezumat al mesajelor sale. O imagine ca aceea din fig. 7-8 apare în acest caz pe ecran. Fiecare linie se referă la câte un mesaj. În acest exemplu, cutia poștală conține opt mesaje.

#	Marcaje	Octeți	Transmițător	Subiect
1	K	1030	Asw	Changes to MINIX
2	KA	6348	Trudy	Not all Trudys are nasty
3	K F	4519	Amy N. Wong	Request for information
4		1236	Bal	Bioinformatics
5		103610	Kaashoek	Material on peer-to-peer
6		1223	Frank	Re: Will you review a grant proposal
7		3110	Guido	Our paper has been accepted
8		1204	Dmr	Re: My student's visit

Fig. 7-8. Un exemplu de afișare a conținutului unei cutii poștale.

Fiecare linie de afișaj conține câteva câmpuri extrase de pe plicul sau din antetul mesajului corespunzător. Într-un sistem simplu de poșta electronică, alegerea câmpurilor afișate este făcută în cadrul programului. Într-un sistem mai sofisticat, utilizatorul poate specifica ce câmpuri să fie afișate, furnizând un **profil al utilizatorului**, adică un fișier care descrie formatul de afișare. În exemplul

considerat, primul câmp reprezintă numărul mesajului. Al doilea câmp, *Marcaje*, poate conține un *K*, însemnând că mesajul nu este nou, dar a fost citit anterior și păstrat în cutia poștală; un *A*, însemnând că deja s-a răspuns la acest mesaj; și/sau un *F*, însemnând că mesajul a fost trimis mai departe altcuiva. Sunt de asemenea posibile și alte marcaje.

Al treilea câmp specifică lungimea mesajului și al patrulea spune cine a trimis mesajul. Din moment ce el este pur și simplu extras din mesaj, acest câmp poate conține prenume, nume complete, inițiale, nume de cont, sau orice altceva și-a ales transmitătorul să pună. În sfârșit, câmpul *Subiect* specifică despre ce este mesajul, într-un scurt rezumat. Persoanele care omit să includă un câmp *Subiect* adesea descoperă că răspunsurile la scrisorile lor tind să nu obțină prioritate maximă.

După ce au fost afișate antetele, utilizatorul poate executa oricare dintre comenzile disponibile, ca de exemplu afișarea unui mesaj, ștergerea unui mesaj și așa mai departe. Sistemele mai vechi lucrează în mod text și de obicei foloseau comenzi de un caracter pentru diversele operații, ca de exemplu *T* (scrie mesaj), *A* (răspunde la mesaj), *D* (șterge mesaj) și *F* (trimite mai departe). Un argument specifică mesajul corespunzător. Sistemele mai recente folosesc interfețe grafice. De obicei, utilizatorul selectează un mesaj cu mouse-ul și apoi apasă pe o iconiță pentru a scrie, răspunde la mesaj, sau pentru a-l trimite mai departe.

Poșta electronică a parcurs un drum lung de pe vremea când era doar transfer de fișiere. Agenți-utilizator sofisticată fac posibilă manevrarea unui volum mare de scrisori. Pentru persoane care primesc și trimit mii de mesaje pe an, asemenea instrumente sunt neprețuite.

7.2.3 Formatele mesajelor

Să ne întoarcem acum de la interfața utilizator la formatul mesajelor de poștă electronică în sine. Mai întâi ne vom uita la e-mailul ASCII de bază, care utilizează RFC 822. După aceea ne vom concentra asupra extensiilor multimedia ale RFC 822.

RFC 822

Mesajele constau dintr-un plic simplu (descriș în RFC 821), un număr de câmpuri antet, o linie goală și apoi corpul mesajului. Fiecare câmp antet se compune (din punct de vedere logic) dintr-o singură linie de text ASCII, conținând numele câmpului, două puncte, și, pentru majoritatea câmpurilor, o valoare. RFC 822 a fost creat acum două decenii și nu distinge clar plicul de câmpurile antet, cum ar face un standard nou. Cu toate că a fost corectat în RFC 2822, o refacere completă n-a fost posibilă datorită răspândirii sale largi. La o utilizare normală, agentul-utilizator construiește un mesaj și îl transmite agentului de transfer de mesaje, care apoi folosește unele dintre câmpurile antet pentru a construi plicul efectiv, o combinație oarecum demodată de mesaj și plic.

Principalele câmpuri antet, legate de transportul de mesaje, sunt înfățișate în fig. 7-9. Câmpul *To*: oferă adresa DNS a receptorului primar. Este permisă de asemenea existența de receptori multipli. Câmpul *Cc*: dă adresa oricărui receptor secundar. În termenii livrării, nu este nici o diferență între un receptor primar și unul secundar. Este în întregime o deosebire psihologică, ce poate fi importantă pentru persoanele implicate, dar este neimportantă pentru sistemul de poștă. Termenul *Cc*: (Carbon copy - copie la indigo) este puțin depășit, din moment ce calculatoarele nu folosesc indigo, dar este bine înrădăcinat. Câmpul *Bcc*: (Blind carbon copy - copie confidențială la indigo) este la fel ca *Cc*:, cu excepția că această linie este ștearsă din toate copiile trimise la receptorii primari și secundari. Acest element permite utilizatorilor să trimită copii unei a treia categorii de receptori, fără ca cei primari și secundari să știe acest lucru.

Antet	Conținut
To:	Adresa(ele) de e-mail a(le) receptorului(ilor) primar(i)
Cc:	Adresa(ele) de e-mail a(le) receptorului(ilor) secundar(i)
Bcc:	Adresa(ele) de e-mail pentru „blind carbon copy”
From:	Persoana sau persoanele care au creat mesajul
Sender:	Adresa de e-mail a transmițătorului curent
Received:	Linie adăugată de fiecare agent de transfer de-a lungul traseului
Return-Path:	Poate fi folosită pentru a identifica o cale de întoarcere la transmițător

Fig. 7-9. Câmpurile antet ale lui RFC 822, legate de transportul de mesaje.

Următoarele două câmpuri, *From:* și *Sender:*, precizează cine a scris și respectiv cine a trimis mesajul. Acestea pot să nu fie identice. De exemplu, se poate ca o directoare executivă să scrie un mesaj, dar ca secretara ei să fie cea care îl trimite efectiv. În acest caz, directoarea executivă va fi afișată în câmpul *From:* și secretara în câmpul *Sender:*. Câmpul *From:* este obligatoriu, dar câmpul *Sender:* poate fi omis dacă este identic cu *From:*. Aceste câmpuri sunt necesare în cazul în care mesajul nu poate fi livrat și trebuie returnat transmițătorului.

O linie conținând *Received:* este adăugată de fiecare agent de transfer de mesaje de pe traseu. Linia conține identitatea agentului, data și momentul de timp la care a fost primit mesajul și alte informații care pot fi utilizate pentru găsirea defecțiunilor în sistemul de dirijare.

Câmpul *Return-Path:* este adăugat de agentul final de transfer de mesaje și are în intenție să indice cum se ajunge înapoi la transmițător. În teorie, această informație poate fi adunată din toate antetele *Received:* (cu excepția numelui cutiei poștale a transmițătorului), dar rareori este completată așa și de obicei conține chiar adresa transmițătorului.

Antet	Conținut
Date:	Data și momentul de timp la care a fost trimis mesajul
Reply-To:	Adresa de e-mail la care ar trebui trimise răspunsurile
Message-Id:	Număr unic, utilizat ulterior ca referință pentru acest mesaj (identificator)
In-Reply-To:	Identificatorul mesajului al cărui răspuns este mesajul curent
References:	Alți identificatori de mesaje relevanți
Keywords:	Cuvinte cheie alese de utilizator
Subject:	Scurt cuprins al mesajului, afișabil pe o singură linie

Fig. 7-10. Câteva câmpuri utilizate în antetul lui RFC 822.

În plus față de câmpurile din fig. 7-9, mesaje RFC 822 pot conține de asemenea o varietate de câmpuri antet, folosite de agenții-utilizator sau de receptorii umani. Cele mai des întâlnite dintre ele sunt prezentate în fig. 7-10. Majoritatea lor se explică de la sine, deci nu vom intra în detaliu la toate.

Câmpul *Reply-To:* este uneori utilizat când nici persoana care a compus mesajul, nici cea care l-a trimis nu vrea să vadă răspunsul. De exemplu, un director de marketing scrie un mesaj prin e-mail pentru a spune clienților despre un nou produs. Mesajul este trimis de o secretară, dar câmpul *Reply-To:* conține șeful departamentului de vânzări, care poate răspunde la întrebări și primi comenzi. Acest câmp este foarte folositor când transmițătorul are două conturi de e-mail și vrea ca răspunsul să ajungă în celălalt.

Documentul RFC 822 afirmă explicit că utilizatorilor le este permis să inventeze noi antete, atâta timp cât acestea încep cu șirul de caractere X-. Se garantează că nici o extindere ulterioară nu va folosi nume ce încep cu X-, pentru a evita conflictele (suprapunerile) dintre antetele oficiale și cele

personale. Uneori studenții care fac pe deșteptii includ câmpuri de tipul *X-Fruit-of-the-Day*: sau *X-Disease-of-the-Week*:, care sunt legale, deși nu întotdeauna clarificatoare.

După antete urmează corpul mesajului. Aici utilizatorii pot pune orice vor. Unii oameni își încheie mesajele cu semnături elaborate, incluzând caricaturi ASCII simple, citate din personalități mai mari sau mai mici, declarații politice și declinări de tot felul (de exemplu: Corporația XYZ nu este răspunzătoare pentru părerile mele; de fapt nu poate nici măcar să le înțeleagă).

MIME - Multipurpose Internet Mail Extensions (extensii de poștă cu scop multiplu)

La începuturile ARPANET, poșta electronică consta exclusiv din mesaje de tip text, scrise în engleză și exprimate în ASCII. Pentru acest context, RFC 822 realiza sarcina complet: specifică antetele, dar lăsa conținutul în întregime în seama utilizatorilor. În zilele noastre, această abordare nu mai este adecvată pentru Internetul care se întinde în lumea întreagă. Problemele includ transmiterea și recepția de:

1. Mesaje în limbi cu accente (de exemplu franceza și germana).
2. Mesaje în alfabet ne-latine (de exemplu ebraică și rusă).
3. Mesaje în limbi fără alfabet (de exemplu chineză și japoneză).
4. Mesaje care nu conțin text deloc (de exemplu audio și video).

O soluție posibilă a fost propusă în RFC 1341 și actualizată în RFC-urile 2045-2049. Această soluție, numită **MIME (Multipurpose Internet Mail Extensions)**, este în prezent larg utilizată. O vom descrie în continuare. Pentru informații suplimentare în legătură cu MIME, vedeți RFC-urile.

Ideea fundamentală a MIME este să continue să folosească formatul RFC 822, dar să adauge structură corpului mesajului și să definească reguli de codificare pentru mesajele non-ASCII. Deoarece respectă RFC 822, mesajele MIME pot fi trimise utilizând programele și protocoalele de poștă existente. Tot ceea ce trebuie modificat sunt programele de transmitere și recepție, pe care utilizatorii le pot face ei înșiși.

Antet	Conținut
MIME-Version:	Identifică versiunea de MIME
Content-Description:	Șir adresat utilizatorului care spune ce este în mesaj
Content-Id:	Identificator unic
Content-Transfer-Encoding:	Cum este împachetat corpul pentru transmisie
Content-Type:	Natura mesajului

Fig. 7-11. Antetele RFC 822 adăugate de către MIME.

MIME definește cinci noi antete de mesaje, așa cum se arată în fig. 7-11. Primul dintre acestea specifică pur și simplu agentului-utilizator care primește mesajul că este vorba de un mesaj MIME și ce versiune de MIME utilizează. Orice mesaj care nu conține un antet *MIME-Version*: este presupus ca fiind un mesaj în text pur, în engleză, și este procesat ca atare.

Antetul *Content-Description*: este un șir de caractere ASCII specificând ce este în mesaj. Acest antet este necesar pentru ca receptorul să știe dacă merită să decodifice și să citească mesajul. Dacă șirul de caractere spune: “Fotografia hamsterului Barbarei” și persoana care primește mesajul nu este un mare iubitor de hamsteri, mesajul va fi probabil mai curând aruncat, decât decodificat într-o fotografie color de înaltă rezoluție.

Antetul *Content-Id*: identifică conținutul. Utilizează același format ca antetul standard *Message-Id*:

Antetul *Content-Transfer-Encoding*: arată cum este împachetat pentru transmisie corpul mesajului, într-o rețea care poate ridica obiecții la majoritatea caracterelor diferite de litere, cifre și semne de punctuație. Sunt furnizate cinci scheme (plus o evadare către noi scheme). Cea mai simplă schemă se referă chiar la text ASCII. Caracterele ASCII utilizează 7 biți și pot fi transportate direct prin protocolul de e-mail, atâta timp cât nici o linie nu are mai mult de 1000 de caractere.

Următoarea schemă ca simplitate este cam același lucru, dar utilizează caractere de câte 8 biți, reprezentând toate valorile de la 0 la 255 inclusiv. Această schemă de codificare încalcă protocolul (original) de e-mail utilizat în Internet, dar este folosită de unele părți ale Internetului, care implementează niște extensii ale protocolului original. În timp ce declararea codificării nu o face să devină legală, faptul că o avem explicit poate cel puțin să lămurească lucrurile atunci când ceva merge prost. Mesajele utilizând codificarea de 8 biți trebuie încă să respecte lungimea maximă a liniei, care este standard.

Este chiar mai rău în cazul mesajelor care utilizează codificare binară. Aceste mesaje sunt fișiere binare arbitrare, care nu numai că utilizează toți cei 8 biți, dar nu respectă nici limita de linie de 1000 de caractere. Programele executabile intră în această categorie. Nu se acordă nici o garanție că mesaje binare vor ajunge corect, dar mulți le trimit oricum.

Modalitatea corectă de a codifica mesaje binare este de a utiliza **codificarea în bază 64**, numită uneori **armură ASCII**. În această schemă, grupuri de câte 24 de biți sunt împărțite în patru unități de câte 6 biți, fiecare dintre aceste unități fiind transmisă ca un caracter ASCII legal. Codificarea este „A” pentru 0, „B” pentru 1, ș.a.m.d., urmate de cele 26 de litere mici, cele 10 cifre, și în cele din urmă + și / pentru 62 și respectiv 63. Secvențele == și = sunt utilizate pentru a arăta că ultimul grup a conținut doar 8 sau respectiv 16 biți. Se ignoră secvențele carriage return și line feed, astfel că ele pot fi inserate după dorință, pentru a păstra liniile suficient de scurte. Utilizând această schemă pot fi trimise sigur texte binare arbitrare.

Pentru mesajele care sunt aproape în întregime ASCII și conțin puține caractere ne-ASCII, codificarea în bază 64 este oarecum inefficientă. În locul acesteia se utilizează o codificare numită **quoted-printable-encoding** (codificare afișabilă marcată). Aceasta este o codificare de tip ASCII pe 7 biți, având toate caracterele cu cod mai mare de 127 codificate sub forma unui semn egal urmat de valoarea caracterului reprezentată prin două cifre hexazecimale.

Rezumând, datele binare ar trebui trimise codificate în bază 64 sau sub formă quoted-printable. Când există motive întemeiate pentru a nu utiliza una dintre aceste scheme, este posibil să se specifice în antetul Content-Transfer-Encoding: o codificare definită de către utilizator.

Ultimul antet înfățișat în fig. 7-11 este cu adevărat cel mai interesant. El specifică natura corpului mesajului. În RFC 2045 sunt definite șapte tipuri, fiecare având unul sau mai multe subtipuri. Tipul și subtipul sunt separate printr-o bară oblică (slash), ca în:

Content-Type: video/mpeg

Subtipul trebuie precizat explicit în antet; nu sunt furnizate valori implicite. Lista inițială de tipuri și subtipuri specificate în RFC 2045 este prezentată în fig. 7-12. De atunci au fost adăugate multe altele, introducându-se intrări adiționale de fiecare dată când a devenit necesar.

Să parcurgem acum lista tipurilor. Tipul *text* este utilizat pentru text simplu. Combinația *text/plain* este folosită pentru mesaje obișnuite care pot fi afișate de îndată ce sunt primite, fără codificare sau procesare ulterioară. Această opțiune permite ca mesajele obișnuite să fie transportate în MIME adăugând doar câteva antete suplimentare.

Tip	Subtip	Descriere
Text	Plain	Text neformatat
	Enriched	Text incluzând comenzi simple de formatare
Image	Gif	Imagini fixe în format GIF
	Jpeg	Imagini fixe în format JPEG
Audio	Basic	Sunet
Video	Mpeg	Film în format MPEG
Application	Octet-stream	Secvență neinterpretată de octeți
	Postscript	Un document afișabil în PostScript
Message	Rfc822	Un mesaj MIME RFC 822
	Partial	Mesajul a fost fragmentat pentru transmisie
	External-body	Mesajul în sine trebuie adus din rețea
Multipart	Mixed	Părți independente în ordine specificată
	Alternative	Același mesaj în formate diferite
	Parallel	Părțile trebuie vizualizate simultan
	Digest	Fiecare parte este un mesaj RFC 822 complet

Fig. 7-12. Tipurile și subtipurile aparținând MIME definite în RFC 2045.

Subtipul *text/enriched* permite includerea în text a unui limbaj simplu de marcare. Acest limbaj furnizează o modalitate independentă de sistem pentru a exprima scrierea cu caractere aldine sau cursive, dimensiunile, alinierea, distanțele dintre rânduri, folosirea de indici superiori sau inferiori și paginarea simplă. Limbajul de marcare se bazează pe SGML, Standard Generalized Markup Language (limbajul standard generalizat de marcare), folosit de asemenea ca bază pentru HTML, utilizat în World Wide Web. De exemplu mesajul

The <bold> time </bold> has come the <italic> walrus </italic> said ...

ar fi afișat sub forma:

The **time** has come the *walrus* said...

Depinde de sistemul receptor să aleagă interpretarea potrivită. Dacă sunt disponibile caractere aldine și cursive, acestea vor putea fi folosite; altfel, pentru a scoate în evidență se pot utiliza culori, scriere cu clipire sau video-invers etc. Sisteme diferite pot face alegeri diferite.

Când Web-ul a devenit popular, a fost adăugat un nou subtip, *text/html* (în RFC 2854) pentru a permite paginilor Web să fie trimise într-un e-mail de tip RFC 822. Un subtip pentru sistemul extins de marcare, *text/xml*, este definit în RFC 3023. Vom studia HTML și XML mai târziu în acest capitol.

Următorul tip MIME este *image*, utilizat pentru trimiterea de imagini fixe. În zilele noastre sunt utilizate multe formate, atât cu, cât și fără compresie, pentru a păstra și transmite imagini. Două dintre acestea, GIF și JPEG, sunt recunoscute de aproape toate programele de navigare, dar există și altele care au fost adăugate la lista originală.

Tipurile *video* și *audio* sunt pentru imagini în mișcare și respectiv pentru imagini cărora li se asociază și sunet. Trebuie notat că *video* include doar informația video, nu și coloana sonoră. Dacă trebuie transmis un film cu sunet, s-ar putea ca porțiunile audio și video să trebuiască să fie transmise separat, depinzând de sistemul de codificare utilizat. Primul format video definit a fost cel inventat de cei ce se intitulează modest Moving Picture Experts Group (MPEG - Grupul de experți în imagini în mișcare), dar de atunci au fost adăugate și altele. În plus față de *audio/basic*, un nou tip audio, *audio/mpeg* a fost adăugat în RFC 3003 pentru a permite oamenilor să trimită fișiere MP3 prin e-mail.

Tipul *application* este utilizat ca un colector pentru formatele care necesită prelucrare externă, neidentificate de nici unul dintre celelalte tipuri. Un *octet-stream* este doar o secvență de octeți nein-

terpretați. La primirea unui asemenea flux, un agent-utilizator ar trebui probabil să-l afișeze, sugerându-i utilizatorului să-l copieze într-un fișier și cerându-i un nume pentru acesta. Procesarea ulterioară este apoi la latitudinea utilizatorului.

Celălalt subtip definit este *postscript*, care se referă la limbajul PostScript, produs de Adobe Systems și larg utilizat pentru descrierea paginilor imprimate. Multe imprimante au înglobate interpretoare PostScript. Deși un agent-utilizator poate pur și simplu să apeleze un interpretor PostScript extern pentru a interpreta fișierele PostScript primite, acest lucru nu este lipsit de pericole. PostScript este un întreg limbaj de programare. Dându-i-se destul timp, o persoană suficient de masochistă ar putea scrie în PostScript un compilator de C, sau un sistem de management de baze de date. Afișarea unui mesaj primit în format PostScript se face executând programul PostScript conținut de acesta. Pe lângă afișarea unui text, acest program poate citi, modifica, sau șterge fișierele utilizatorului și poate avea și alte efecte laterale neplăcute.

Tipul *message* permite încapsularea în întregime a unui mesaj în altul. Această schemă este utilă, de exemplu pentru trimiterea mai departe a e-mailului, cu *forward*. Când un mesaj RFC 822 complet este încapsulat într-un mesaj exterior, ar trebui utilizat subtipul *rfc822*.

Subtipul *partial* face posibilă împărțirea unui mesaj încapsulat în bucăți de mesaj și trimiterea separată a acestora (de exemplu, dacă mesajul încapsulat este prea lung). Parametrii fac posibilă reasamblarea în ordinea corectă a tuturor părților, la destinație.

Și în sfârșit, subtipul *external-body* poate fi utilizat pentru mesaje foarte lungi (de exemplu, filme video). În loc de a include fișierul MPEG în mesaj, se dă o adresă FTP și agentul utilizator al receptorului o poate aduce din rețea în momentul în care este necesar. Această facilitate este în special utilă când se trimite un film la o întreagă listă de poștă și se presupune că doar câțiva dintre membrii acesteia îl vor vedea (gândiți-vă la e-mailurile inutile, conținând reclame video).

From: elinor@abcd.com

To: carolyn@xyz.com

MIME-Version: 1.0

Message-Id: <0704760941.AA00747@abcd.com>

Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm

Subject: Pământul înconjoară soarele de un număr întreg de ori

Acesta este preambulul. Agentul utilizator îl ignoră. O zi bună.

--qwertyuiopasdfghjklzxcvbnm

Content-Type: text/richtext

Happy birthday to you

Happy birthday to you

Happy birthday dear <bold> Carolyn </bold>

Happy birthday to you

--qwertyuiopasdfghjklzxcvbnm

Content-Type: message/external-body;

access-type=„anon-ftp”;

site=„bicycle.abcd.com”;

directory=„pub”;

name=„birthday.snd”;

content-type: audio/basic

content-transfer-encoding: base64

--qwertyuiopasdfghjklzxcvbnm

Fig. 7-13. Un mesaj multipart conținând alternative de tip text formatat și audio.

Ultimul tip este *multipart*, care permite unui mesaj să conțină mai multe părți, începutul și sfârșitul fiecărei părți fiind clar delimitat. Subtipul *mixed* permite fiecărei părți să fie diferită de celelalte, fără a avea o structură adițională impusă. Multe programe de e-mail permit utilizatorului să aibă una sau mai multe părți atașate la un mesaj text. Acestea sunt trimise folosind tipul *multipart*.

În contrast cu tipul *multipart*, subtipul *alternative* permite ca fiecare parte să conțină același mesaj, dar exprimat într-un alt mediu sau într-o codificare diferită. De exemplu, un mesaj ar putea fi trimis ca ASCII simplu, ca text formatat și ca PostScript. Un agent-utilizator proiectat corespunzător, la primirea unui asemenea mesaj, îl va afișa, dacă va fi posibil, în PostScript. A doua alegere va fi textul formatat. Dacă nici una dintre aceste alternative nu ar fi posibilă, s-ar afișa text ASCII obișnuit. Părțile ar trebui ordonate de la cea mai simplă, la cea mai complexă, pentru a ajuta receptorii care folosesc agenți-utilizator pre-MIME să înțeleagă mesajul (chiar și un utilizator pre-MIME poate citi text ASCII simplu). Subtipul *alternative* poate fi folosit de asemenea pentru limbaje multiple. În acest context, Rosetta Stone poate fi privit ca precursor al mesajului de tip *multipart/alternative*.

Un exemplu multimedia este prezentat în fig. 7-13. Aici, o felicitare este transmisă atât sub formă de text cât și sub formă de cântec. Dacă receptorul are facilități audio, agentul utilizator va aduce fișierul de sunet, *birthday.snd* și îl va interpreta. Dacă nu, versurile vor fi afișate pe ecran într-o liniște de mormânt. Părțile sunt delimitate de două cratime urmate de șirul (definit de utilizator) specificat în parametrul *boundary*.

Observați că antetul *Content-Type* apare în trei poziții în acest exemplu. La primul nivel indică faptul că mesajul are mai multe părți. În cadrul fiecărei părți specifică tipul și subtipul acesteia. În sfârșit, în corpul celei de-a doua părți, este necesar pentru a indica agentului utilizator ce fel de fișier extern trebuie să aducă. Pentru a exprima ușoara diferență de utilizare, s-au folosit litere mici, deși toate antetele sunt *case insensitive* (nu fac diferență între literele mari și cele mici). Antetul *content-transfer-encoding* este în mod similar necesar pentru orice corp extern care nu este codificat ca ASCII pe 7 biți.

Întorcându-ne la subtipurile corespunzătoare mesajelor *multipart*, vom spune că mai există două posibilități. Subtipul *parallel* este utilizat când toate părțile trebuie să fie interpretate simultan. De exemplu, adesea filmele au un canal audio și unul video. Ele sunt mai de efect dacă aceste două canale sunt interpretate în paralel și nu consecutiv.

În sfârșit, subtipul *digest* este utilizat când multe mesaje sunt împachetate împreună, într-unul compus. De exemplu, niște grupuri de dialog de pe Internet pot aduna mesaje de la abonații lor și apoi să le trimită în afară ca un singur mesaj de tip *multipart/digest*.

7.2.4 Transferul mesajelor

Sistemul de transfer de mesaje se ocupă cu transmiterea mesajelor de la expeditor la receptor. Cea mai simplă cale de a realiza acest lucru constă în stabilirea unei conexiuni de transport de la mașina sursă la cea de destinație și apoi, pur și simplu în trimiterea mesajului. După ce examinăm cum se face acest lucru în mod normal, vom studia câteva situații în care metoda nu funcționează și vom vedea ce trebuie făcut în aceste cazuri.

SMTP – Simple Mail Transfer Protocol (Protocol simplu de transfer de poștă)

În cadrul Internetului poșta electronică este livrată prin stabilirea de către mașina sursă a unei conexiuni TCP la portul 25 al mașinii de destinație. La acest port se află un demon de e-mail care știe SMTP (Simple Mail Transfer Protocol). Acest demon acceptă conexiunile și copiază mesajele

de la ele în cutiile poștale corespunzătoare. Dacă mesajul nu poate fi livrat, se returnează transmitătorului un raport de eroare conținând prima parte a mesajului nelivrat.

SMTP este un protocol simplu de tip ASCII. După stabilirea conexiunii TCP la portul 25, mașina transmițătoare, operând în calitate de client, așteaptă ca mașina receptoare, operând ca server, să vorbească prima. Serverul începe prin a trimite o linie de text, declarându-și identitatea și spunând dacă este pregătit sau nu să primească mesaje. Dacă nu este, clienții eliberează conexiunea și încearcă din nou mai târziu.

```
S: 220 xyz.com SMTP service ready
C: HELO abcd.com
S: 250 xyz.com says hello to abcd.com
C: MAIL FROM: <elinor@abcd.com>
S: 250 sender ok
C: RCPT TO: <carolyn@xyz.com>
S: 250 recipient ok
C: DATA
S: 354 Trimite mail; terminat cu "." pe linie nouă
C: From: elinor@abcd.com
C: To: carolyn@xyz.com
C: MIME-Version: 1.0
C: Message-Id: <0704760941.AA00747@abcd.com>
C: Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm
C: Subject: Pământul înconjoară soarele de un număr întreg de ori
C:
C: Acesta este preambulul. Agentul utilizator îl ignoră. O zi bună.
C:
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: text/enriched
C:
C: Happy birthday to you
C: Happy birthday to you
C: Happy birthday dear <bold> Carolyn </bold>
C: Happy birthday to you
C:
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: message/external-body;
C: access-type="anon-ftp";
C: site="bicycle.abcd.com";
C: directory="pub";
C: name="birthday.snd";
C:
C: content-type: audio/basic
C: content-transfer-encoding: base64
C: --qwertyuiopasdfghjklzxcvbnm
C: .
S: 250 message accepted
C: QUIT
S: 221 xyz.com closing connection
```

Fig. 7-14. Transferul unui mesaj de la *elinor@abcd.com* la *carolyn@zxy.com*.

Dacă serverul este dispus să primească e-mail, clientul anunță de la cine vine scrisoarea și cui îi este adresată. Dacă un asemenea receptor există la destinație, serverul îi acordă clientului permisiunea să trimită mesajul. Apoi clientul trimite mesajul și serverul îl confirmă. În general nu este necesară atașarea unei sume de control deoarece TCP furnizează un flux sigur de octeți. Dacă mai există și alte mesaje, acestea sunt trimise tot acum. Când schimbul de mesaje, în ambele direcții, s-a încheiat, conexiunea este eliberată. În fig. 7-14 este prezentată o mostră de dialog referitoare la trimiterea mesajului din fig. 7-13, incluzând codurile numerice utilizate de SMTP. Liniile trimise de client sunt marcate cu C:, iar cele trimise de server cu S:.

Câteva comentarii în legătură cu fig. 7-14 ar putea fi utile. Prima comandă a clientului este într-adevăr *HELO*. Din posibilele abrevieri de patru caractere ale cuvântului *HELLO*, aceasta are numeroase avantaje față de concurența sa cea mai mare. Motivul pentru care toate comenzile trebuiau să aibă patru caractere s-a pierdut în negura vremii.

În Fig. 7-14, mesajul este trimis la un singur receptor și de aceea este utilizată o singură comandă *RCPT*. Mai multe asemenea comenzi sunt permise pentru a trimite un singur mesaj mai multor receptori. Fiecare dintre ele este confirmată sau rejectată individual. Chiar dacă unii dintre receptori sunt rejectați (deoarece ei nu există la destinație), mesajul poate fi trimis celor rămași.

În sfârșit, deși sintaxa comenzilor de patru caractere de la client este rigid specificată, sintaxa replicilor este mai puțin rigidă. Doar codul numeric contează cu adevărat. Fiecare implementare poate pune după cod ce șiruri de caractere vrea.

Pentru a înțelege mai bine cum funcționează SMTP și câteva din celelalte protocoale descrise în acest capitol, încercați-le! În orice caz, mai întâi mergeți la o mașină conectată la Internet. Într-un sistem UNIX introduceți comanda:

```
telnet mail.isp.com 25
```

înlocuind numele DNS cu numele serverului de mail al ISP-ului dvs. Pe un sistem Windows, faceți clic pe Start, apoi Run, apoi tastați comanda în căsuța de dialog. Această comandă va stabili o conexiune Telnet (adică TCP) pe portul 25 pe mașina respectivă. Portul 25 este portul SMTP (vezi Fig. 6-27 pentru câteva porturi uzuale). Probabil o să primiți un răspuns de genul:

```
Trying 192.30.200.66...
Connected to mail.isp.com
Escape character is '^]'.
220 mail.isp.com Smail #74 ready at Thu, 25 Sept 2002 13:26 +0200
```

Primele trei linii spun ce face telnet-ul. Ultima linie este de la serverul SMTP de pe mașina de la distanță, anunțând disponibilitatea acesteia de a vorbi cu dvs. și de a accepta e-mail. Pentru a vedea ce comenzi acceptă, tastați:

```
HELP
```

De aici înainte, este posibilă o secvență de comenzi ca cea din fig. 7-14, începând cu comanda *HELO* dată de client.

E bine de notat faptul că folosirea liniilor de text ASCII pentru comenzi nu e un accident. Cele mai multe protocoale Internet funcționează așa. Folosirea textului ASCII face ca protocoalele foarte ușor de testat și depanat. Ele pot fi testate trimițând manual comenzi, cum am văzut mai sus, pentru care copiile mesajelor (eng.: dumps) sunt ușor de citit.

Chiar dacă protocolul SMTP este bine definit, mai pot apărea câteva probleme. O problemă este legată de lungimea mesajelor. Unele implementări mai vechi nu pot să lucreze cu mesaje mai mari de 64KB. O altă problemă se referă la expirări de timp (timeout). Dacă acestea diferă pentru server și client, unul din ei poate renunța, în timp ce celălalt este încă ocupat, întrerupând conexiunea în mod neașteptat. În sfârșit, în unele situații, pot fi lansate schimburi infinite de mesaje. De exemplu, dacă mașina 1 păstrează lista de poștă A și mașina 2 lista de poștă B și fiecare listă conține o intrare pentru cealaltă, atunci orice mesaj trimis oricărui dintre cele două liste va genera o cantitate nesfârșită de trafic de e-mail.

Pentru a atinge câteva dintre aceste probleme, în RFC 2821 s-a definit protocolul SMTP extins (ESMTP). Clienții care doresc să-l utilizeze trebuie să trimită inițial un mesaj *EHLO* în loc de *HELO*. Dacă acesta este rejectat, atunci serverul este unul standard de tip SMTP și clientul va trebui să se comporte în modul obișnuit. Dacă *EHLO* este acceptat, înseamnă ca sunt permise noile comenzi și noii parametri.

7.2.5 Livrarea finală

Până acum, am presupus că toți utilizatorii lucrează pe mașini capabile să trimită și să primească e-mail. După cum am văzut, e-mail-ul este livrat prin stabilirea unei conexiuni TCP între expeditor și destinatar și apoi prin trimiterea e-mail-ului prin ea. Acest model a funcționat bine zeci de ani, atât timp cât toate calculatoarele din ARPANET (și mai târziu din Internet) erau, de fapt, conectate la rețea și gata să accepte conexiuni TCP.

Totuși, odată cu apariția celor care accesează Internet-ul folosind un modem cu care se conectează la ISP-ul lor, acest lucru nu mai ține. Problema este următoarea: Ce se întâmplă când Elinor vrea să-i trimită Carlynei un e-mail și Carolyn nu este conectată la rețea în acel moment? Elinor nu va putea să stabilească o conexiune TCP cu Carolyn și astfel, nu va putea utiliza protocolul SMTP.

O soluție este ca agentul de transfer de mesaje de pe o mașină ISP să accepte e-mail-ul pentru clienții săi și să-l stocheze în cutiile lor poștale pe o mașină a ISP-ului. Din moment ce acest agent poate fi conectat la rețea tot timpul, se poate trimite e-mail 24 de ore pe zi.

POP3

Din nefericire, această soluție dă naștere altei probleme: cum își ia utilizatorul e-mail-ul de la agentul de transfer de mesaje al ISP-ului? Soluția acestei probleme este crearea unui alt protocol care să permită agenților de transfer mesaje (aflați pe calculatoarele clienților) să contacteze agentul de transfer mesaje (de pe o mașină ISP) și să facă posibilă copierea e-mail-ului de la ISP la utilizator. Un astfel de protocol este **POP3 (Post Office Protocol Version 3- Protocol de poștă, versiunea 3)**, definit în RFC 1939.

Situația anterioară (când atât expeditorul cât și destinatarul aveau conexiune permanentă la Internet) este ilustrată în fig. 7-15(a). O situație în care expeditorul este efectiv conectat la rețea (online) dar destinatarul nu, este ilustrată în fig. 7-15(b).

POP3 începe când utilizatorul pornește programul cititor de poștă (mail reader). Acesta sună la ISP (în caz că nu există deja o conexiune) și stabilește o conexiune TCP cu agentul de transfer de mesaje, prin portul 110. Odată ce conexiunea a fost stabilită, protocolul POP3 trece succesiv prin următoarele trei stări:

1. Autorizare.

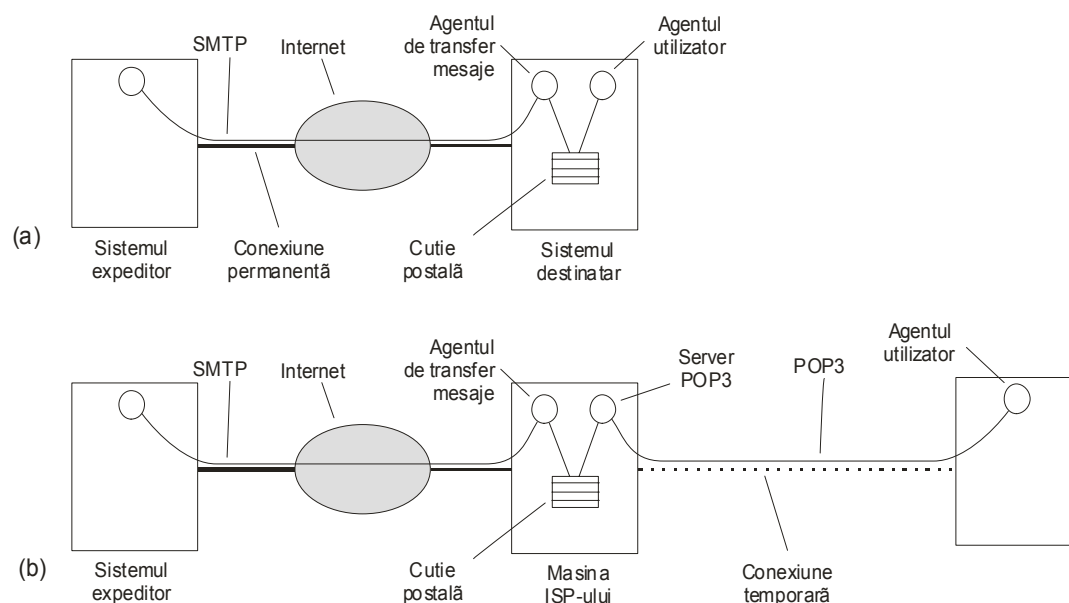


Fig. 7-15. (a) Trimiterea și citirea poștei când destinatarul are o conexiune permanentă la Internet, iar agentul utilizator rulează pe aceeași mașină ca și agentul de transfer de mesaje. (b) Citirea e-mail-ului când destinatarul are o conexiune comutată(dial-up) la un ISP.

2. Tranzacționare.
3. Actualizare.

Starea de autorizare se referă la admiterea utilizatorului în sistem (login). Starea de tranzacționare tratează colectarea e-mail-urilor și marcarea lor pentru ștergere din cutia poștală. Starea de actualizare se ocupă cu ștergerea efectivă a mesajelor.

Această comportare poate fi observată tastând ceva de genul:

```
telnet mail.isp.com 110
```

unde *mail.isp.com* reprezintă numele DNS al serverului de mail de la ISP. Telnet stabilește o conexiune TCP prin portul 110, pe care ascultă serverul POP3. După acceptarea conexiunii TCP, serverul trimite un mesaj ASCII anunțându-și prezența. De obicei, el începe cu +OK urmat de un comentariu. Un exemplu de scenariu este arătat în fig. 7-16 începând după ce conexiunea TCP a fost stabilită. Ca și mai înainte, liniile marcate cu C: sunt ale clientului (utilizatorului) iar cele cu S: sunt ale serverului (agentul de transfer de mesaje de la ISP).

În timpul stării de autorizare, clientul trimite numele său de utilizator și parola. După conectarea cu succes, clientul poate să trimită comanda *LIST*, care determină serverul să listeze conținutul cutiei poștale. Lista este terminată cu un punct.

Apoi, clientul poate regăsi mesajele folosind comanda *RETR* și le poate marca pentru ștergere cu *DELE*. Când toate mesajele au fost primite (și eventual marcate pentru ștergere), clientul trimite comanda *QUIT* pentru terminarea stării de tranzacționare și intrarea în starea de actualizare. Când serverul a șters toate mesajele, el trimite un răspuns și desființează conexiunea TCP.


```
S: +OK Serverul POP3 este pregătit
C: USER carolyn
S: +OK
C: PASS vegetables
S: +OK autentificare cu succes
C: LIST
S: 1 2505
S: 2 14302
S: 3 8122
S: .
C: RETR 1
S: (trimite mesajul 1)
C: DELE 1
C: RETR 2
S: (trimite mesajul 2)
C: DELE 2
C: RETR 3
S: (trimite mesajul 3)
C: DELE 3
C: QUIT
S: +OK Serverul POP3 întrerupe legătura
```

Fig. 7-16. Folosirea protocolului POP3 pentru a aduce trei mesaje.

Deși este adevărat că protocolul POP3 are abilitatea de a descărca un anumit mesaj sau un anumit grup de mesaje păstrându-le pe server, cele mai multe programe de e-mail descarcă tot și golesc cutia poștală. Ca urmare, practic singura copie rămâne înregistrată pe discul utilizatorului. Dacă acesta se strică, toate e-mail-urile pot fi pierdute definitiv.

Să recapitulăm pe scurt cum lucrează e-mail-ul pentru clienții unui ISP. Elinor creează un mesaj pentru Carolyn folosind un program de e-mail (adică, agentul utilizator) și face clic pe o icoană pentru a-l trimite. Programul de e-mail trimite mesajul agentului de transfer de mesaje de pe calculatorul Elinorei. Agentul de transfer de mesaje vede că mail-ul este pentru *carolyn@xyz.com* și folosește DNS pentru a căuta înregistrarea MX pentru *xyz.com* (unde *xyz.com* este ISP-ul Carolyni). Această cerere întoarce numele DNS al serverului de mail *xyz.com*. Agentul de transfer de mesaje caută acum adresa IP a acestei mașini folosind din nou DNS, de exemplu *gethostbyname*. Apoi, el stabilește o conexiune TCP cu serverul SMTP pe portul 25 de pe această mașină. Folosind o secvență de comenzi SMTP, asemănătoare celei din fig. 7-14, el transferă mesajul în cutia poștală a Carolyni și întrerupe conexiunea TCP.

După un timp, Carolyn își pornește PC-ul său, se conectează la ISP și pornește programul de e-mail. Programul de e-mail stabilește o conexiune TCP cu serverul POP3 pe portul 110 al serverului de poștă al ISP-ului. Numele DNS sau adresa IP a acestei mașini este configurată în mod normal atunci când programul de e-mail este instalat sau când este făcut contractul cu ISP-ul. După ce conexiunea TCP a fost stabilită, programul de e-mail al Carolyni lansează protocolul POP3 pentru a aduce conținutul cutiei sale poștale pe discul fix folosind comenzi ca cele din fig. 7-16. Odată ce a fost transferat tot e-mail-ul, conexiunea TCP este eliberată. De fapt, conexiunea cu ISP-ul poate fi desființată acum, din moment ce tot e-mailul este pe discul fix al Carolyni. Desigur, pentru a trimite un răspuns, va fi nevoie din nou de conexiunea cu ISP-ul, care, în general, nu este întreruptă imediat după aducerea poștei.

IMAP

Pentru un utilizator cu un singur cont de e-mail, la un singur ISP, care este tot timpul accesat de la un singur PC, POP3 este bun și larg folosit datorită simplității și robusteții sale. Totuși, există în industria calculatoarelor un adevăr bine înrădăcinat, acela că imediat ce un lucru funcționează bine, cineva va începe să ceară mai multe facilități (și să aibă mai multe probleme). Asta s-a întâmplat și cu e-mail-ul. De exemplu, multă lume are un singur cont de e-mail la serviciu sau la școală și vrea să-l acceseze de pe PC-ul de acasă, de pe calculatorul portabil în călătoriile de afaceri și din Internet café-uri în vacanțe. Cu toate că POP3 permite asta, din moment ce în mod normal el descarcă toate mesajele la fiecare conectare, rezultatul constă în răspândirea e-mail-ului utilizatorului pe mai multe mașini, mai mult sau mai puțin la întâmplare, unele dintre ele nefiind ale utilizatorului.

Acest dezavantaj a dat naștere unei alternative a protocolului de livrare finală, **IMAP (Internet Message Access Protocol** – Protocol pentru accesul mesajelor în Internet), care este definit în RFC 2060. Spre deosebire de POP3, care în mod normal presupune că utilizatorul își va goli căsuța poștală la fiecare conectare și va lucra deconectat de la rețea (off-line) după aceea, IMAP presupune că tot e-mail-ul va rămâne pe server oricât de mult, în mai multe căsuțe poștale. IMAP prevede mecanisme extinse pentru citirea mesajelor sau chiar a părților de mesaje, o facilitate folositoare când se utilizează un modem încet pentru citirea părții textuale a unui mesaj cu mai multe părți audio și video de mari dimensiuni. Întrucât premisa de folosire este că mesajele nu vor fi transferate pe calculatorul utilizatorului în vederea stocării permanente, IMAP asigură mecanisme pentru crearea, distrugerea și manipularea mai multor cutii poștale pe server. Astfel, un utilizator poate păstra o cutie poștală pentru fiecare corespondent și poate muta aici mesajele din inbox după ce acestea au fost citite.

IMAP are multe facilități, ca de exemplu posibilitatea de a se referi la un mesaj nu prin numărul de sosire, ca în fig. 7-8, ci utilizând atribute (de exemplu, dă-mi primul mesaj de la Bobbie). Spre deosebire de POP3, IMAP poate de asemenea să accepte atât expedierea mesajelor spre destinație cât și livrarea mesajelor venite.

Stilul general al protocolului IMAP este similar cu cel al POP3-ului, după cum se arată în fig. 7-16, cu excepția faptului că există zeci de comenzi. Serverul IMAP ascultă pe portul 143. În fig. 7-17 este prezentată o comparație între POP3 și IMAP. E bine de notat, totuși, că nu toate ISP-urile oferă ambele protocoale și că nu toate programele de e-mail le suportă pe amândouă. Așadar, atunci când alegeți un program de e-mail, este important să aflați ce protocoale suportă și să vă asigurați că ISP-ul oferă cel puțin unul din ele.

Caracteristica	POP3	IMAP
Unde este definit protocolul	RFC 1939	RFC 2060
Portul TCP folosit	110	143
Unde este stocat e-mail-ul	PC-ul utilizatorului	Server
Unde este citit e-mail-ul	Off-line	On-line
Timpul necesar conectării	Mic	Mare
Folosirea resurselor serverului	Minimă	Intensă
Mai multe cutii poștale	Nu	Da
Cine face copii de siguranță la cutiile poștale	Utilizatorul	ISP-ul
Bun pentru utilizatorii mobili	Nu	Da
Controlul utilizatorului asupra scrisorilor preluate	Mic	Mare
Descărcare parțială a mesajelor	Nu	Da
Volumul discului alocat (disk quota) este o problemă	Nu	Ar putea fi în timp
Simplu de implementat	Da	Nu
Suport răspândit	Da	În creștere

Fig. 7-17. O comparație între POP3 și IMAP.

Facilități de livrare

Indiferent dacă este folosit POP3 sau IMAP, multe sisteme oferă legături pentru procesarea adițională a mesajelor e-mail sosite. Un instrument deosebit de valoros pentru mulți utilizatori de e-mail este reprezentat de capacitatea de a construi filtre. Acestea sunt reguli care se verifică la sosirea mesajelor sau la pornirea agentului utilizator. Fiecare regulă specifică o condiție și o acțiune. De exemplu, o regulă ar putea spune că orice mesaj venit de la șef trebuie pus în cutia poștală numărul 1, orice mesaj de la un anumit grup de prieteni se duce în cutia poștală numărul 2 și orice alt mesaj conținând anumite cuvinte în Subiect este aruncat fără comentarii.

Unii ISP oferă filtre care clasifică automat mesajele sosite ca fiind importante sau nerelevante (spam) și memorează fiecare mesaj în cutia poștală corespunzătoare. Asemenea filtre funcționează verificând mai întâi dacă sursa este un autor cunoscut de mesaje „spam”. Apoi examinează subiectul. Dacă sute de utilizatori au primit un mesaj cu același subiect, probabil că el este nerelevant. Există și alte tehnici folosite pentru detecția mesajelor lipsite de importanță.

O altă caracteristică a livrării, pusă la dispoziție adesea, este posibilitatea de a retrimite (temporar) poșta la o adresă diferită. Această adresă poate fi și un calculator utilizat de un serviciu comercial de comunicații, care va contacta utilizatorul prin radio sau satelit, afișând *Subject:* linie pe pagerul său.

O altă trăsătură comună a livrării finale este abilitatea de a instala un **demon de vacanță**. Acesta este un program care examinează fiecare mesaj sosit și trimite o replică insipidă cum ar fi:

Salut. Sunt în vacanță. Mă întorc pe 24 august. O zi bună.

Asemenea răspunsuri pot să specifice, de asemenea, cum să fie tratate problemele urgente, alte persoane care pot fi contactate pentru probleme specifice etc. Majoritatea demonilor de vacanță păstrează urma celor cărora le-au trimis replici și se abțin de la a trimite unei aceleiași persoane o a doua asemenea replică. Demonii buni verifică și dacă mesajul sosit a fost trimis de la o listă de mail și în acest caz, nu mai răspund deloc. (Cei care trimit mesaje în timpul verii la liste mari de e-mail, probabil că nu doresc să primească sute de replici în care să le fie detaliate planurile de vacanță ale fiecăruia.)

Autorul s-a lovit recent de o formă extremă de prelucrare a livrării când a trimis o scrisoare unei persoane care pretinde că primește 600 de mesaje pe zi. Identitatea sa nu va fi deconspirată aici, ca nu cumva jumătate dintre cititorii acestei cărți să-i trimită și ei scrisori. Să-l numim în continuare John.

John și-a instalat un robot de e-mail care verifică fiecare mesaj sosit, ca să vadă dacă este de la un nou corespondent. Dacă este așa, trimite înapoi o replică standard în care explică faptul că nu mai poate să citească personal toate mesajele. În schimb a produs un document FAQ (Frequently Asked Questions) personal, unde răspunde la multe întrebări care i se pun de obicei. În mod normal, grupurile de știri și nu persoanele au documente FAQ.

Documentul FAQ al lui John dă adresa acestuia, numărul de fax și numerele de telefon și spune cum poate fi contactată firma sa. Arată cum poate fi chemat ca vorbitor și explică cum pot fi obținute lucrările sale și alte documente. Furnizează de asemenea referințe la programele scrise de el, o conferință pe care o organizează, un standard al cărui editor este și așa mai departe. E posibil ca această abordare să fie necesară, dar poate că un FAQ personal reprezintă simbolul final al statutului.

Poșta electronică pe Web (Webmail)

Un subiect care merită menționat este poșta electronică pe Web. Anumite situri de Web, cum ar fi Hotmail sau Yahoo oferă servicii de poșta electronică oricui dorește. Ele funcționează după cum

urmează. Au agenți normali de transfer de mesaje, care așteaptă la portul 25 conexiuni noi de SMTP. Pentru a contacta, să spunem Hotmail, trebuie să obținem înregistrarea sa DNS *MX*, de exemplu tastând

host —a —v hotmail.com

pe un sistem UNIX. Să presupunem că serverul de poștă electronică se numește *mx10.hotmail.com*; atunci tastând

telnet mx10.hotmail.com 25

se poate stabili o conexiune TCP prin care se pot trimite comenzi SMTP în modul obișnuit. Deocamdată, nimic special, cu excepția faptului că aceste servere mari sunt adeseori ocupate, ca atare se poate să dureze ceva mai mult până vă este acceptată o cerere de conexiune TCP.

Partea interesantă este cum se transmite poșta electronică. În principiu, atunci când utilizatorul se duce la pagina de Web a poștei electronice, îi este prezentat un formular în care i se cere un nume de cont și o parolă. Când utilizatorul face clic pe **Sign In**, numele de cont și parola sunt trimise serverului, care le validează. Dacă autentificarea s-a făcut cu succes, serverul găsește cutia poștală a utilizatorului și construiește o listă similară cu cea din fig. 7-8, cu diferența că are formatul unei pagini de Web în HTML. Pagina Web este transmisă apoi programului de navigare pentru a fi afișată. Pe multe din elementele paginii se pot executa clic-uri, astfel că mesajele pot fi citite, șterse, ș.a.m.d.

7.3 WORLD WIDE WEB

Web-ul este un context arhitectural pentru accesul la documente, răspândite pe mii de mașini din Internet, între care există legături. În 10 ani a evoluat de la o aplicație pentru transmiterea de date utile pentru fizica energiilor înalte la o aplicație despre care milioane de oameni cred că este Internetul. Popularitatea sa enormă se datorează faptului că are o interfață grafică plină de culoare, ușor de utilizat de către începători și în același timp oferă o cantitate imensă de informație - de la animale mitologice la tribul Zulu, pe aproape orice subiect posibil.

Web-ul (cunoscut și ca **WWW**) a apărut în 1989 la CERN, Centrul European de Cercetări Nucleare. CERN are câteva acceleratoare utilizate de echipe mari de cercetători din țările europene pentru cercetări în fizica particulelor. Deseori aceste echipe au membri din peste zece țări. Majoritatea experiențelor sunt foarte complicate și necesită ani de pregătire și construire de echipamente. Web-ul a apărut din necesitatea de a permite cercetătorilor răspândiți în lume să colaboreze utilizând colecții de rapoarte, planuri, desene, fotografii și alte tipuri de documente aflate într-o continuă modificare.

Propunerea inițială pentru crearea unei colecții de documente având legături între ele (Web) a fost făcută de fizicianul Tim Berners-Lee, fizician la CERN, în martie 1989. Primul prototip (bazat pe text) era operațional 18 luni mai târziu. În decembrie 1991, s-a făcut o demonstrație publică la conferința Hypertext'91 în San Antonio, Texas.

Aceasta demonstrație și publicitatea aferentă au atras atenția altor cercetători, fapt care l-a determinat pe Marc Andreessen de la University of Illinois să înceapă să dezvolte primul program de navigare grafic, Mosaic. Acesta a fost lansat în februarie 1993. Mosaic a fost atât de popular încât un an mai târziu Marc Andreessen a plecat pentru a forma o nouă companie, Netscape Communications Corp., care se ocupa cu dezvoltarea de software pentru Web. Când Netscape a devenit o com-