Akademska in raziskovalna mreža Slovenije
Academic and Research Network of Slovenia

ARNES, Jamova 39, 61000 Ljubljana, Slovenia
tel. + 386 61 125 15 15 fax + 386 61 125 54 54
arnes@arnes.si
http://www.arnes.si/

UPORABA STORITEV OMREŽJA INTERNET NA RAČUNALNIKU STENAR

PRIROČNIK

Ljubljana, April 1995

V PostScript zapisu je preko *anonymous ftp* dostopen na strežniku *ftp.arnes.si*. Tiskano verzijo je možno naročiti v ARNESu.

Po mnenju Ministrstva za znanost in tehnologijo št. 415-02-35/95, z dne 7. 4. 1995, sodi publikacija med proizvode, za katere se plačuje 5% prometni davek.

Kazalo

1.

Uvod

Storitve globalnega računalniškega omrežja imenovanega Internet dobivajo vedno pomembnejšo vlogo v izobraževalnih, razvojnih in raziskovalnih procesih povsod po svetu. Povezava računalnikov omogoča elektronsko pošto, elektronske konference in dostop do tisočev podatkovnih baz po vsem svetu. V svetovni Internet je povezano več milijonov računalnikov in storitve, ki jih nudi, uporablja nekaj deset milijonov uporabnikov.

Javni zavod ARNES načrtuje, organizira in upravlja akademsko in raziskovalno omrežje Slovenije, ki je del svetovnega Interneta. Organizacije s področja raziskovanja, razvoja, izobraževanja in kulture lahko zaprosijo za vključitev svojih računalnikov oziroma svojih lokalnih omrežij v ta sistem. Na ta način so povezane z drugimi organizacijami, dostopajo lahko do različnih podatkovnih baz in uporabljajo druge Internet storitve.

Vključitev v akademsko in raziskovalno mrežo Slovenije in s tem v svetovni Internet se lahko izvede na različne tehnične načine.

Najpreprostejši in najcenejši način povezave je posredni dostop preko kakšnega računalnika, ki je že vključen v Internet. To pomeni, da je najprej potrebno na takem računalniku dobiti dovoljenje za delo in uporabo Internet storitev. Nato je mogoče do tega računalnika dostopati preko telefonskega omrežja od doma ali z delovnega mesta, osebni računalnik pa pri tem služi le kot terminal. ARNES vzdržuje računalnik z imenom *Stenar*, ki omogoča tak dostop do Interneta. Na ta način lahko uporabniki preprosto in poceni pridejo do Internet storitev, vendar je rešitev primerna le za posameznike in ne omogoča multimedijskega dostopa do podatkov v Internetu.

Poceni vendar zahtevnejši način povezave je vključitev lastnega računalnika ali lastnega lokalnega omrežja preko telefonskega omrežja direktno v Internet. Ta rešitev zahteva več programske opreme na lastnem računalniku (oziroma omrežju), zahteva boljše vzdrževanje in zaščito, omogoča pa poleg tekstovnega tudi multimedijski dostop do podatkov v Internetu. Ker uporablja za prenos telefonsko omrežje, ki ne zmore hitrega prenosa podatkov, je primeren le za manjše organizacije.

Dražja in zmogljivejša povezava lokanega omrežja in vseh računalnikov na tem lokalnem omrežju v akademsko in raziskovalno omrežje Slovenije pa je mogoča preko posebnega najetega voda. Ta povezava omogoča vse storitve in hiter prenos podatkov, zahteva pa dodatno opremo in dodatno vzdrževanje lokalnega omrežja.

Ta priročnik je napisan za tiste uporabnike akademske in raziskovalne mreže Slovenije, ki uporabljajo Internet storitve preko računalnika *Stenar*.

1.1 Povezava na *Stenar* in Internet storitve

Raziskovalne organizacije, inštituti, razvojni oddelki v industriji, izobraževalne in kulturne inštitucije, knjižnice ter samostojni raziskovalci lahko dobijo dostop v Internet preko računalnika *Stenar*. ARNES dodeljuje uporabniška imena enemu ali več posameznikom v teh organizacijah. Organizacija mora v ta namen izpolniti posebno prijavnico, posamezniki, ki dobijo geslo, pa še izjavo o načinu uporabe Internet storitev. ARNES nato pisno pošlje vsakemu posamezniku dodeljeno uporabniško ime, geslo in njegov bodoči elektronski naslov. Vsako uporabniško ime ima lahko le en elektronski poštni predal.

ARNES si glede na razpoložljivo opremo in zmogljivosti pridržuje pravico omejevati število uporabnikov iz posamezne organizacije.[1] Prav tako lahko ARNES ukine uporabniško ime, če pride do kršitev pravil uporabe.

Posameznik, ki ima uporabniško ime na *Stenarju* in želi na njem uporabljati Internet storitve, potrebuje osebni računalnik, posebno programsko opremo na tem računalniku, modem in telefonski priključek.

Osebni računalnik je lahko kakršen koli. Nanj je potrebno instalirati komunikacijsko programsko opremo, ki ga spremeni v terminal računalnika *Stenar*. Drugo poglavje tega priročnika podrobneje opisuje tovrstne programe. V ARNESu je mogoče naročiti disketo s programom *WinQVT* in ustrezni priročnik, ki je primeren za tiste uporabnike, ki imajo osebni računalnik vrste IBM-PC z okoljem Windows. Obstaja pa še mnogo drugih primernih programov za take in drugačne osebne računalnike.

Za delo preko telefonskega omrežja je potreben modem. Drugo poglavje tega priročnika podrobneje opisuje modeme, njihove lastnosti in uporabo.

Povezava do *Stenarja* se vzpostavi tako, da uporabnik s pomočjo modema pokliče eno od ARNESovih vstopnih telefonskih številk: (061) 9744, (062) 9744, (063) 9744, (064) 9744, (065) 9744, (066) 9744, (068) 9744 ali (069) 9744. Ob vzpostavitvi telefonske povezave se na ekranu pokaže menu, kjer je potrebno izbrati povezavo na *Stenar*, nato pa na zahtevo vtipkati uporabniško ime in geslo. S tem je odprta terminalska povezava na računalnik *Stenar*.

[1] V primeru, da želi veliko sodelavcev določene organizacije uporabljati Internet storitve, je bolje, da se ta organizacija poveže v akademsko in raziskovalno omrežje Slovenije z najetim vodom.

Operacijski sistem računalniku *Stenar* je VMS.[2] Pri uporabi Internet storitev ni nujno obvladati veliko podrobnosti o tem operacijskem sistemu, vseeno pa je potrebno poznati nekaj osnov. Tretje poglavje priročnika tako opisuje predvsem najosnovnejše ukaze za delo z datotekami. Opisan je tudi urejevalnik besedil *EDT*, ki pride v poštev, kadar uporabnik sestavlja na *Stenarju* elektronska sporočila ali se pogovarja v elektronskih konferencah. Prav tako je v tem poglavju razložena uporaba elektronske pošte.

Uporaba ostalih Internet storitev na računalniku *Stenar* je opisana v četrtem poglavju priročnika. Opisane so osnovne lastnosti najpomembnejših Internet storitev in način, kako jih na *Stenarju* pokličemo. Podrobnejše lastnosti posameznih storitev pa so opisane v katerikoli boljši knjigi[3], ki opisuje Internet.

Uporaba Internet storitev na *Stenarju* je brezplačna, uporabnik mora plačati le impulze za lokalno telefonsko povezavo (Telekomu Slovenije) do klicane ARNESove vstopne točke.

Uporabniki, ki pri delu na omrežju naletijo na težave, lahko pošljejo elektronsko sporočilo na naslov *helpdesk@arnes.si*. Če bo problem jasno in temeljito opisan, vam bomo v ARNESu lahko pomagali z odgovorom. V primeru, da ne morete do računalnika *Stenar* oziroma da ne morete poslati elektronskega sporočila, nam pošljite opis problema s faksom na številko (061) 125 54 54 ali pa telefonirajte na številko (061) 125 44 55.

[2] Tudi VAX/VMS ali OpenVMS.

[3] npr. Ed Krol, *The Whole Internet*, O'Reilly & Associates, Inc., 1994

1.2 Pravila obnašanja v Internetu

S tem, ko dobi uporabnik možnost dela na *Stenarju*, dobi tudi dostop do svetovnega računalniškega omrežja Internet, preko tega pa tudi možnost komunikacije z drugimi računalniškimi omrežji, ki tvorijo globalno komunikacijsko strukturo med seboj povezanih računalnikov.

Kaj pomeni vstop v ta svet in kakšen medij je pravzaprav Internet? Predstavljajmo si, da vstopimo v ogromno, nekoliko razmetano knjižnico, ki ji ne vidimo konca. V njej se gnete nekaj milijonov ljudi. V knjižnici so poleg kupov knjig, slik, in videotrakov tudi skupine ljudi, ki knjižnico uporabljajo kot klubski prostor.

Ko vstopimo v ta svet, se je treba zavedati, da si ga delimo z velikim številom ljudi, ki lahko izhajajo iz zelo različnih okolij, da se ta prostor razteza preko držav z različnimi zakoni in da v njem veljajo določena, mnogokrat nenapisana pravila. Uporabniki se morajo podrejati tem pravilom oziroma tako imenovani *etiki omrežja*, pa tudi slovenskim in mednarodnim pravnim normam.

Pri opisu posameznih storitev bomo o pravilih obnašanja povedali več, tu naštejmo le najpomembnejše:

Nekomercialna uporaba
: ARNES nudi Internet storitve le določenu krogu organizacij v Sloveniji in to le za njihovo raziskovalno, razvojno in izobraževalno delo. Posojanje uporabniških imen drugim organizacijam in posameznikom ter uporaba storitev v komercialne namene tako ni dovoljena.

Spoštovanje avtorskih pravic
: Uporabniki ne smejo računalnika *Stenar* ali omrežja uporabljati za prenos in kopiranje nelegalnih kopij programov in druge zaščitene intelektualne lastnine.

Spoštovanje pravic drugih uporabnikov omrežja
: Uporabniki ne smejo namenoma nadlegovati ali motiti dela drugih uporabnikov omrežja ali namenoma zlorabljati zmogljivosti sistema ali omrežja tako, da s tem ovirajo delo drugih uporabnikov.

Spoštovanje tajnosti podatkov
: Uporabniki morajo spoštovati tajnost podatkov, ki so v lasti drugih uporabnikov omrežja.

Uporaba drugih računalnikov le z dovoljenjem
: Uporabniki lahko uporabljajo druge računalnike le s povprejšnjim dovoljenjem lastnika ali skrbnika tega računalnika, razen v primeru javno dostopnih storitev – tudi v tem primeru pa se morajo držati objavljenih pravil in postopkov.

Varovanje gesla

Najpogosteje nastanejo zlorabe pri uporabi računalniškega omrežja zaradi slabega varovanja gesel. Računalniku se mora vsak uporabnik identificirati preko uporabniškega imena in gesla. Nekatera uporabniška imena so lahko skupinska, večinoma pa se glasi uporabniško ime na eno samo osebo. Ta oseba je tudi odgovorna za vse delo na računalniku, ki se izvaja pod njenim uporabniškim imenom. Zato mora geslo ostati tajno, kar dosežemo z upoštevanjem nekaterih pravil:

- Gesla ne zapisujemo na papir, ki nato leži na mizi zraven terminala ali računalnika, kjer ga lahko vsakdo prebere (če si ga res ne moremo zapomniti, pa je najbrž dovolj varno, če ga nosimo v denarnici). Geslo občasno spreminjamo.
- Gesla nikomur ne "posojamo".
- Izberemo geslo, ki ga je težko uganiti (zlasti se izogibajmo očitnim geslom: lastno ime ali priimek oz. ime sorodnika, ime organizacije...).

Čemu taka skrivnostnost? V vaši okolici se lahko najde kdo, ki bo uporabljal vaše uporabniško ime za delo na omrežju na način, ki morda odstopa od uveljavljenih pravil obnašanja. S tem, ko bo uporabljal vaše uporabniško ime, pa bo lažje zakril svojo pravo identiteto (morebitne posledice dejanj, ki so bila storjena z vašega uporabniškega imena, pa nosite vi). Geslo je najbolje primerjati s ključem vhodnih vrat, tega pa ponavadi ne delimo naokoli.

1.3 Uporabljene oznake

V priročniku bomo uporabljali naslednje dogovore pri pisavi:

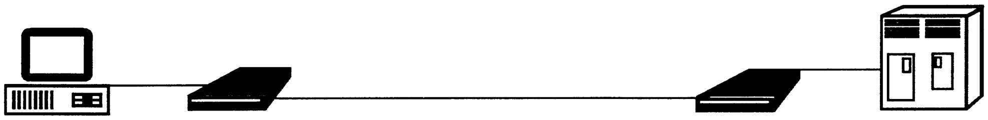
- Ukazi so v besedilu in v primerih natisnjeni debelo, npr. **exit**.
- Besedilo, ki ga izpiše računalnik v odgovor, ter izbire v menujih so natisnjeni `takole`.
- Imena (datotek, imenikov, računalnikov, nekaterih programov, ...) ter parametri ukazov so običajno natisnjeni *poševno*.
- Poudarjeni izrazi v besedilu so natisnjeni *poševno*.
- Oznaka [Ctrl–Z] pomeni, da držimo tipko [Ctrl] pritisnjeno in pritisnemo še tipko [Z].
- Oznaka [PF1] [7] pomeni, da pritisnemo najprej prvo tipko, potem pa še drugo.

Ostale posebnosti pisave bomo pojasnili sproti ali pa bodo razvidne iz besedila.

2.

Dostop do oddaljenega računalnika prek telefonskega omrežja

Računalniki so običajno povezani med seboj prek posebnega podatkovnega omrežja. Če velika hitrost in zanesljivost povezave nista nujno potrebni, lahko računalnike povežemo prek navadnega telefonskega omrežja. Takšno rešitev uporabimo običajno v primeru, ko želimo povezati osebni računalnik z nekim oddaljenim računalnikom. Za tako povezavo sta potrebna dva modema in program za komunikacijo na osebnem računalniku. V tem razdelku bomo zato najprej govorili o modemih, nato pa o programih, ki osebni računalnik spremenijo v terminal oddaljenega računalnika in s tem omogočajo komunikacijo prek telefonskega omrežja.



Slika 2.1: Modemska povezava osebnega računalnika z oddaljenim računalnikom.

2.1 Modemi

Modem (MOdulator/DEModulator) je namenjen pretvorbi digitalnih (računalniških signalov) v analogne signale, najpogosteje za prenos prek telefonskega omrežja. Modem je nujno potreben pri dostopanju do drugih računalnikov prek običajnega telefonskega omrežja. V tem poglavju bomo najprej opisali lastnosti različnih tipov modemov, nato pa razložili, kako jih priključimo na računalnik in v telefonsko omrežje.

2.1.1 Lastnosti modemov in njihove oznake

Delovanje modemov je standardizirano. To pomeni, da obstajajo standardi, ki določajo različne lastnosti modemov. Smiselno je, da se pri izbiri modema ravnamo predvsem po oznakah standardov, ki jih posamezni modem podpira[1].

Morda ni odveč pripomniti, da lahko med seboj komunicirajo modemi različnih znamk. Pri izbiri oz. uporabi modema nam torej ni treba skrbeti, katere znamke je modem "na drugi strani".

V nadaljevanju bomo opisali najvažnejše standarde za hitrost, odkrivanje in odpravljanje napak in zgoščanje podatkov.

Hitrost

Standardi za hitrost določajo *najvišjo možno* hitrost prenosa, s katero si modema lahko izmenjujeta podatke prek telefonskega omrežja. Dejanska hitrost je včasih nižja, saj je odvisna tudi od kvalitete ustreznega dela telefonskega omrežja. Poznamo več standardov, ki so v naslednji tabeli razvrščeni po hitrosti:

V.22	Modem, ki zadošča temu standardu, je danes že zastarel, saj zmore le 1200 bit/s, to pa je prepočasno za udobno delo.
V.22bis	Modem, ki zadošča temu standardu, se še marsikje uporablja, čeprav je s 2400 bit/s za večino uporabnikov že prepočasen.
V.32	Hitrejši sistem, ki omogoča hitrost 9600 bit/s. Je zelo zanesljiv in pogosto deluje tudi prek telefonskega omrežja slabše kvalitete.
V.32bis	Nadgradnja prejšnjega sistema. V.32bis je s 14400 bit/s precej hitrejši od V.32. Večjo hitrost pa je mogoče izkoristiti le prek dobrega telefonskega omrežja, sicer pa dela na hitrosti 9600 bit/s.
V.32terbo	To je nestandardna razširitev sistema V.32bis. Nakup odsvetujemo.
V.34	To je nov standard. Modem, ki zadošča temu standardu, je najhitrejši in zmore hitrost 28.800 bit/s. Delovna verzija tega standarda se je imenovala *V.fast*. Nakup modema z oznako *V.fast* odsvetujemo.

[1] Pri nakupu lahko torej zanemarimo druge tehnične podatke, npr. bite na sekundo, baude itd.

Če komunicirata modema različnih standardov, se načeloma povežeta na najvišji skupni hitrosti, ki jo še oba podpirata. Modema V.32bis in V.32 se na primer povežeta po standardu V.32, V.22bis in V.32bis pa se povežeta po standardu V.22bis. Novejši standardi namreč vključujejo tudi podporo pomembnejših starejših standardov in tako npr. modem, ki podpira V.32bis, ustreza tudi standardom V.32, V.22bis in V.22. V primeru slabših razmer modema avtomatsko preideta na nižjo hitrost.

Odkrivanje in odpravljanje napak

To je postopek, ki omogoča, da modema na obeh koncih povezave zaznata in sama popravita morebitne napake, ki nastanejo zaradi motenj na liniji. Sistem deluje zelo zanesljivo, vendar se včasih le zgodi, da vidimo "smeti" oz. napake na liniji. Uveljavljena sta predvsem dva standarda:

MNP 4	Starejši standard, vendar deluje zelo zanesljivo, zato je danes običajno del vsakega modema.
V.42	Novejši standard za avtomatsko korekcijo napak pri prenosu. Malce slabše se obnese na slabih linijah, v takem primeru modem preklopimo na MNP4.

Zgoščanje podatkov

Večino podatkov, ki jih želimo s pomočjo modemov pretočiti prek telefonskega omrežja, je mogoče precej zgostiti, tako da zavzemajo manj prostora in zato tudi prenos traja manj časa. To velja predvsem za navadna besedila. Tu srečamo naslednji oznaki:

MNP 5	Starejši standard, ki je danes v modemih predvsem zaradi združljivosti s starejšimi modeli. Omogoča do dvakratno zgoščanje.
V.42 bis	Novejši standard, ki se prilagaja podatkom, to pomeni, da uporablja zgoščanje podatkov, kadar je to mogoče. Omogoča do štirikratno zgoščanje. Najlepše se obnese pri interaktivnem delu na oddaljenem računalniku, vendar je na splošno tako "pameten", da je lahko stalno vključen.

2.1.2 Zunanji ali notranji modem?

Običajno dobimo modeme, ki ustrezajo gornjim standardom, v dveh različicah: zunanjega ali notranjega. *Zunanji modem* je skorajda samostojna enota, ki ima svoje ohišje in napajanje in se na računalnik priključi prek posebnega kabla s priključkom RS/EIA232 (ustrezni kabel zahtevajte takoj ob nakupu!). Prednost zunanjega modema je prenosnost, saj ni del računalnika in ga zlahka prenašamo okrog. Povrhu tega nam v težavah ni treba resetirati celega računalnika, ampak samo modem.[2] *Notranji modem* je narejen v obliki kartice, ki jo vgradimo v računalnik. Tak modem je cenejši (od nekaj deset do nekaj sto DEM), saj ne potrebuje ohišja in samostojnega napajanja.

2.1.3 Izbira modema

Cena modema ni premo sorazmerna s hitrostjo, ki jo modem zmore. Tako lahko za dvakrat višjo ceno dobimo nekajkrat hitrejši modem. Danes je smiselno kupiti le modem, ki zmore vsaj 9600 bitov na sekundo (V.32 ali več). Pri tem najcenejši model ni vedno najboljša izbira, saj imamo lahko potem pri delu precej več problemov.

Če je le možno, pa priporočamo nakup hitrejšega modema, to je V.32bis (14400 bitov/s) ali še bolje V.34 (28800 bitov/s), saj se začetna investicija hitro povrne zaradi manjših telefonskih računov, večja hitrost pa omogoča tudi uporabo sodobnejših storitev.

Vsi novejši hitri modemi imajo običajno že vgrajen dodatek za pošiljanje in sprejemanje telefaksov.

Velja pa, da je dejanska hitrost komunikacije zelo odvisna od kvalitete telefonskega omrežja ter nastavitve modema in računalnika, in ne samo od tehničnih lastnosti modema, zato se moramo pri nakupu zavedati tudi tega.

Na tržišču srečamo modeme različnih proizvajalcev. Kot pri nakupu avtomobilov velja, da so cene odvisne od količine raznih dodatkov[3], zanesljivosti in seveda blagovne znamke. Med uveljavljenimi proizvajalci z v povprečju nekoliko višjimi cenami modelov pri nas srečamo npr. Motorolo, Nokio in Racal, popularni znamki sta tudi ZyXel in US Robotics, ki navadno ponujata ugodnejše cene. Poleg teh so seveda še druge znamke, nekatere med njimi so bolj, nekatere pa manj uveljavljene. Računalniške revije občasno objavljajo analize zmogljivosti in cen, ki nam lahko pomagajo pri nakupu.

Cenovni razpon za modeme solidne kvalitete je trenutno (pomlad 1995) od 250 do 1500 DEM za standard V.32 bis oziroma 500 do 2500 DEM za modeme po standardu V.34.

Ko kupujemo modem, se je treba zavedati, da se cene nenehno spreminjajo (v glavnem padajo), pojavljajo se tudi novi modeli.

Pred nakupom zato preverite trenutno stanje na tržišču (spremljajte analize v revijah), posvetujte se z nekom, ki že ima izkušnje na tem področju, in preverite, kako modem deluje. Pri modemih bodite pozorni tudi na ateste, ki nam zagotavljajo, da so primerni za naše okolje in prihranijo skrbi ob morebitnem pregledu inšpekcije. Pri nakupu preverite,

[2] Zelo pomembno, če uporabljamo sisteme, ki morajo stalno teči, npr. Unix.

[3] Mnogi od teh dodatkov — podpora za najete linije, sinhrona kompresija — niso zanimivi za "domačo uporabo".

ali je zagotovljen servis in kakšna je garancija.

2.1.4 Priključitev modema

Če morate modem priključiti sami, najprej preberite navodila v priročniku, ki ste ga dobili z modemom. Pred začetkom priključevanja ugasite računalnik.

Priključitev zunanjega modema

Priključitev zunanjega modema je razmeroma preprosta. Na računalniku poiščemo priključek RS/EIA232, ki je običajno malce širši (25 žil), najdemo pa ga tudi v ožji izvedbi (npr. priključek za miško, ki ima običajno le 9 žil). S kablom povežemo računalnik in modem. Nato odklopimo telefon s telefonskega omrežja in namesto njega priključimo modem. Če sta na modemu dve vtičnici, naj bo v telefonsko omrežje povezana tista, ki ima napis LINE. Drugo vtičnico v tem primeru povežemo s kratkim telefonskim kablom, ki ga običajno dobimo z modemom, na telefonski aparat. Tako lahko uporabljamo telefon, kadar ne prenašamo podatkov prek modema, ne da bi znova prevezovali kable. Sedaj priključimo le še napajalnik, ki ga dobimo poleg modema, ter vključimo modem in računalnik. Modem je tako pripravljen na delo; s pomočjo priloženih navodil lahko preverimo, ali na njem gorijo ustrezne lučke.

Vgradnja notranjega modema

Vgradnja notranjega modema je bolj zapletena. Notranji modem priključimo neposredno na matično ploščo računalnika, zato moramo uporabiti izvijač in računalnik odpreti.[4] Pred tem obvezno ugasimo računalnik! Na matični plošči je običajno vedno še kako prosto mesto (reža) za kartice, kamor lahko namestimo kartico z notranjim modemom. Kartico vtaknemo v podnožje za to namenjene reže, tako da imamo navzven obrnjeno stran s priključki za telefonski kabel, ter privijemo nazaj vse vijake in zapremo računalnik. Ker se notranji modem priključuje neposredno na računalnik, ne potrebujemo dodatnega napajalnika in kabla.

[4] Nekateri prodajalci računalniške opreme ne dovolijo odpiranja računalnikov, ki so še v garanciji, zato se v tem primeru posvetujte z njimi.

2.2 Programi za komunikacijo

Če želimo delati na oddaljenem računalniku, potrebujemo poleg modema še program za osebni računalnik (PC[5]), ki skrbi za uspešno komunikacijo. Glavne naloge tega programa so:

Upravljanje modema: S pomočjo tega programa lahko modemu pošiljamo ukaze in po potrebi spreminjamo nastavitve.

Posnemanje terminalov: Pri delu z oddaljenim računalnikom se mora PC obnašati tako, da bo oddaljenemu računalniku predstavljal znani tip terminala.

Prenos datotek: Omogoča prenos datotek med osebnim in oddaljenim računalnikom prek vzpostavljene klicne povezave.

Preden program za komunikacijo začnemo rutinsko uporabljati, opravimo začetne nastavitve, ki določajo način njegovega delovanja. Te nastavitve potem ostanejo v veljavi ob vsaki uporabi komunikacijskega programa, dokler jih znova ne spremenimo.

2.2.1 Začetne nastavitve komunikacijskega programa

Ko je modem priključen, moramo program za komunikacijo ustrezno nastaviti. Določiti moramo priključek, njegovo hitrost, obliko podatkov in tip posnemanja funkcij terminala.

Določitev priključka pomeni, da moramo programu povedati, prek katerih vrat je modem priključen na PC. To običajno izvedemo s pomočjo menija `SETUP` ali `COMM`. Če imamo zunanji modem, je ta običajno na priključku COM1 ali COM2; pri notranjih pa na COM3 ali COM4. Ali smo izbrali pravi priključek, se lahko prepričamo tako, da poskusimo poslati na modem kak ukaz, npr. **at** [Enter]. Če je vse v redu, nam modem odgovori z `OK`, sicer poskusimo z drugim priključkom.

Ko smo določili priključek, nastavimo hitrost povezave med našim računalnikom in modemom. Smiselno je nastaviti to hitrost za eno stopnjo višje, kot je hitrost, ki jo modem zmore prek telefonskega omrežja. Če bomo uporabljali zgoščanje podatkov, nastavimo hitrost serijskega priključka na še višjo vrednost. Primer: če imamo modem po standardu V.22bis (2400 bit/s), nastavimo hitrost priključka na 4800 bit/s, če pa bomo uporabljali zgoščanje podatkov po standardih MNP5 ali V.42bis, nastavimo hitrost na 9600 bit/s.

Običajno nastavimo obliko podatkov na način `8N1` (8 bitov za znak, brez paritete in 1 stop bit). Zelo pogosto je ta način že vnaprej nastavljen in ga ni treba spreminjati.

[5]Vse specifične navedbe v tem poglavju se nanašajo na IBM PC združljive osebne računalnike z operacijskim sistemom MS DOS ali MS Windows.

Nazadnje moramo nastaviti še ustrezen tip emulacije. Nastavitev ustrezne emulacije (posnemanja funkcij) pomeni, da priredimo delovanje zaslona in tipkovnice na našem osebnem računalniku zahtevam, ki jih pričakuje oddaljeni računalnik.[6] Program za komunikacijo nam omogoča, da se naš računalnik vede kot terminal izbranega tipa. Običajno je smiselno, da izberemo emulacijo terminala serije VT. V večini primerov zadošča že nastavitev tipa VT100, če pa naš program to omogoča, izberemo emulacijo višjega tipa v tej seriji.[7]

Izbrane nastavitve s posebnim ukazom (ponavadi **Save**) shranimo, da bodo veljale tudi ob naslednji uporabi programa.

2.2.2 Upravljanje modema

Prek programa za komunikacijo modemu ukazujemo, naj vzpostavi zvezo, jo prekine, nam dà informacije o vzpostavljeni zvezi, z njim lahko spreminjamo nastavitve za odpravljanje napak oziroma zgoščanje podatkov.

Dandanes vsi modemi podpirajo tako imenovane Hayesove ukaze, ki jih je prvi uporabil proizvajalec modemov Hayes. Večina Hayesovih ukazov je standardiziranih, vendar ne vsi. To pomeni, da ne smemo slepo tipkati ukazov, ki nam jih svetujejo uporabniki drugih modemov, saj imajo lahko ti ukazi na vašem modemu popolnoma drug pomen, kakor na modemu drugega tipa in proizvajalca. Zato si moramo pri delu z modemom pomagati z originalnim priročnikom.

Če nismo spreminjali nastavitev, nam modem vedno vrne informacijo o izvedbi zadnjega ukaza. Najpogosteje dobimo odziv

`OK` za uspešno izvedene ukaze, oziroma
`ERR` ali `ERROR` za neobstoječe ukaze ali opcije.

Če odgovora ne dobimo, je verjetno prišlo do napake, ali pa smo pred tem z ukazno opcijo izključili potrjevanje izvedbe ukazov.

Osnovne ukaze si oglejmo na primeru dostopa do ARNESovega računalnika *Stenar*.

Vedno je koristno začeti z ukazom **atz**, ki postavi modem v začetno stanje.

Nadaljujemo z ukazom **atx3**, ki povzroči, da modem ne posluša signalov na liniji. Tako modemu preprečimo, da bi pred klicem preveril, ali je linija prosta (večina modemov žal ne prepozna ustreznega signala na slovenskem telefonskem omrežju).

[6] Oddaljeni računalnik je verjetno večuporabniški sistem in z njim uporabniki komunicirajo prek terminalov. Terminal sestavljata tipkovnica in ekran in je namenjen samo vnosu in prikazu podatkov. Računalniški sistemi imajo različne tipe terminalov, za večino sistemov pa so primerni terminali serije VT (npr. VT100, VT220, VT320, VT420).

[7] Dostop do knjižničnega sistema IZUM/COBISS zahteva npr. tip VT220 ali višjega.

Z ukazom **atdp***številka* povemo modemu, da želimo poklicati neko številko.[8] Pri telefonih Z ukazom **atdp1315151** na primer povemo modemu, naj pokliče številko 131-51-51 in vzpostavi zvezo.

V času klicanja ne smemo pritiskati ničesar, saj bi to pomenilo, da hočemo prekiniti klic! Če se zveza vzpostavi, zaslišimo piskajoč zvok, ki pove, da modema komunicirata. Po nekaj sekundah se javi oddaljeni računalnik in nas povpraša po imenu. Če je linija zasedena, se izpiše obvestilo `BUSY`, v primeru težav na zvezi ali naše prekinitve s pritiskom na katerokoli tipko pa dobimo obvestilo `NO CARRIER`. V takem primeru čez čas znova poskusimo z ukazom **atdp***številka*.

Primer uspešnega dialoga:

atz
`OK`
atx3
`OK`
atdp1315151
`CONNECT 19200`

2.2.3 Posnemanje terminalov

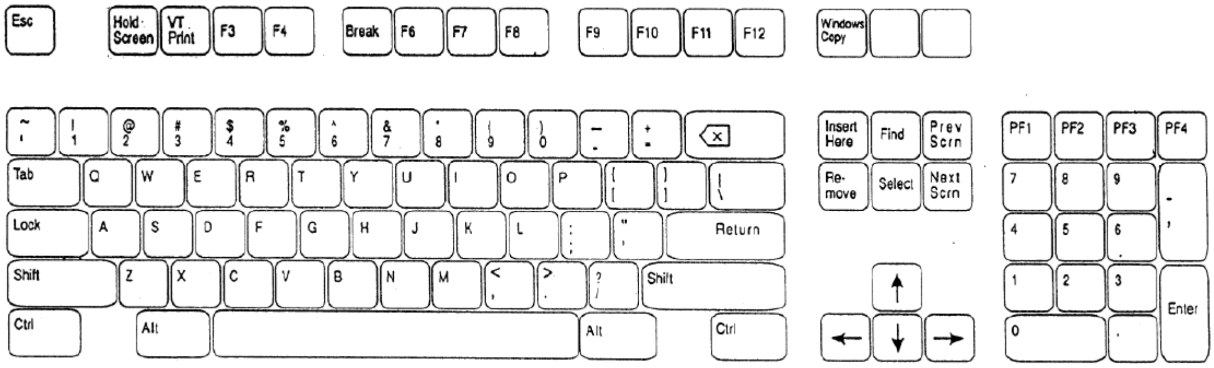
Vsak terminal opravlja dve glavni funkciji, to je vnos podatkov s tipkovnico in izpis podatkov na zaslon. Kot smo že omenili, večina računalniških sistemov podpira serijo terminalov VT, zato je običajno smiselno v programu za komunikacijo izbrati tako nastavitev, ki nam naš osebni računalnik "spremeni" v enega od terminalov serije VT.

Ko je program za komunikacijo vključen, bo naš PC prikazoval podatke z oddaljenega računalnika na način, kot ga določa standard terminala (običajni znakovni način z omejeno znakovno grafiko), kar izključuje prikaz bogatejše grafične informacije, kakršno na primer dobimo z okni.

Ko je program za komunikacijo vključen, je tipkovnica našega PCja v neki meri podobna tipkovnici izbranega tipa terminala.

Slika 2.2 kaže razpored tipk na tipkovnici običajnega osebnega računalnika. Dober komunikacijski program bo posnemal terminal tipa VT tako, kot je to označeno na posameznih tipkah. Če primerjamo funkcije, ki jih tipke običajno opravljajo, s funkcijami, ki jih opravljajo pri posnemanju terminala, vidimo, da ostane osrednji del tipkovnice (črke, številke, $\boxed{\text{Ctrl}}$, $\boxed{\text{Return}}$) nespremenjen. Drug pomen pa dobijo tako imenovane funkcijske tipke. Za delo na operacijskem sistemu VMS so na primer pomembne tipke $\boxed{\text{PF1}}$, $\boxed{\text{PF2}}$, $\boxed{\text{PF3}}$,

[8] Velja za pulzno izbiranje številk (pri izbiranju številke 4 npr. slišimo "tk-tk-tk-tk"). Če imamo telefon (in centralo) s tonskim izbiranjem (slišimo ton za vsako številko), je namesto **atdp** ustrezen ukaz **atdt**.



Slika 2.2: Na sliki je razpored tipk, kot ga običajno najdemo na tipkovnici osebnega računalnika. Na tipkah pa so vpisane funkcije terminala VT420, ki jih te tipke izvajajo, če imamo instaliran dober program za komunikacijo.

PF4, ki jih pri emulaciji najdemo v zgornji vrsti stranske tipkovnice. Samostojno stoječa skupina tipk za premikanje (puščice) obdrži tudi pri emulaciji svojo funkcijo.

Obstajajo tudi komunikacijski programi, ki pri posnemanju terminala drugače razporedijo funkcije tipk po tipkovnici.

2.2.4 Prenos datotek

Kadar delamo na oddaljenem (gostiteljskem) računalniku prek terminalske povezave z modemom, se moramo ves čas zavedati, da se datoteke, s katerimi delamo, nahajajo na gostiteljskem računalniku, ne pa na našem (osebnem) računalniku. Če je gostiteljski računalnik del svetovnega računalniškega omrežja (npr. *Stenar*), lahko na tem računalniku sprejemamo (in seveda pošiljamo) elektronsko pošto, hranimo zanimive datoteke, ki smo jih našli na drugih računalnikih v omrežju, itd. Ko pa želimo datoteke izpisati na svojem tiskalniku ali jih obdelovati na svojem računalniku, jih moramo prej še prenesti prek modema z gostiteljskega računalnika na svojega.

Pri pripravi datotek (ali pisem, ki jih želimo odposlati) je pametno, da jih pripravimo že na svojem računalniku, in šele nato prenesemo na oddaljeni računalnik. Na ta način znižamo telefonski račun, saj med pripravo datoteke ne zasedamo telefonskega priključka, hkrati pa tudi drugim omogočimo, da delajo na oddaljenem računalniku. Poleg tega delo na oddaljenem računalniku zahteva, da se naučimo novih ukazov za rutinska opravila, kar nam spet jemlje čas. Značilen primer je npr. priprava besedila; urejevalniki se med seboj precej razlikujejo, najbolje pa seveda obvladamo tistega na "domačem" osebnem računalniku.

Če želimo datoteko prenesti z oddaljenega računalnika na svoj računalnik ali obratno, potrebujemo tako na svojem kot na oddaljenem računalniku program, ki bo znal povezati oba računalnika in uspešno prenašati datoteke. V nadaljevanju bomo opisali dva najpogostejša taka programa, ki podpirata ustrezna protokola za prenos datotek: *Kermit* in *Zmodem*. Prvi je starejši in počasnejši, vendar ga najdemo povsod in je zelo zanesljiv. Drugi je novejši, dovolj zanesljiv in v splošnem hitrejši, zato si je pridobil precejšnjo popularnost.

Komunikacijskemu programu moramo povedati, kateri protokol za prenos datotek želimo uporabljati, kar ponavadi storimo z izbiro v meniju `Transfer` (oz. `Protocols` ipd.), kjer je v večini primerov na voljo *Zmodem*, *Kermit*, ali pa oba. Običajno nastavimo še tip prenosa, ki ga bomo uporabljali, to je prenos besedila (ASCII) oz. binarnih datotek (npr. programov ali stisnjenih datotek). Nastavitve, ki jih pogosto uporabljamo, je pametno shraniti.

Uporaba programa *Zmodem* na računalniku *Stenar*

Na *Stenarju* sta vnaprej definirana ukaza (natančneje, simbola[9]), ki nam olajšata prenašanje datotek. Ukaza sta **sz** (Send Zmodem – za pošiljanje) in **rz** (Receive Zmodem – za sprejemanje).[10]

Prenos datoteke z domačega računalnika na *Stenar* poteka po naslednjem postopku:

- Ko smo uspešno povezani in prijavljeni na računalnik *Stenar*, tam vtipkamo ukaz za sprejem datoteke:
    1. Če prenašamo datoteke v čisti ASCII obliki (izvirniki programov, besedila brez kode za oblikovanje,[11] opisi slik v PostScriptu, ipd.), potem vtipkamo ukaz **rz -a** (določilo **-a** pove, da gre za prenos v ASCII obliki).
    2. Če prenašamo datoteke v binarni obliki (sem spadajo izvedbene kode programov, besedila z vgrajeno kodo za oblikovanje videza, slike, in podobno), potem na *Stenarju* vtipkamo ukaz **rz**.
- V našem komunikacijskem programu izberemo pravo opcijo za pošiljanje datotek (`Upload`, `Transmit` ali kaj podobnega).
- Potem programu povemo, katero datoteko želimo poslati na oddaljeni računalnik, ki je v tem primeru *Stenar*.

Ko se programa sporazumeta in preneseta datoteko, jo lahko najdemo na *Stenarju* pod (običajno) enakim imenom, kot je na našem računalniku (če smo s svojega računalnika poslali npr. datoteko *pismo.txt*, se bo tudi na *Stenarju* imenovala *pismo.txt*). Če ne določimo drugače, se datoteka prenaša s tistega imenika (direktorija), v katerem se trenutno nahajamo, enako pa velja na oddaljenem računalniku (datoteka se "odloži" v trenutno izbrani

[9] Več o tem kasneje v poglavju o operacijskem sistemu VMS.

[10] Tu govorimo o t.im. "starem" *Zmodemu*, ki je v javni lasti. Simbola **sz** in **rz** sta torej ekvivalentna ukazoma **exe osz** in **exe orz**.

[11] Ta oblika je včasih označena tudi kot *plain text* ali kaj podobnega.

imenik), zato ne pozabimo pred prenosom datoteke preveriti, ali se na obeh računalnikih nahajamo v pravem imeniku.

Ko imamo datoteko na drugem sistemu, jo lahko npr. pošljemo z elektronsko pošto naprej in podobno. Kot bomo videli kasneje, lahko po elektronski pošti direktno pošiljamo le datoteke v ASCII obliki.

Prenos datoteke v drugo smer, torej s *Stenarja* na domači računalnik pa gre takole:

- Ko smo uspešno povezani in prijavljeni na računalnik *Stenar*, se najprej pomaknemo v imenik (direktorij), v katerem se nahaja datoteka, ki jo želimo prenesti. Nato vtipkamo ukaz za pošiljanje datoteke:

  1. Če prenašamo datoteko v čisti ASCII obliki (vzemimo za primer, da se ta datoteka imenuje *pismo.txt*), vtipkamo ukaz **sz -a** *pismo.txt* (zdaj moramo namreč oddaljenemu računalniku – *Stenarju* – povedati, katero datoteko naj pošlje).
  2. Če prenašamo datoteke v binarni obliki (naj se tokrat imenuje *slika.gif*), potem na *Stenarju* vtipkamo ukaz **sz** *slika.gif*

  Če so s pošiljanjem težave, včasih pomaga dodatno določilo **-e**.

- Če je naš komunikacijski program dovolj pameten (skoraj vsi novejši so taki), bo sam ugotovil, da mora datoteko sprejeti, tako da nam ni treba narediti ničesar več. Če nimamo te sreče, moramo v meniju za prenos izbrati opcijo `Donwload` ali `Receive`, s katero prenesemo datoteko na domači računalnik.

Enako kot prej velja, da ima datoteka na našem računalniku enako ime kot na oddaljenem, z naslednjimi omejitvami, specifičnimi za DOS[12]: če je ime datoteke daljše od 8 znakov, se ostali znaki odrežejo; v DOSu nedovoljeni znaki v imenu datoteke (npr. znak "_") se običajno spremenijo v znak "X". Prenešeno datoteko na osebnem računalniku seveda najdemo v trenutno privzetem imeniku.

Uporaba programa *Kermit* na računalniku *Stenar*

Prenos poteka zelo podobno kot s programom *Zmodem*. Pred prenosom datotek na *Stenarju* poženemo program *Kermit*, kar storimo z ukazom:

$ **kermit**

Program prevzame nadzor nad ukazi, kar opazimo po spremenjenem pozivu

[12] Z DOSom je v tem priročniku vedno mišljen operacijski sistem MS DOS.

`Kermit-32>`

Ukaze, ki so nam na voljo, lahko preučimo z ukazom **help**.

Kermit privzame, da prenašamo datoteke v ASCII obliki. Če želimo prenašati binarne datoteke, nastavimo način prenosa s

`Kermit-32>` **set file type binary**[13]

Nadaljnji postopek prenosa poteka po naslednjih korakih:

1. Če prenašamo datoteke z domačega računalnika na *Stenar*:
   - vtipkamo **receive**,
   - nato moramo v našem komunikacijskem programu izbrati `Kermit` kot protokol za prenos, potem pa še pravo opcijo za pošiljanje datotek (`Upload` ali `Transmit`).
   - nazadnje programu povemo, katero datoteko želimo poslati na oddaljeni (remote) računalnik, ki je v tem primeru *Stenar*.
2. Če prenašamo datoteke s *Stenarja* "domov":
   - vtipkamo **send** *ime_datoteke*,
   - nato v našem komunikacijskem programu izberemo `Kermit` kot protokol za prenos, potem pa še pravo opcijo za pošiljanje datotek (`Download` oz. `Receive`).

Iz programa *Kermit* izstopimo z **exit**.

2.2.5 Izbira komunikacijskega programa

Kakšen program potrebujemo, je precej odvisno od našega dela. Pogosto nam zadošča že emulacija VT100, ki jo zmore tako rekoč vsak boljši program, včasih pa potrebujemo emulacijo višjega tipa. Za dostopanje do knjižničnega sistema COBISS, na primer, potrebujemo najmanj terminal VT220, ki omogoča nalaganje lastnega nabora znakov, da lahko prikaže slovenske črke.

Nekateri programi so v *javni lasti*,[14] (npr. *Kermit, WinQVT*), obstajajo tudi *preizkusni programi*[15] (npr. *Telix, Telemate*). Kupimo pa lahko tudi dobre profesionalne programe

[13] Nastavitev "v drugo smer" je seveda **set file type ascii**.

[14] Angleško *public domain*. Taki programi so brezplačno na voljo. Nekatere lahko dobite na ARNESovem strežniku *ftp.arnes.si*; kako priti do njih, pa izveste v razdelku 4.2.2.

[15] Angleško *shareware*. Ti programi so vsem na voljo za preizkušanje, če pa jih redno uporabljamo, smo dolžni avtorju plačati uporabnino, ki je običajno majhna.

z dokumentacijo in vso podporo (npr. *Reflection, KeaTerm, ProComm*). Cena takšnih programov je velikostnega reda nekaj sto DEM v tolarski protivrednosti.

Povejmo še, da običajno ob nakupu modema dobimo priložen tudi kak komunikacijski program, vendar ponavadi manj kakovostnega.

Na kratko bomo predstavili le nekaj programov za komunikacijo: *WinQVT*, *Telix*, *Reflection4* in *KeaTerm*.

Za običajno delo povsem zadošča program *WinQVT*, ki deluje pod MS Windows. Zelo preprost za uporabo, omogoča pa solidno emulacijo terminala VT220. Program je javno dostopen na ARNESovem strežniku, lahko pa ga naročite pri ARNESu tudi po pošti. Na enak način lahko dobite tudi slovenska navodila za uporabo.

Telix je javno dostopen program, ki ga lahko preizkusite, preden plačate avtorju uporabnino. Deluje pod DOSom, obstaja pa tudi verzija za MS Windows. *Telix* je zelo uporaben program za delo z modemom, saj je dobro podprt s strani avtorja in se veliko uporablja. Nudi nam vse potrebno za delo z raznovrstnimi sistemi (npr. za klic na BBS). Delo s programom je zelo preprosto; program lahko usposobimo že v nekaj minutah, saj so privzete vrednosti dobro izbrane. Zelo dobro oponaša tudi druge terminale. Program lahko najdemo med preizkusno programsko opremo na na ARNESovem FTP strežniku.

Reflection4 in *KeaTerm* sta za razliko od *Telixa* visoko profesionalna izdelka, ki ju moramo kupiti pri prodajalcu, skupaj z navodili in zagotovljeno tehnično podporo. Oba programa delujeta v okolju Windows in sta dovolj preprosta, da ju lahko začnemo uporabljati takoj in beremo navodila sproti. Za razliko od prejšnjega programa podpirata malce več možnosti za komunikacijo, saj omogočata delo prek serijskih linij, LAT, DECnet, TCP/IP, X.25 itd. Programa znata tudi bolje posnemati terminale, saj sta združljiva s tipom VT420. V začetku se nam to morda ne zdi pomembno, pri pogostejšem delu pa znamo ceniti dodatne možnosti (npr. hkratno delo z več sejami, makre, predefinirano tipkovnico in podobno). Oba programa nudita približno enako in tudi staneta približno enako. *ProComm* podpira manj različnih protokolov, uporabniku pa prav tako nudi dovolj udobja.

3.

Navodila za delo na sistemu VAX/VMS

Uvod

Mnogi uporabniki omrežja ARNES opravljajo svoje vsakodnevno delo na lokalnih računalnikih (PC-jih, delovnih postajah), za povezavo z omrežjem pa jim služi računalnik VAX z operacijskim sistemom VMS,[1] kakršen je računalnik z imenom *Stenar*.

V tem poglavju je podanih nekaj osnovnih navodil za delo z operacijskim sistemom VMS. To besedilo vas ne nauči vsega o tem sistemu. Namenjeno je tistim, ki se ne želijo poglabljati v VMS, radi bi le pošiljali (in brali) pošto, sodelovali v elektronskih konferencah ter raziskovali neizčrpne vire informacij po vsem svetu. Bolj zaokroženo seznanjanje z VMS omogočajo specializirani priročniki, kot je npr. "Operacijski sistem VMS za začetnike" Maje Ferle.

V prvem delu tega poglavja so podana osnovna navodila za delo s sistemom VMS. V naslednjem delu so opisana najnujnejša navodila za delo z datotekami. Sledi kratek opis urejevalnika besedil *EDT*. Podrobneje je obdelano pošiljanje elektronske pošte s programom *MAIL* in sistem *NOTES*, ki omogoča sodelovanje v slovenskih elektronskih konferencah.

Opozorilo: Vsi uporabniki nimajo enakih tipkovnic, zato ni bilo mogoče natančno poimenovati nekaterih tipk, ki jih uporabljamo pri vnašanju ukazov – na različnih tipkovnicah so lahko različno označene. Primer je tipka $\boxed{\text{Return}}$, ki jo uporabljamo za zaključevanje vrstice ali ob koncu ukaza. Označena je lahko tudi s $\boxed{\hookleftarrow}$ ali z $\boxed{\text{Enter}}$, ki pa je v nekaterih aplikacijah ne smemo zamenjevati s tipko $\boxed{\text{Enter}}$ na stranski tipkovnici. Zato je pri tipkah z dvoumnimi oznakami poudarjeno, kakšno funkcijo naj bi opravljale.

[1]Novejša različica tega operacijskega sistema, ki je sedaj v uporabi, se pravzaprav imenuje OpenVMS. Vendar bomo v tem priročniku uporabljali udomačeno skupno ime VMS.

3.1 Osnovna navodila

3.1.1 Prostorske omejitve na večuporabniških sistemih

VMS je značilen primer sistema, zasnovanega za hkratno večuporabniško delo. Delo na takem sistemu se precej razlikuje od dela z osebnim računalnikom. Za razliko od osebnega računalnika, kjer ima uporabnik dostop do vseh datotek na disku (sistemski programi, uporabniške datoteke – naj so njegove ali ne), sta na večuporabniškem sistemu lastništvo in dostop do datotek zelo natančno razmejena s sistemom zaščit in privilegijev. Vsakemu uporabniku pripada na disku imenik datotek (direktorij), v katerem sme ustvarjati nove datoteke. Uporabnik ima na razpolago omejen prostor na disku, ki ga sme zasedati s svojimi datotekami. Obseg tega prostora (*disk quota*) je odvisen od zasedenosti diskov na določenem sistemu in za "navadnega uporabnika" običajno ni zelo velika – nekaj Mbytov prostora. Posledica je večja potreba po urejenosti in pravočasnem odstranjevanju odvečnih datotek. Sistemsko sporočilo

```
%SYSTEM-F-EXDISKQUOTA, disk quota exceeded
```

ki se lahko pojavi ob poskusu kopiranja ali odpiranja nove datoteke, pomeni, da je uporabnik zapolnil svoj prostor na disku. Reakcija je zato lahko ena sama – čiščenje. Le v izjemnem primeru lahko sledi prošnja sistemskemu administratorju za povečanje osebnega prostora na disku.

3.1.2 Kako začnemo?

Preden lahko začnemo delati na večuporabniškem sistemu, se moramo nanj prijaviti. To nam omogoča uporabniško ime, ki smo ga dobili pri sistemskem administratorju. Postopek teče takole: ko se z modemom uspešno povežemo z gostujočim računalnikom, se ta navadno predstavi, vsekakor pa sledi poziv

```
Username:
```

Če se nič ne zgodi, je včasih potrebna spodbuda s pritiskom (ali dvema) na tipko $\boxed{\text{Return}}$. Če tudi to ne pomaga, potem zveza morda ni tako uspešno vzpostavljena, kot bi si želeli.

Na ta poziv odtipkamo svoje uporabniško ime (vnos seveda zaključimo s tipko $\boxed{\text{Return}}$). Sledi nov poziv

```
Password:
```

Zdaj odtipkamo še geslo, vendar ga ne bomo videli. Če smo se zmotili, nas sistem zavrne z odpovedjo "`User autorization failure`". Dregnemo ga z $\boxed{\text{Return}}$ in začnemo še enkrat.

Če je uspelo, nam sistem izreče dobrodošlico z lokalnim sporočilom, in končno ugledamo sistemski poziv (angleško *prompt*), navadno takle

```
$
```

Zdaj lahko začnemo vnašati ukaze. Najprej pa si zapomnimo, da se moramo po končanem delu na tem sistemu odjaviti z ukazom **logout**.

Če so nam geslo dodelili drugi, ga najbrž želimo spremeniti, kar se tako ali tako spodobi storiti vsaj vsakih nekaj mesecev. To na *Stenarju* opravimo z ukazom **passwd**.[2] Računalnik nato zahteva od nas staro in novo geslo, ki ga še enkrat preveri. Celoten dialog poteka takole:

```
$ passwd
_Old password: starogeslo
_New password: novogeslo
_Verification: novogeslo
```

pri čemer sistem gesla seveda ne pokaže na zaslonu.

Včasih sistem sam od sebe zahteva spremembo gesla. To se zgodi v rednih časovnih presledkih (nekaj mesecev) ali kadar nam sistemski administrator nastavi začasno geslo. V takem primeru je treba ubogati, saj od tistega trenutka dalje staro geslo ne velja več.

Še beseda o izbiri in varovanju gesla. Večkrat poudarjamo, da delo na omrežju poleg ugodnosti prinaša tudi odgovornost. Zloraba uporabniškega imena, ki smo jo omogočili iz malomarnosti, lahko povzroči škodo ali pa nam vsaj nakoplje kup sitnosti. Zato na svoje geslo skrbno pazimo in ga ne smemo "posojati". Najbolje je, da ga nikamor ne zapisujemo, če pa že moramo, ga nosimo npr. v denarnici in ga ne puščamo na mizi. Večina vdorov v računalniške sisteme se posreči z ugibanjem preprostih gesel, zato je dobro, če se jih naučimo pametno izbirati.

Geslo naj nikdar ne bo enako uporabniškemu imenu, niti kakšnemu lahko dostopnemu podatku, ki se nanaša na uporabnika (priimek, telefonska številka). Boljše je tudi, če sploh ni beseda, ki jo je moč najti v kakem slovarju, kajti vlomilski programi znajo uporabljati tudi slovarje. Najboljše je (navidez) naključno zaporedje črk, če pa primešamo še nekaj številk (naj ne bodo samo številke!), toliko bolje. Seveda naj bo to kombinacija, ki si jo lahko zapomnimo, da nam je ni treba imeti zapisane na listku. Uporaben trik je npr. kombinacija dveh besed s številko, še bolj pa prve ali zadnje črke besed iz stavka, ki si ga zlahka zapomnimo, na primer `vdio13uk`[3].

[2]Na drugih VMS sistemih je ustrezen ukaz **set password**.
[3]Vsak Dan Imam Ob 13. Uri Kosilo.

3.1.3 Sintaksa ukazne vrstice

Za lažje razumevanje omenimo, da je kratica DCL, ki jo boste večkrat srečali (npr. v *HELPu*), oznaka za ukazni jezik (Digital Command Language). Izraz "ukaz v DCL" torej v praksi pomeni nekaj podobnega kot "DOSov ukaz", le da ima DCL precej bogatejšo strukturo, ki omogoča pisanje zmogljivih ukaznih procedur.

Navaden ukaz je v splošnem videti nekako takole:

$ *ukaz/določila parametri /določila...*

Pri tem sistem ne razlikuje med velikimi in malimi črkami (kar npr. za Unix ne drži). Določil je lahko več. Tudi parametrov je lahko več, med seboj in od ukaza pa morajo biti ločeni s presledki. Poševna črta je del določila (brez presledka!), pred njim je presledek dovoljen, ne pa nujen. Določila podrobneje določajo delovanje ukazov. Tako npr. ukaz

$ **type/page** *bla.txt*

izpiše na zaslon datoteko *bla.txt* po delih (kolikor gre pač na zaslon), medtem ko samo **type** izbruhne vso datoteko naenkrat. Ukaze, določila in parametre (če seveda ne gre za imena) lahko okrajšamo toliko, da se še ločijo od drugih ukazov (določil...).

Za podrobnejši opis delovanja posameznih ukazov je na voljo ukaz **help**. Vodi nas po drevesno strukturirani informaciji o ukazih in njihovih parametrih, kar ni preveč zapleteno: na poziv `Topic?` oz. `Subtopic?` vtipkamo eno od gesel, ki nam jih računalnik trenutno ponuja, da dobimo o njem podrobnejšo informacijo:

$ **help**

... sledijo navodila in seznam ukazov

`Topic?` **type**

... sledi opis ukaza in seznam določil

`TYPE Subtopic?` **/page**

... opis učinka določila

...

Če namesto tega pritisnemo tipko Return, se vrnemo na prejšnji nivo, s tipko ? pa ponovno prikličemo informacijo s tega nivoja, da si lahko še enkrat ogledamo gesla, ki so trenutno na voljo. Ubogajte nasvet v glavnem menuju in preberite najprej *Instructions* in *Hints*. Iz kateregakoli nivoja lahko iz programa izstopimo s Ctrl–Z, pri čemer zopet ugledamo sistemski poziv.

3.1.4 Posebne tipke

Pri vnašanju ukazov si lahko pomagamo z nekaterimi tipkami za popravljanje napak. S puščicama ← in → premikamo kazalec po ukazni vrstici, v katero lahko vrivamo znake (kombinacija tipk Ctrl–A preklaplja med vrivanjem in prekrivanjem znakov). Tipka Delete zbriše znak pred kazalcem[4], kombinacija tipk Ctrl–U pa zbriše trenutno ukazno vrstico.

V dodatno pomoč nam je tipka ↑, ki prikliče prejšnjo (pa še prejšnjo, pa še eno pred njo...) ukazno vrstico, ki jo seveda lahko nekoliko prilagodimo in si tako prihranimo tipkanje novega ukaza. Tipka ↓ se giblje v nasprotno smer (naprej) med ukazi, ki jih računalnik hrani v spominu.

Poleg omenjenih tipk za urejanje ukazne vrstice poznamo še nekaj tipk s posebno funkcijo:

Ctrl–Z
Običajno vrnitev ("Exit") iz posebnega programa (kot v primeru *HELP*). Včasih znak za konec datoteke.

Ctrl–C
Se uporablja znotraj kompleksnejših programov (recimo v urejevalniku besedil) za prekitev izvajanja internega ukaza ("Cancel", "Abort"). Pri prekinjanju preprostejših programov (ukazov) se obnaša podobno kot Ctrl–Y, le da slednja izvajanje dokončno ustavi – **continue** ne pomaga več.

Ctrl–Y
Prekine izvajanje trenutnega ukaza oz. programa ("Interrupt") in vrne sistemski poziv (vrne nadzor sistemu). Tipka, ki vas reši skoraj vedno, ko se kaj "dokončno" zatakne, glede na to, da *tega* računalnika ponavadi ne boste mogli sami ugasniti. Če je res hudo, je vredno poskusiti večkrat zapored. Pozor, uporabniki urejevalnika *Wordstar* in podobni, ki vam ta tipka pomeni čisto nekaj drugega – deluje tudi v urejevalniku besedil! Če pa se že zatipkate – ukaz **continue** v DCL nadaljuje izvajanje prekinjenega programa.

[4] Ista tipka, ki jo v ta namen uporabljate na svojem računalniku; na nekaterih tipkovnicah je lahko označena z Backspace ali celo ←; nahaja se na glavnem delu tipkovnice nad Return in z oznako Delete bomo imeli vedno v mislih to tipko, ne pa tipke Delete, ki jo včasih najdemo na stranski tipkovnici.

Ctrl–S
Zamrzne ekran in tipkovnico – izpis na ekran se ustavi, vključno z izpisovanjem znakov, ki prihajajo s tipkovnice. Posledično se (začasno) ustavijo programi, ki izpisujejo na zaslon (npr. **type**), tipkanje pa navidez nima učinka. Če se vam kdaj zazdi, da na lepem nič ne deluje, pomislite tudi, ali se ni morda od kod prikradel kakšen Ctrl–S!

Ctrl–Q
Odmrzne ekran in tipkovnico – sprosti izpisovanje na zaslon, ki je bilo zaustavljeno s Ctrl–S.

Ctrl–O
Izmenično ukine/omogoči prikazovanje izpisa podatkov na zaslon. Po učinku se od para Ctrl–S/Ctrl–Q razlikuje po tem, da se v tem primeru izpisovanje nadaljuje, le mi ga ne vidimo, in tako del izpisanih podatkov "izgubimo".

Tipka No Scroll, Hold Screen, Pause ali kaj podobnega izmenično pošilja Ctrl–S in Ctrl–Q.

3.2 Datoteke in imeniki datotek (direktoriji)

3.2.1 Delo z datotekami na VMS

Celotno ime datoteke je naslednje oblike:

RAČUNALNIK::*ENOTA*:[*imenik_datotek*]*ime.podaljšek;verzija*

Tako kot pri ukazih tudi v imenih ni pomembno, ali so črke velike ali majhne. Ta oblika v grobem ustreza DOSovi sintaksi *enota:\imenik_datotek\ime.podaljšek.* Dodano je ime računalnika, na katerem se datoteka nahaja (pomembno za omrežje), in številka verzije datoteke. Ker bodo naše datoteke vedno na istem računalniku in isti enoti (disku), lahko na prva dva dela imena (zapisana z veliki črkami) odslej popolnoma pozabimo, ker ju računalnik privzame sam.

Ostane nam [*imenik_datotek*]*ime.podaljšek;verzija.* Ime datoteke je lahko sestavljeno iz največ 39 znakov, ki so lahko črke (brez šumnikov!), številke, ter znaka $ in _ (podčrtaj).

Podaljšek je sestavljen iz enakih znakov kot ime. Nekateri standardni podaljški označujejo tip datoteke (*.dir, .pas, .txt, ...*)

Imeniki datotek (direktoriji) imajo enako vlogo na vseh računalnikih, namreč pregledno razdeljujejo veliko število datotek. Na sistemu z več uporabniki je vsakemu uporabniku dodeljeno področje (uporabniški imenik) na disku, kjer ima svoje datoteke. Ostale datoteke so v splošnem nedosegljive, razen javnih (npr. programi). Ime vašega (glavnega) imenika je enako vašemu uporabniškemu imenu.

V imenu datoteke na VMS je imenik ločen od datoteke z oglatimi oklepaji, podimeniki pa med seboj s pikami. Za zgled primerjajmo ime konkretne datoteke v DOSu z enako sestavljenim imenom datoteke v VMS:

DOS : *C:\tone\recepti\torte\sacher.txt*
VMS : *Stenar::USER$DISK:[tone.recepti.torte]sacher.txt;1*

oziroma brez privzetega dela imena (računalnik, enota):

DOS : *\tone\recepti\torte\sacher.txt*
VMS : *[tone.recepti.torte]sacher.txt;1*

Pri vsaki spremembi ali nastanku datoteke z istim imenom nastane nova verzija datoteke. Zaporedna številka verzije je zadnji podatek v imenu, ki je značilen za sistem VMS. Slaba stran tega principa je, da se tako ustvarja veliko datotek s skoraj enako vsebino, ki zasedajo prostor na disku. Dobra stran pa je, da včasih "čez sedem let tudi stara verzija prav pride", predvsem pa se nam tako ne more zgoditi, da bi npr. pri prepisovanju po nesreči "povozili" datoteko, ki ima naključno enako ime (nastane nova verzija!). Najnovejša verzija datoteke ima najvišjo številko in ta je tudi privzeta vrednost, če je ne navedemo (običajno želimo delati z zadnjo verzijo datoteke).

Kar takoj povejmo, da se odvečne navlake starih verzij znebimo z ukazom **purge**, ki pobriše vse razen zadnje verzije datoteke, razen če z določilom posebej ne zahtevamo, da bi jih radi obdržali več. Tako na primer ukaz

$ purge *bla.bla* **/keep=3**

zbriše vse verzije datoteke *bla.bla* razen zadnjih treh. Zaradi večje preglednosti in varčevanja s prostorom je priporočljivo, da stare verzije redno brišemo.

Kako datoteke, ki bi jih potem radi zbrisali, najprej naredimo, pa bomo izvedeli v naslednjem razdelku, ki govori o urejevalniku besedil.

Omenili smo že, da ukaz **type** *ime.dat* izpiše vsebino datoteke na zaslon, kot smo navajeni tudi na osebnih računalnikih. Podobno velja za ukaz **dir**, ki izpiše seznam datotek v trenutnem imeniku. Omenimo določili **/size** in **/date**, ki dodata k izpisu še velikosti in datume nastanka datotek. Velikost je izražena v *blokih* (blok = 512 znakov).

Če hočemo naenkrat "opraviti" z več datotekami, jih lahko navedemo zaporedno in jih ločimo z vejicami, na primer

$ **type** *tole.dat,tistole.txt,pa_recimo_se_tole.txt*

Kadar imajo datoteke podobna imena, si lahko nekaj tipkanja prihranimo z uporabo znaka $*$, ki nadomesti poljubno skupino znakov, ter znak %, ki nadomesti natanko en znak. Z vključevanjem teh znakov[5] dobimo namesto navadnega imena datoteke *imenski vzorec*. Ukaz se izvede za vse datoteke, ki ustrezajo vzorcu. Gornji ukaz bi lahko napisali recimo takole:

$ **ty** $*$*tole.%%t*

Temu vzorcu seveda lahko ustreza še kakšna druga datoteka, na kar je dobro paziti predvsem pri brisanju. Tako npr. vzorcu *a*$*$*a.%h* ustreza tako *aa.ah* kot *abra1_2_3kadabra.oh*. Za vzorec se šteje tudi nepopolno ime datoteke. Povedali smo že, da v tem primeru veljajo privzete vrednosti. V nekaterih primerih je privzeta vrednost $*$. Ukaz **dir** je na primer ekvivalenten ukazu **dir** $*.*$

Za brisanje datotek uporabljamo ukaz **delete**. Pri imenu datoteke moramo navesti tudi številko verzije, ali pa $*$, kadar želimo pobrisati vse verzije datoteke. Kadar brišemo več datotek (z uporabo $*$), je pogosto koristno dodati določilo **/confirm**, ki povzroči, da nas sistem za vsako datoteko, ki ustreza vzorcu, pred brisanjem vpraša za potrditev, na primer

$ **del/confirm** *z*$*$*.txt;*$*$
`USER$DISK:[TONE]ZELO_VAZNA_DATOTEKA.TXT;23, delete? [N]:`

Privzet odgovor ob pritisku na Return je **n** ("nikar!"), lahko pa odgovorimo tudi **y** ("le pogumno zbriši!"), **a**(ll) ("zbriši to in še vse ostale in ne sprašuj več!") in **q**(uit) ("prenehaj z brisanjem").

Pri nekaterih ukazih, kot so **directory**, **type** ali **delete**, lahko uporabljamo določili **/since** in **/before**, ki izbirata datoteke glede na čas nastanka. Ukaz

$ **dir** [*tone.recepti*] **/since**=*1-jan-1994*

na primer pokaže vse datoteke na podimeniku *recepti*, ki so nastale po letu 1993. Datum napišemo v obliki *dan–mesec–leto* kakor v zgornjem primeru: *dan* in *leto* sta številki, *mesec* pa zapišemo s prvimi tremi črkami (angleškega) imena meseca. To je standardni datumski format v VMS, v priročnikih označen z *DD-MMM-YYYY*. Veljavna vrednost za določili je še *YESTERDAY*, privzeta vrednost (če datuma ne napišemo) pa je *TODAY*.

Prepisovanje datotek (kopiranje) opravlja kot na osebnem računalniku ukaz

$ **copy** *stara.dat nova.dat*

[5] Pravimo jim *wildcard* ali po domače *džoker*, kot pri kartah.

ki ustvari novo datoteko in vanjo prepiše vsebino stare, ki se seveda ohrani. Kadar prepisujemo datoteko iz enega imenika v drugega (glej razdelek o imenikih), je potrebno navesti tudi to. Tako si lahko Tone prepiše recept iz Janezovega podimenika v svojega takole:

$ **copy** *[janez.recepti]pizza_napolitana.txt* *[tone.pizze]napolitana.txt*

Tudi pri tem ukazu lahko uporabljamo imenske vzorce. Ukaz

$ **copy** *[janez.recepti]*pizza*.txt* [ ]

na primer prepiše vse Janezove recepte, katerih imena vsebujejo besedo “pizza” (če zaščita datotek tega ne prepoveduje) na naš trenutni imenik (vzorec “[ ]” pomeni naš trenutni imenik, ime pa ostane enako kot pri izvirniku, ker nismo navedli novega), medtem ko

$ **copy** *_torta.txt vsetorte*

naredi novo datoteko *vsetorte.txt* (*.txt* privzame), na katero prepiše po vrsti vse datoteke, ki ustrezajo imenu na levi. Stare datoteke se seveda ohranijo.

Tisto prepisovanje, ki je za uporabnike tega priročnika najbolj zanimivo, to je prenos datotek med dvema računalnikoma, obravnavamo v drugih poglavjih tega priročnika. Poleg prepisovanja pozna VMS še preimenovanje

$ **rename** *staro.ime novo.ime*

ki seveda ne ustvari nobene nove datoteke.

3.2.2 Delo z imeniki datotek

Kot že rečeno, je vsakemu uporabniku na VMS dodeljeno področje na disku – njegov uporabniški imenik. Ob prijavi na sistem se znajdemo v tem imeniku, ki nosi tudi naše uporabniško ime.

Uporabnik TONE bo tako po prijavi na sistem na zahtevo

$ **show default**

ki pokaže trenutni imenik, dobil odgovor

```
USER$DISK:[TONE]
```

ki vsebuje ime njegovega imenika in ime diska, kjer se imenik nahaja. Nove podimenike si lahko TONE napravi z ukazom **create/directory**, na primer takole:

$ **create/directory** [*tone.pisma*]
$ **cre/dir** [*.recepti*]
$ **cre/dir** [*.recepti.torte*]

V drugem in tretjem ukazu pred piko manjka ime imenika. Računalnik na to mesto vstavi ime trenutnega imenika (v našem primeru *tone*). Zdaj ima Tone v glavnem imeniku dva podimenika (*pisma* in *recepti*), v podimeniku *recepti* pa še podimenik *torte*. V izpisu seznama datotek prepoznamo imenike po podaljšku *.dir*. S tremi zaporednimi pikami v parametru označujemo trenutni imenik in vse podimenike, ki so v drevesni strukturi nižje od danega imenika. Če bi želeli spisek vseh datotek tipa *txt*, ki se nahajajo na trenutnem imeniku in na vseh podimenikih trenutnega imenika (in njihovih podimenikih...), to lahko napravimo z ukazom

$ **dir** [...]∗*.txt*

Vrednost trenutnega imenika spreminjamo (po domače: "se sprehajamo po direktorijih") z ukazom **set default**. Po ukazu

$ **set def** [*.pisma*]

(ki je skrajšana oblika za **set def** [*tone.pisma*]) je torej novi trenutni imenik [*tone.pisma*]. Za en imenik višje splezamo na kratko s

$ **set def** [–]

Namesto ukaza **set default** lahko uporabljamo tudi simbol **sd** ali pa, kdor je tega bolj vajen, **cd**[6].

"Preseljevanje" datotek iz enega imenika v drugega ("move") opravimo s preimenovanjem:

$ **ren** [*.recepti*]*malinova_torta.txt* [*.recepti.torte*]

Po izvršitvi tega ukaza se recept za malinovo torto nahaja v podimeniku [*.recepti.torte*], v podimeniku [*.recepti*] pa ostanejo stare verzije datoteke, če jih še nismo pobrisali. Če pa bi namesto **ren** uporabili **copy**, bi imeli enaki datoteki v obeh podimenikih. Kot pri prepisovanju se manjkajoči podatki v parametrih (pred prvo piko v oglatih oklepajih, številka verzije, ime nove datoteke) nadomestijo s privzetimi vrednostmi (trenutni imenik, zadnja verzija, isto ime).

[6]Velja za uporabnike na računalniku *Stenar*, ni standardni del VMS. Pojasnilo o simbolih sledi na koncu poglavja, zaenkrat jih lahko enačite z ukazi.

3.2.3 Zaščita datotek

Datoteke ali cele imenike lahko zaščitimo proti različnim skupinam uporabnikov (tudi proti sebi) za različne skupine dostopa. Skupine uporabnikov so:

SYSTEM – uporabniki s sistemskimi privilegiji,
OWNER – lastnik datoteke,
GROUP – uporabniki iz iste skupine[7],
WORLD – vsi uporabniki na sistemu in v omrežju.

Vrste dostopa do datoteke pa so:

READ – dovoljeno branje,
WRITE – dovoljeno pisanje v datoteko (ali imenik),
EXECUTE – dovoljeno izvajanje (npr. za program),
DELETE – dovoljeno brisanje.

Zaščito datoteke ali imenika datotek si lahko ogledamo z **dir/protection** *datoteka*, nastavimo pa jo z ukazom

$ set protection=(s:rwe,o:rwed,g:re,w) *datoteka*

V oklepaju za vsako skupino uporabnikov (s, o, g, w) navedemo, katere vrste dostopa ji dovolimo. V zgornjem primeru je prepovedano brisanje za skupino *SYSTEM* (razen če se posebej ne potrudijo), lastniku je dovoljeno vse, članom iste skupine je dovoljeno branje in izvajanje programov, vsem ostalim pa nič (kar ni dovoljeno, je prepovedano).[8] Včasih se je pametno zavarovati pred samim seboj (brisanje po nemarnem); tako so vnaprej zavarovani vsi imeniki. Če bi radi zbrisali kak podimenik, ga je treba najprej "izprazniti", nato pa pred brisanjem še spremeniti zaščito, na primer takole:

$ set prot=(o:rwed) *odvecni.dir;1*
$ del *odvecni.dir;1*

Še komentar: zaščita se ni spremenila za tiste skupine, ki jih nismo navedli v ukazu (vse razen lastnika).

[7]Vsi zunanji uporabniki na *Stenarju* so v isti skupini
[8]Pozor: skupine "o", "g" in "w" so hierarhične: če je nekaj dovoljeno širsi skupini, ne more biti prepovedano za ožjo.

3.2.4 Simboli kot posebna oblika ukazov

Simboli, ki nadomeščajo nize znakov in vrednosti, in logična imena, ki nadomeščajo fizična imena datotek, imenikov in enot, so ključnega pomena predvsem za ukazne procedure, zelo prikladni pa so tudi pri neposrednem delu z VMS.

Za namene tega priročnika bomo le površno pojasnili razliko med simboli in standardnimi ukazi v DCL. Kogar to ne zanima, lahko v tem priročniku brez škode bere "ukaz" namesto "simbol".

Nam bo torej simbol pomenil niz znakov, ki ga definiramo sami (ali pa to naredijo vzdrževalci sistema) kot sinonim nekemu (običajno bolj zapletenemu) ukazu. Poglejmo si to na primeru ukaza **show default**. Ena od prijetnih lastnosti DCL je namreč ta, da za ukaze in določila uporablja angleške besede, ki povzemajo njihovo delovanje. Tako si jih lažje zapomnimo. Vendar se uporabniki hitro naveličajo vsakič pisati **set def** (minimum za ta primer), če ukaz pogosto uporabljajo, še posebej, kadar so navajeni na kakšnem drugem računalniku uporabljati na primer ukaz **cd**. Zato lahko definiramo simbol (sinonim) takole:

$ cd == "set default"

Kot rečeno, je ta simbol na *Stenarju* že vnaprej definiran. Uporabljamo ga kot navaden ukaz:

$ cd [.*recepti*]

Tako definiran simbol velja, dokler smo prijavljeni na sistemu. Če ga želimo uporabljati stalno, lahko gornjo definicijo z urejevalnikom besedil dodamo v inicializacijsko ukazno datoteko *login.com* v glavnem imeniku, ki se izvrši ob vsaki prijavi. Še enkrat pa opozarjamo, da je opis simbolov v tem priročniku namenoma močno okrnjen, zato odsvetujemo definiranje novih simbolov, če ne veste natančno, kaj delate.

Za skupino uporabnikov je na računalniku *Stenar* definiranih nekaj simbolov, ki naj bi olajšali uporabo nekaterih pogostejših ukazov, kot so klici programov, ki opravljajo omrežne storitve. Nekaj simbolov je definiranih za ves sistem. Vse definirane simbole si lahko ogledamo z ukazom

$ show symbol/global/all

Namen tega kratkega pojasnila je predvsem ta, da bi se zavedeli razlike med simbolom in standardnim ukazom v DCL. To pomeni, da je za simbol, ki je omenjen v tem priročniku, do določene mere *verjetno*, ne pa tudi nujno, da ima enak učinek na kakem drugem računalniku tipa VAX.

Tabela najpogosteje uporabljanih ukazov in določil

UKAZ	PARAMETER	UČINEK
directory	—	Pokaže seznam datotek v tekočem imeniku.
directory	*datoteka*	Pokaže le tiste datoteke, ki ustrezajo vzorcu *datoteka*.
dir/date/**si**ze	*datoteka*	Seznamu doda še velikost in datum nastanka datotek.
type	*datoteka*	Izpiše vsebino datoteke.
type/**pa**ge	*datoteka*	Izpiše vsebino po straneh.
show **quo**ta	—	Izpiše, koliko uporabnikovega prostora na disku je porabljenega in koliko je še prostega.
show **def**ault	—	Pokaže tekoči imenik.
create/**dir**ectory	*imenik*	Ustvari nov imenik.
set default	*imenik*	*imenik* postane tekoči imenik.
copy	*dat1 dat2*	Prepiše *dat1* na *dat2*.
rename	*dat1 dat2*	Preimenuje *dat1* v *dat2*.
delete	*datoteka*	Zbriše datoteko.
delete/**co**nfirm	*datoteka*	Zbriše datoteko, a prej vpraša.
purge	*datoteka*	Zbriše vse stare verzije datoteke.
purge/**keep**=n	*datoteka*	Enako, le da obdrži zadnjih n verzij.
logout	—	Odjava od sistema ob zaključku dela.
help	—	Pomoč na sistemu.

3.3 Urejevalnik besedil *EDT*

Če smo navajeni na kak urejevalnik besedil, je privajanje na drugačnega ponavadi nevesela naloga. Za uporabnike, ki jim je ta priročnik namenjen, bo urejevalnik besedil na sistemu VMS le pomožno orodje, ki še najbolj prav pride npr. pri takojšnjem odgovarjanju na pošto, pisanju odgovorov v elektronske konference in pri podobnih manj zahtevnih obdelavah besedila. Za zahtevnejšo obdelavo in vnaprejšnjo pripravo besedila (tudi pisanje pisem in prispevkov v konference) raje uporabljajmo svoj računalnik, in ko končamo, prenesimo datoteke prek modema in telefonskega omrežja na oddaljeni računalnik.

3.3.1 Osnovni ukazi za urejanje besedil

V tem poglavju se bomo seznanili z osnovami urejevalnika *EDT*, ki je najbolj razširjen na VMS sistemih. Da ne bo pomote – besedila, ki jih urejamo z *EDT*, so navadne ASCII datoteke, ne pa dokumenti z oblikovanim besedilom (različne pisave, elastični presledki, grafika). To ne pomeni ovire, saj bomo potrebovali prav tak urejevalnik. Eden od razlogov je ta, da z elektronsko pošto lahko normalno pošiljamo le ASCII datoteke (binarne datoteke kodiramo v ASCII obliko). Te omejitve se moramo zavedati tudi, ko besedilo pripravljamo na domačem urejevalniku. Datoteke, ki naj bi po omrežju potovale kot navadna besedila, je treba torej vedno pripravljati kot navadna (ASCII) besedila (npr. "nondocument" način v urejevalniku *Wordstar*), ali pa jih pretvoriti v obliko ASCII pri shranjevanju z urejevalnikom (tako *Word* kot *WordPerfect* npr. ponujata izbiro oblike pri shranjevanju besedila; pri nastavitvi opcije `format` torej izberemo obliko `ASCII Text`).

EDT pokličemo s simbolom **ed** in kot parameter dodamo ime datoteke, ki jo želimo urejati:

$ **ed** *besedilo.txt*

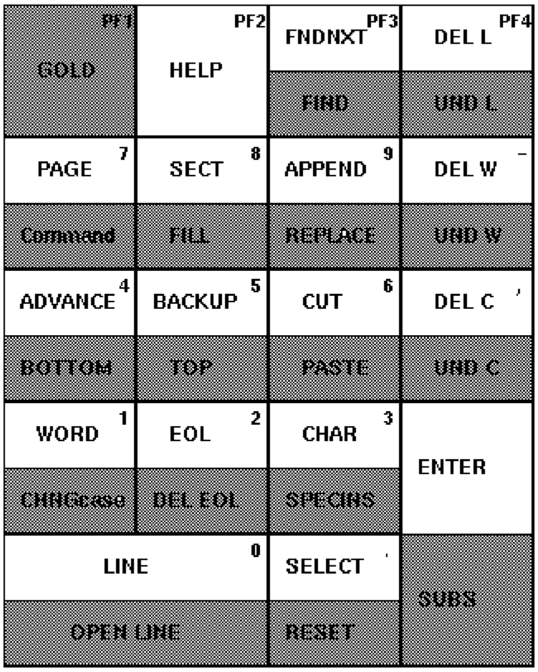
Če datoteka s tem imenom ne obstaja, jo *EDT* naredi. Najosnovnejše delo je enako preprosto kot v kakem drugem zaslonskem urejevalniku: besedilo preprosto tipkamo, uporabljamo tipko [Return] za prelom vrstice in tipko [Delete] za brisanje znaka pred kazalcem. S pomočjo puščic se pomikamo v vse smeri po besedilu. Vtipkano besedilo se vriva na mestu kazalca. Delo lahko končamo s [Ctrl–Z], nakar se pojavi poziv $*$ in čaka ukazov. Od tod dokončno izstopimo z **exit**, če želimo besedilo/spremembe shraniti in **quit**, če tega ne želimo. V prvem primeru se tvori nova verzija datoteke, stara pa se ohrani. Ker se tako ob vsakem shranjevanju sprememb ohrani stara verzija, je potrebno redno brisanje tistih, ki jih ne potrebujemo več (recimo z ukazom **purge**).

Vedeti je treba, da tu [Ctrl–Z] še ne pomeni izstopa iz urejevalnika, temveč le izstop v tako imenovani vrstični način dela, ki ima svoj nabor ukazov. Eden od koristnih ukazov (poleg **help**, ki nam pove še za preostale ukaze) je na primer **include** *nekaj.txt*, ki nam na trenutni položaj v urejevalniku vrine zunanjo datoteko *nekaj.txt*. Vendar pa so nekateri od teh ukazov prilagojeni vrstičnemu načinu dela. Nazaj v zaslonski način urejanja pridemo z ukazom **change** (zadostuje tudi **c**).

3.3.2 Urejanje besedila v *EDT* s pomočjo stranske tipkovnice

Načeloma je mogoče shajati z minimalnim znanjem iz prejšnjega razdelka, kar je koristna tolažba pri večini zaslonskih urejevalnikov. Seveda pa je tudi za nezahtevne koristno poznavanje vsaj še nekaterih ukazov, ki nam delo olajšajo – npr. brisanje celih vrstic in besed, izrezovanje in premeščanje kosov besedila, iskanje niza v besedilu, vključevanje zunanjih datotek v besedilo in podobno.

Pri tem lahko naletimo na dodatno težavo. *EDT* ima namreč večino osnovnih ukazov za delo z besedilom razporejenih na stranski (numerični) tipkovnici, kot je to prikazano na sliki 3.1.



Slika 3.1: Razporeditev funkcij urejevalnika *EDT* na stranski tipkovnici. V desnem zgornjem vogalu tipke je njena oznaka, kot jo pozna terminal VT100.

Oznake na tipkovnici so take, kot pri terminalu tipa VT100 (tudi VT220, VT320 in VT420 – gl. tudi sliko 2.2). Takšna razporeditev ukazov je sicer prav udobna – ko se je navadimo. Težava je v tem, da ta tipkovnica ni čisto taka, kot smo je vajeni na osebnih računalnikih. Kadar torej delamo prek modema in imamo pred seboj namesto standardnega terminala svoj osebni računalnik, ki oponaša terminal, moramo vedeti, katere tipke dejanske tipkovnice na našem PCju ustrezajo posameznim tipkam *EDTjeve* stranske tipkovnice, to pa je odvisno od konkretne tipkovnice in terminalskega emulatorja.[9] Pri dobrih emulatorjih lahko z ne preveč truda tudi sami spreminjamo to preslikavo.

Kakorkoli že, ko ugotovimo (če ne gre drugače, s pogumnim poskušanjem), katere tipke na naši tipkovnici ustrezajo tipkam PF1, PF2, PF3 in PF4, smo bolj ali manj na konju. Te tipke se (navadno) držijo skupaj, žal pa to niso vedno zgornje štiri tipke numerične tipkovnice, medtem ko so tipke 0 — 9 na numerični tipkovnici skoraj gotovo "ta prave". Možne alternativne kandidatke za PF1 — PF4 so funkcijske tipke F1 — F4.

Da ne bo zmede s poimenovanjem tipk, se domenimo, da bo vsak ukaz za *EDT* v nadaljnjem besedilu podan hkrati z oznako ustrezne tipke (ali kombinacije tipk) na standardni tipkovnici terminala tipa VT100. Oznaka PF1 5 – <TOP> v našem besedilu bo torej pomenila, da moramo pritisniti (zaporedoma) tipki, ki sta na standardni (VT100) numerični tipkovnici označeni s PF1 in 5 (na sliki levo), kar ustreza ukazu <TOP> v urejevalniku besedil *EDT* (gl. sliko 3.1) – oznaka za pomišljajem je torej le informativna in je ne tipkamo. Če imamo srečo s tipkovnico in terminalskim emulatorjem, bodo tipke, ki so na našem računalniku na istem mestu, delovale enako.

Ko enkrat odkrijemo tipko PF2 – <HELP>, se nam ob pritisku nanjo nariše numerična tipkovnica z oznakami ukazov v *EDT*, enaka tisti na sliki 3.1.[10]. Zdaj lahko s pritiskom na poljubno tipko (kombinacijo tipk) na tipkovnici odkrijemo njeno funkcijo in tako preverimo njihovo delovanje. Kratki opisi so nam pri tem zelo v pomoč.

Predstavimo na hitro nekaj najpogostejših ukazov, ki jih podpira numerična tipkovnica, skladno z dogovorjenim poimenovanjem.

PF2 – <HELP>
: Nariše razporeditev funkcijskih tipk v *EDT*. S pritiskom na katerokoli tipko na tipkovnici pa lahko ugotovimo njeno dejansko funkcijo. Pomoč zapustimo s pritiskom na tipko za presledek.

PF4 – <DEL L>
: Zbriše tekočo vrstico od kazalca naprej, vključno z znakom za novo vrsto.

PF1 **PF4** – <UND L>
: Vrne zadnjo zbrisano vrstico na mesto kazalca (uporabno za kopiranje vrstice – tudi večkratno).

– – <DEL W>
: Zbriše tekočo besedo (ponavadi do presledka) od kazalca naprej[11].

[9] To je program, ki posnema terminal – gl. razdelek 2.2.3.

[10] Zato dvojne oznake. Na PC vam PF2 nič ne pomeni, vedeli pa boste, kje mora ležati tipka za ukaz <HELP>

[11] Če je na vaši tipkovnici na mestu VT100 tipk – in ' le ena tipka, kot na sliki, je njena funkcija verjetno <DEL W>/<UND W>.

PF1 **–** – <UND W>
Vrne zadnjo zbrisano besedo na mesto kazalca.

4 – <ADVANCE>
Smer premikanja in iskanja je *naprej* (navzdol).

5 – <BACKUP>
Smer premikanja in iskanja je *nazaj* (navzgor).

PF1 **4** – <BOTTOM>
Skok na konec datoteke.

PF1 **5** – <TOP>
Skok na začetek datoteke.

0 – <LINE>
Skok na začetek vrstice (v smeri izbranega premikanja).

1 – <WORD>
Skok na začetek besede (v smeri izbranega premikanja).

. – <SELECT>
Začetek označevanja besedila. Od tu premikamo kazalec naprej ali nazaj, lahko tudi vpisujemo, označeni del pa se pobarva. Označeno besedilo lahko npr. izrežemo in po potrebi odložimo drugam (tudi večkrat) ali pa izpišemo na datoteko.

6 – <CUT>
Izreže označeno besedilo (ki izgine) in ga shrani v pomožni pomnilnik. Tam ostane, dokler ga ne popackamo z naslednjim izrezovanjem ali dokler ne zapustimo *EDT*.

PF1 **6** – <PASTE>
Odloži del besedila, ki je trenutno v pomožnem pomnilniku, na mesto kazalca.

PF1 **PF3** – <FIND>
Iskanje niza v besedilu. Na poziv `Search:` vpišemo iskani niz (brez Return) in pritisnemo 4 ali 5 (določimo smer iskanja).

PF3 – <FINDNEXT>
Poišče prvi (naslednji) primerek iskanega niza (določenega s PF1 PF3) v trenutni smeri premikanja. S ponovnimi pritiski na to tipko najdemo še naslednje pojavitve enakega niza.

PF1 **Enter** – <SUBSTITUTE>
(Preverite, če je tipka prava!) Nadomesti najdeni niz z besedilom iz pomožnega pomnilnika. Tipična uporaba: pritisnemo . – <SELECT>, vpišemo nadomestni tekst, ga izrežemo in shranimo s 6 – <CUT>; nato kot zgoraj poiščemo določen niz; potem pritisnemo PF1 Enter – <SUBSTITUTE> in ga zamenjamo. Še zadnja skrivnost: zaporedje PF1 *100* PF1 Enter oziroma <GOLD>*100*<SUBSTITUTE> opravi 100 zaporednih zamenjav (ali manj, če ne najde nobenega niza več v smeri premikanja). Številka je seveda lahko poljubna, trik pa je uporaben tudi v kombinaciji z drugimi ukazi.

Kombinacija tipk PF1 7 – <COMMAND> prikliče poziv `Command:`, na katerega lahko odgovorimo z različnimi ukazi. Naštejmo jih le nekaj:

`Command:` **help**
Pomaga odkriti, kakšne ukaze lahko uporabimo v odgovor na `Command:`

`Command:` **set wrap 70**
Ta ukaz določa, da kazalec preskoči na začetek nove vrstice, kadar pri pisanju presežemo (recimo) 70. kolono. Prihrani pritiskanje Return med pisanjem, predvsem uporabno za omejevanje širine besedila pri pisanju sporočil v elektronski pošti ali prispevkov v elektronske konference, kjer je zaradi berljivosti zaželena omejena širina besedila.

`Command:` **write** *bla.txt* **select**
Prepiše del besedila, ki smo ga pred tem označili, na datoteko *bla.txt*. Če namesto **select** napišemo **=paste**, bo na datoteko prepisal trenutno vsebino pomožnega pomnilnika *paste* (zadnji izrezani del besedila).

`Command:` **include** *ble.txt* Vsebino zunanje datoteke *ble.txt* prepiše na mesto kazalca v datoteko, ki jo urejamo.

`Command:` **exit**
Shrani urejano datoteko in konča. Enako kot **exit** v vrstičnem načinu (glej prejšnji razdelek).

`Command:` **quit**
Konča in zavrže urejano datoteko (ne shrani sprememb).

Te ukaze vedno zaključujemo s tipko Enter na numerični tipkovnici. Zopet: preverite, če je tipka prava s pomočjo PF2 - <HELP>!

Od posebnih tipk smo že prej omenili Ctrl–Y, s katero grobo prekinemo delo. Poleg tega, da se lahko naslednji hip rešimo z ukazom **continue**, nam ostane po vsaki grobi prekinitvi (tudi če bi šlo za okvaro na *Stenarju*) poleg stare verzije datoteke še ena datoteka z istim imenom kot "obdelovana", le da s podaljškom *.jou* (journal). Na njej je shranjeno zaporedje naših ukazov do prekinitve, ki se obnovi, če ob prvem naslednjem klicu *EDT* dodamo določilo **/recover**, torej

$ ed/rec *ime.txt*

če smo pred prekinitvijo urejali datoteko *ime.txt*. Pred očmi se nam na hitro odvrti vse, kar smo z datoteko počeli pred prekinitvijo (lahko manjka ukaz ali dva na koncu, če je šlo kaj narobe) in ko se "film" konča, lahko z delom nadaljujemo.

Druga posebna tipka je še Ctrl–W, s katero osvežimo sliko na zaslonu, kadar se na njem pojavijo motnje ali sporočila. Tudi kadar prekinemo (telefonsko) povezavo z računalnikom, na katerem delamo v *EDT*, ponavadi zadostuje, da po ponovni vzpostavitvi zveze in priključitvi odklopljenega procesa sliko le osvežimo.

3.4 Elektronska pošta – VMS *MAIL*

Elektronska pošta je osnovni način meduporabniške komunikacije preko računalniškega omrežja. Na računalnikih VAX nam branje, urejanje in pošiljanje sporočil omogoča program *MAIL*. Vse delo poteka v dialogu s programom, ki nam tudi hrani sporočila v posebni datoteki – pogosto ji pravimo kar *poštni nabiralnik* (*mailbox*).[12]

Program *MAIL* poženemo preprosto:

$ mail

Program se odzove s pozivom `MAIL>` in čaka ukazov. Ukaza, ki si ju kaže vedno najprej zapomniti, sta **exit** (oziroma Ctrl–Z) za zaključek dela in **quit** (oziroma Ctrl–Y), kadar gre kaj narobe (v drugem primeru ostane vsebina nabiralnika taka, kot je bila pred zagonom *MAIL*). Kot ponavadi nam **help** pomaga odkriti še ostale ukaze. Uporaba je enaka kot pri ukazu **help** v DCL.

3.4.1 Branje starih sporočil

Ko poženemo program *MAIL*, lahko preberemo vsa sporočila, ki jih hranimo v nabiralniku. Njihov seznam dobimo z ukazom **dir**[13]

`MAIL>` **dir**

```
                                                    MAIL
 # From                Date                 Subject

 1 IN%"Benjamin.Zwittni Cet, 14.apr.1994  test
 2 STENAR::TOMI         Cet, 14.apr.1994  RE: test
 3 IN%"postmaster@arnes Cet, 14.apr.1994  Undeliverable mail: SMTP
 4 STENAR::BONAC        Cet, 21.apr.1994  ALI SE DOBIMO ?
 5 STENAR::NETNEWS      Cet,  5.maj.1994  sestanek
 6 STENAR::TOMI         Cet,  5.maj.1994  RE: sestanek
 7 STENAR::BONAC        Ned,  8.maj.1994  sestanek --> sreda ?
```

[12]To med drugim pomeni, da sporočila niso shranjena na navadnih tekstovnih datotekah. Če torej hočemo sporočilo izpisati na tiskalnik, ga moramo najprej iz nabiralnika prepisati v navadno datoteko (glej ukaz **extract**).

[13]Načelo, da imajo enako "zveneči" ukazi v različnih okoljih (DCL in uporabniški programi) funkcije, ki so si v nekem smislu podobne, je na VMS precej razširjeno; to uporabnikom pogosto olajša delo z novimi programi.

Sporočila so oštevilčena in časovno urejena. Navedena sta tudi naslov pošiljatelja in *predmet* sporočila ali *zadeva* (subject).

Prvo sporočilo preberemo z ukazom **read** oziroma kar s pritiskom na tipko Return. Enako beremo nadaljnja sporočila, če jih je več. Če pa želimo prebrati na primer le tretje sporočilo, napišemo

`MAIL>` **read** 3

ali še enostavneje

`MAIL>` 3

Izpisano sporočilo je lahko videti na primer takšnole

```
    #1          Cet, 14.apr.1994 14:18:47                       MAIL
From:   IN%"Benjamin.Zwittnig@arnes.si"  "Benjamin Zwittnig"
To:     IN%"tomi.dolenc@arnes.si"
CC:
Subj:   test

Return-path: <Benjamin.Zwittnig@arnes.si>
Received: from kanin.arnes.si by arnes.si (PMDF V4.2-13 #5447) id
 <01HB62EBNVLC0005D5@arnes.si>; Thu, 14 Apr 1994 14:18:34 GMT +1
Received: from localhost.arnes.si by kanin.arnes.si with SMTP using
DNS (PP) id
 <07630-0@kanin.arnes.si>; Thu, 14 Apr 1994 14:18:26 +0200
Date: Thu, 14 Apr 1994 14:18:25 +0200
From: Benjamin Zwittnig <Benjamin.Zwittnig@arnes.si>
Subject: test
To: tomi.dolenc@arnes.si
Message-id: <01HB62EBO58I0005D5@arnes.si>
Content-transfer-encoding: 7BIT

Besedilo sporocila je le tale vrstica, pa se ta je brez sumnikov.
```

Prvih pet vrstic sestavlja *glavo* sporočila (header), ki je značilna za program *MAIL*. Poleg datuma in naslova pošiljatelja (`From:`) vsebuje še naš naslov (`To:`), predmet sporočila (`Subj:`). Če je bilo sporočilo poslano v vednost tudi na druge naslove, so ti našteti v posebnem polju (`CC:`).

Pri sporočilih, ki jih pošiljamo le v okviru slovenskega DECneta[14], sledi tej glavi takoj besedilo sporočila. Kadar pa potuje sporočilo izven slovenskega DECneta (npr. preko Interneta), je v sporočilo na začetku, kot v gornjem primeru, vključena še dodatna informacija o njegovi poti. Za to informacijo se ponavadi ne menimo, saj so vsi bistveni podatki (kdo, komu, kdaj) že v osnovni glavi.

Na dnu izpisa je besedilo sporočila, ki ga v našem primeru ni prav veliko.

3.4.2 Branje novih sporočil

Če nas v nabiralniku čaka še neprebrano sporočilo, nas *MAIL* ob zagonu obvesti

```
You have 1 new message
```

oziroma z ustreznim besedilom v primeru več takih sporočil. Če takoj zatem pritisnemo Return, se bo prikazalo (prvo) neprebrano sporočilo. *MAIL* namreč vsa še neprebrana sporočila hrani v posebni "mapi" ali "predalu" z oznako *NEWMAIL*[15], ki jo bomo opazili v desnem zgornjem kotu ekrana. Program privzema, da želimo najprej pregledati to mapo. Ukaz **dir** nam torej sedaj kaže le neprebrana sporočila. Ukaz **read** oziroma tipko Return seveda uporabljamo za branje na enak način kakor prej. Prebrana sporočila se bodo pri naslednjem klicu programa *MAIL* znašla na kupčku z ostalimi že prebranimi sporočili.

Če si želimo med pregledovanjem novih sporočil pogledati pošto, ki smo jo prebirali med prejšnjimi klici programa *MAIL*, si predal stare pošte v nabiralniku odpremo z ukazom

`MAIL>` **select** *mail*

Kadar pa se sredi dela s programom *MAIL* prikaže obvestilo, da smo pravkar dobili novo sporočilo, le-tega lahko preberemo z ukazom **read/new**.

3.4.3 Zapis sporočila na navadno tekstovno datoteko

Omenili smo že, da poštni nabiralnik na sistemih VAX/VMS ni navadna tekstovna datoteka. Če bi radi kakšno sporočilo prenesli na svoj osebni računalnik, da bi ga tam izpisali na tiskalniku ali ga kako drugače obdelovali, ga je treba najprej iz nabiralnika spraviti na "navadno" (ASCII) datoteko. Sporočilo, ki ga trenutno beremo, lahko shranimo na datoteko z ukazom

`MAIL>` **extract** *pismo.txt*

[14] V grobem je to omrežje nekaj sto računalnikov tipa VAX v Sloveniji.

[15] Več o tem v razdelku "Razporejanje sporočil v mape".

Na datoteko *pismo.txt* se bo prepisalo celotno sporočilo z glavo vred. Če bi radi prepisali le besedilo sporočila, uporabimo ukaz z določilom **extract/noheader**. Žal v primeru, da je sporočilo prišlo prek meja slovenskega DECneta, to določilo odstrani le tisti prvi del glave, ki jo je dodal *MAIL*.

3.4.4 Pošiljanje sporočil

Če je to naše prvo srečanje s programom *MAIL*, najbrž še nismo prejeli nobene pošte, ki bi jo lahko prebrali. Premostimo to težavo tako kot Pika Nogavička: napišimo si pismo sami. Začnemo tako, da v programu *MAIL* napišemo ukaz **mail** ali **send**:

`MAIL>` **send**

Takoj zatem nas program s pozivom `To:` vpraša, komu želimo poslati sporočilo. Ker ga pošiljamo sebi, je odgovor kar naše *uporabniško ime*. Vzemimo, da je to ime *Pika* (kot vedno ni pomembno ali pišemo velike ali male črke). Običajno pošiljamo pošto uporabnikom na drugih računalnikih – kako je potrebno v takih primerih napisati naslov, bomo pojasnili z zadnjem razdelku tega poglavja.

Če imamo tako izbrano nastavitev[16], nas na tem mestu program s pozivom `CC:` vpraša, komu razen naslovniku želimo sporočilo poslati v vednost. Zatem sledi še vprašanje `Subj:`, kamor vpišemo predmet sporočila.

Ko smo tako oblikovali glavo sporočila, se aktivira urejevalnik *EDT*, v katerem oblikujemo besedilo. Spomnimo se, da pri tem lahko vanj vključimo tudi kakšno zunanjo datoteko z ukazom PF1 7 **include** *zunanja.txt* (glej poglavje o urejevalniku *EDT*).

Sporočilo se odpošlje takoj po izstopu iz *EDT*, razen če smo ga zapustili s **quit**.

Pošiljanje vnaprej pripravljenih datotek

Če imamo sporočilo že pripravljeno v datoteki, npr. *pismo.txt*, ga odpošljemo s

`MAIL>` **send** *pismo.txt*

in šele potem vpišemo naslovnika in predmet sporočila. Tudi v tem primeru se znajdemo v urejevalniku *EDT*, da lahko sporočilo po potrebi dopolnimo. Ko smo z njim zadovoljni, zapustimo *EDT* (na primer s Ctrl–Z in **exit**), in pismo odpotuje.

[16] Ukaz **set CC** v *MAIL*.

Posredovanje sporočil

Včasih želimo poslati prejeto sporočilo naprej naslednjemu naslovniku. Sporočilo, ki ga želimo tako posredovati naprej, najprej prikličemo na zaslon in nato uporabimo ukaz **forward**, ki zahteva od nas odgovore na vprašanje `To:`, `CC:` in `Subject:`. Potem se sporočilo naloži še v urejevalnik *EDT*, kjer ga lahko dopolnimo, preden ga pošljemo naprej. Kadar sporočilu ne želimo ničesar dodajati, lahko uporabimo ukaz z določilom **forward/noedit**.

Pošiljanje binarnih datotek

Rekli smo že, da po elektronski pošti potujejo datoteke v ASCII obliki. Kadar želimo poslati binarno datoteko, jo je torej treba najprej kodirati v ASCII. Obstaja več vrst kodiranja in ustreznih programov. Prejemnik mora kodirano sporočilo seveda dekodirati s programom, ki pozna uporabljeno kodirno metodo. Včasih to zna že sam program za branje elektronske pošte.[17] Zelo razširjeno – iz zgodovinskih razlogov – na Internetu je npr. kodiranje z *uuencode*. Na *Stenarju* kličemo ta program s simbolom **uue**. Ko tako datoteko prejmemo po pošti, pa jo je seveda treba dekodirati z enako metodo (na *Stenarju* simbol **uud**).

Za primer vzemimo, da bi radi nekomu poslali datoteko *slika.gif*. Najprej jo moramo torej kodirati v ASCII z ukazom (seveda ne v programu *MAIL*)

$ **uue** *slika.gif*

Kot rezultat dobimo novo datoteko z imenom *slika.uue*. To datoteko lahko pošljemo kar z

$ **mail** *slika.uue*

To je krajša oblika pošiljanja datotek, kadar je to edino, kar želimo opraviti s programom *MAIL*. Program nas vpraša le še po naslovu in ne nalaga datoteke v urejevalnik besedil.

Ko tako sporočilo prejmemo, se ne zmenimo za njegovo nečitljivo vsebino, pač pa ga z ukazom

`MAIL>` **extract/noheader** *slika.uue*

shranimo v datoteko s podaljškom *.uue*. Nato izstopimo iz programa *MAIL* in datoteko dekodiramo v originalno binarno obliko z ukazom

$ **uud** *slika.uue*

[17] Naš program tega ne zna, obstaja pa na *Stenarju* še drug program za pošiljanje pošte, ki že sam pozna nekaj standardnih ustreznih kodiranj.

3.4.5 Odgovarjanje na sporočila

Če smo pismo res poslali sebi (naše uporabniško ime mora biti v tem primeru "PIKA", sicer bo sporočilo dobila kdove katera Pika, ali pa se bo *MAIL* pritožil, da je naslovnik neznan) in če delamo na računalniku *Stenar*, dobimo obvestilo:

```
New mail on node STENAR by STENAR::PIKA
```

Svežo pošto med delom s programom *MAIL* preberemo z ukazom **read/new**:

`MAIL>` **read/new**

```
    #1          Pet, 1.apr.1994  03:09:18               NEWMAIL
From:   STENAR::PIKA         "Pika Nogavicka"
To:     PIKA
Subj:   Moje prvo pismo

 Mja luba Pika, ...
```

V zgornjem primeru je majhna prevara: računalnik seveda ne more iz uporabniškega imena "PIKA" ugotoviti, da se pošiljatelj piše Nogavička. Ker se spodobi, da je v glavi pisma, ki ga pošiljamo, tudi (vsaj) naše polno ime, povejmo, kako se to uredi enkrat za vselej:

`MAIL>` **set personal_name** *"Pika Nogavicka"*

Poleg imena lahko seveda navedemo tudi druge koristne podatke, kot je telefonska številka ali ime organizacije, kjer smo zaposleni.

Za uporabnike na *Stenarju* velja, da je ta informacija že samodejno vključena in si jo lahko ogledamo z ukazom:

`MAIL>` **show personal_name**

Kako zdaj prijaznemu pošiljatelju odpisati? Na sporočilo, ki ga trenutno beremo, lahko takoj odgovorimo z ukazom

`MAIL>` **reply**

V tem primeru je naslovnik kar pošiljatelj tekočega sporočila, predmet pa enak predmetu njegovega sporočila z dodano predpono "RE:". Tako se brez spraševanja znajdemo v urejevalniku besedil, kamor je že posrkana vsebina sporočila, na katerega odgovarjamo. To nam olajša odgovor, posamezne vrstice sporočila lahko obdržimo v odgovoru kot citate (označimo jih recimo z znakom '>' v prvi koloni), ostale pa zbrišemo.

Če se želimo z odgovorom v miru potruditi, je postopek malo daljši: sporočilo z ukazom **extract** prepišemo na datoteko, ki jo potem obdelamo z urejevalnikom besedila. Če nas pri tem ovira šibko poznavanje urejevalnika *EDT*, prenesemo datoteko na svoj računalnik in jo obdelamo tam. Ne smemo pozabiti, da mora biti besedilo brez posebnih znakov (npr. za poravnavanje, poudarjanje itd.)

Nato v *MAIL* poiščemo in pokličemo sporočilo, na katerega želimo odgovoriti, uporabimo **reply** in v urejevalniku *EDT* z ukazom **include** (glej poglavje o *EDT*) v odgovor vključimo pripravljeno datoteko, preneseno nazaj na *Stenarja* s svojega računalnika.

Pripravljeno datoteko lahko v odgovor pošljemo seveda tudi z ukazom **send**, kot smo opisali prej. V tem primeru moramo seveda navesti naslov, pred predmet pa lahko pritaknemo "RE:".

3.4.6 Brisanje sporočil

Če prebranega sporočila ne potrebujemo več, ga lahko zbrišemo z

`MAIL>` **delete**

Kot pri **read** pa lahko tudi ukazu **delete** dodamo številko sporočila, ki ga želimo zbrisati.

V naslednjem razdelku se bomo naučili, kako lahko po nesreči zbrisano sporočilo še rešimo.

3.4.7 Razporejanje sporočil v mape

Sporočila, ki jih hranimo v nabiralniku (mailbox), so razdeljena po *mapah* (folders). Osnovne mape, ki jih ustvari sam program, so

mail	—	glavna mapa z že prebranimi sporočili
newmail	—	mapa z neprebranimi sporočili (torej ni vedno prisotna)
wastebasket	—	koš, ki vsebuje pravkar zavržena sporočila

Kadar so v nabiralniku neprebrana sporočila, je ob zagonu *MAIL* privzeta mapa *Newmail*, sicer pa glavna mapa. Novo mapo ustvarimo tako, da vanjo z ukazom **move** odložimo sporočilo, kot bomo videli malo kasneje. Mapo, ki jo želimo pregledovati, izberemo z ukazom **select**, sicer pa je delo enako, kot smo ga vajeni. Sporočila beremo, pišemo in nanje odgovarjamo, ne glede na to, v kateri mapi se nahajamo.

Iz povedanega je razvidno, da lahko pravkar zbrisano sporočilo še "poberemo iz koša", ki se sicer prazni ob izstopu iz programa *MAIL*. Najprej se "preselimo" v mapo *wastebasket*:

`MAIL>` **select** *wastebasket*

Vsebino mape lahko pregledamo z ukazom **dir**, ki izpiše predmete sporočil v tekoči mapi, skupaj z zaporednimi številkami:

```
                                                WASTEBASKET
  # From                  Date              Subject

  1 STENAR::PIKA          Pet, 1.apr.1994   Moje prvo pismo
```

Tekoče sporočilo lahko preselimo nazaj v mapo *mail* ali pa v kako drugo mapo:

`MAIL>` **move** *arhiv*

Če mapa *arhiv* ne obstaja, nas *MAIL* vpraša, ali naj jo ustvari. Če v kaki mapi pobrišemo vsa sporočila, mapa avtomatično izgine.

Seznam vseh map pa dobimo z ukazom

`MAIL>` **dir/folder**

3.4.8 Naslovi v elektronski pošti

Tako kot pri običajni papirnati pošti, je tudi pri elektronski pošti za uspešno dostavo pisma treba poznati pravilen naslov prejemnika. Tega lahko dobimo na različne načine: z vizitko, po pošti, od znancev, v reviji, v elektronskem imeniku ...

Za razliko od običajne papirnate pošte, kjer so naslovi standardizirani, pa lahko pri uporabi sistemov za elektronsko pošto srečamo mnogo različnih oblik naslovov. Ogledali si bomo le tiste, s katerimi se uporabnik na *Stenarju* najpogosteje sreča.

1. Naslov uporabnika na istem računalniku je kar njegovo uporabniško ime, kot smo že videli v dosedanjih zgledih:

   PIKA

2. Če je naslovnik v slovenskem omrežju DECnet, je naslov sestavljen iz imen računalnika in uporabnika, ki ju ločita dve dvopičji:

   STENAR::PIKA

3. Naslovi za mednarodno elektronsko pošto so nekoliko bolj zapleteni, ker lahko nastopajo v več različnih oblikah:

   - **Oblika RFC-822** je sestavljena iz dveh delov: iz imena uporabnika in hierarhičnega naslova, ki odraža organiziranost po državah in organizacijah. Oba dela sta ločena z znakom "@".

     Pika.Nogavicka@guest.arnes.si

   - **Oblika X.400** je sestavljena iz večih delov, ki so med seboj ločeni z ločilom, ki je lahko ";" ali "/":

     S=Pika;G=Nogavicka;OU=guest;O=arnes;P=ac;A=mail;C=si

   V obeh primerih gre dejansko za isti naslov, ki je zapisan v dveh različnih oblikah.

Programi, ki jih uporabljamo za pošiljanje in prejemanje elektronske pošte nikoli ne poznajo vseh možnih oblik, v katerih naslovi nastopajo. Program *MAIL* pozna le prvi dve obliki ter obliko RFC-822. Pri pošiljanju pošte Piki Nogavički z uporabniškim imenom *PIKA* na računalniku *Stenar* bi torej v polje "To:" morali v gornjih treh primerih napisati:

1. PIKA
2. STENAR::PIKA
3. IN%"Pika.Nogavicka@guest.arnes.si"

Vidimo, da v prvih dveh primerih napišemo kar sam naslov, v tretjem pa je naslov uklenjen še v narekovaja, pred njim pa stoji *IN%*. To je vse, kar si je treba zapomniti, kadar imamo v rokah naslov v obliki RFC-822. Na srečo je ta oblika v praksi najpogostejša.

Če smo od nekoga dobili zgolj naslov v obliki X.400, ne pa tudi v obliki RFC-822, ga pri pošiljanju pošte s programom *MAIL* ne moremo uporabiti neposredno. Najprej ga moramo pretvoriti v obliko RFC-822. To lahko naredimo s pomočjo posebnega pretvornika naslovov. ARNESov pretvornik naslovov med oblikama X.400 in RFC-822 lahko uporabljamo:

- Interaktivno, tako, da se prijavimo na uporabniško ime *addrconv* na računalnik z imenom:
  - Internet: addrconv.arnes.si
  - DECnet: KANIN
  - Sipax.25: 1611000087

  nakar sledimo navodilom programa za pretvorbo. Ta pretvornik najdemo tudi na straneh ARNESovega WWW strežnika, kjer je uporaba še enostavnejša.
- Preko elektronske pošte, tako, da pošljemo na naslov
  - addrconv@arnes.si
  - S=addrconv;O=arnes;P=ac;A=mail;C=si

  elektronsko pošto, ki vsebuje v vsaki vrstici po en naslov, za katerega želimo prevedbo.

Da bi drugim prihranili težave pri pošiljanju elektronske pošte in da bi bili čim lažje dosegljivi, vedno sporočajmo svoj elektronski naslov v vseh oblikah! Npr:

X.400 : S=Pika;G=Nogavicka;OU=guest;O=arnes;P=ac;A=mail;C=si
RFC-822 : Pika.Nogavicka@guest.arnes.si
DECnet : STENAR::PIKA

3.4.9 Kaj v primeru težav?

Če pri pošiljanju elektronske pošte pride do težav, ki jih ne znate sami razrešiti, lahko za pomoč povprašate sodelavca ali prijatelja oz. osebo, ki je v vaši organizaciji zadolžena za elektronsko pošto ali pa ima na tem področju največ izkušenj. Če še to ne bo pomagalo, in imate uporabniško ime na ARNESovem računalniku, pošljite čimboljši opis vaših težav na naslov

X.400 : S=postmaster;O=arnes;P=ac;A=mail;C=si
RFC-822 : postmaster@arnes.si
DECnet : POSTMASTER

Opis naj vsebuje naslednje podatke:

- Naslov, na katerega poskušate poslati pošto.
- Kdaj ste zadnjič neuspešno poskusili.
- Kaj se je zgodilo ob tem poskusu. Če ste dobili sporočilo o zavrnitvi oz. napaki, ga priložite (originalno vsebino, ki ste jo napisali vi, pobrišite).
- Ste na ta naslov pošto že kdaj uspešno poslali? Če ste, kdaj?
- Ste s tega naslova pošto že kdaj dobili? Če ste, priložite glavo ("solato", ki se nahaja pred začetkom vsebine) dobljenega sporočila.

S tem boste bistveno pospešili odkrivanje vzroka vaših težav.

3.5 Konferenčni sistem – VAX *NOTES*

NOTES je program, ki na računalnikih VAX, medsebojno povezanih v omrežje, omogoča skupinsko medsebojno dopisovanje, ki mu pravimo "elektronske konference" ali "oglasna deska". Predstavljamo si ga lahko kot velik elektronski časopis – ali še bolje, skupino časopisov, ki jih sestavljajo le pisma bralcev. Vsak časopis je posvečen neki splošni temi in na posamezen časopis smo lahko "naročeni" ali pa ne. V žargonu *NOTES* takemu časopisu ustreza *konferenca*. *NOTES* je program, ki omogoča branje in pisanje prispevkov v teh konferencah ter seveda "naročanje" in "odjavljanje" konferenc, razpoložljivih v določenem lokalnem omrežju. Najbolj znana sorodna sistema sta BBS ("Bulletin Board System"), ki je namenjen predvsem z modemi povezanim osebnim računalnikom (lokalno in mednarodno), in Usenet News, ki združuje različne računalnike, povezane v globalno omrežje.

VAX *NOTES* v Sloveniji povezuje nekaj sto vozlov v slovenskem omrežju DECnet. Teme posameznih konferenc segajo od večjega števila računalniških in različnih drugih strokovnih (med drugim *E-mail* in *Internet*) do "rekreativnih" (izleti, razni športi, hobiji) in debatnih (politične, verske, ...).

Program poženemo z ukazom

$ notes

V odgovor dobimo – če naš terminalski emulator le ni preslab ali napačno nastavljen (glej 2.2.3) – izrisano tabelo, ki nam kaže naročene konference, nad njo pa poziv

```
Notes>
```

ki pričakuje naših ukazov. Lahko bi začeli z najočitnejšim – **help** – in se kot ponavadi iz pomoči na VMS marsikaj naučili o ukazih v *NOTES*. Namesto tega uberemo lagodnejšo pot. Uporabniki na *Stenarju* lahko iz tabele razberejo, da so že naročeni na tri konference:

Conference_on_Conferences
Notes_on_Notes
Sample_Conference

V *Sample_Conference* se, če le razumemo angleško, na preprost način naučimo osnovne uporabe *NOTES* in še kaj zraven. Konferenco najprej odpremo z ukazom

`Notes>` **open** *Sample_Conference*

– zadošča razpoznavni del imena – in sledimo navodilom (na začetku potrebujemo le tipko Return). Pouči nas, da je vsaka konferenca razdeljena na ožje diskusijske teme, kjer so prispevki organizirani kot *odgovori* (*reply*) na zastavljeno *temo* (*topic*). Obe vrsti prispevkov imenujemo s skupnim imenom *notice* (notes). Vse notice so označene dvoštevilčno s *številka_teme.številka_odgovora*, tako npr. notica *7.3* predstavlja 3. "odgovor" na temo, zastavljeno v *7.0*. Pomensko gre seveda lahko za odgovor na "odgovor" *7.2*. S to oznako se v ukazih sklicujemo na posamezne notice.

Preglejmo nekaj osnovnih ukazov:

7.3
Prikaže notico *7.3*.

Return
Bere naprej (isto notico ali naslednji odgovor).

Enter
Enako kot Return, ko zmanjka odgovorov, prebere naslednjo (po času nastanka) notico.

reply
Odgovori na tekočo notico.

write
Odpre novo temo (napišemo uvodno notico).

dir
Seznam tem (**dir/all** pokaže še vse odgovore).

next uns
Prebere naslednjo neprebrano notico po času nastanka.

Kot pri mnogih uporabniških programih pod VMS je tudi tu delo olajšano z uporabo stranske tipkovnice (tudi tu PF2 pomeni pomoč). Ni pa nenadomestljiva, saj lahko ukaze pišemo neposredno v ukazno vrstico v odgovor na poziv `Notes>`.

Ukaz **reply** nas postavi v urejevalnik besedila, kamor ponavadi posrka vsebino brane notice, da si lahko s citiranjem posameznih delov – pred citirane vrstice navadno vstavimo znak > v prvo kolono, ostale vrstice seveda pobrišemo – pomagamo pri odgovoru. Za uporabnike na *Stenarju* je privzet urejevalnik *EDT*, ki je za silo opisan v tem priročniku. Če vam ta urejevalnik ustreza, lahko preskočite temo v *Sample_Conference*, ki govori o izbiri in prilagajanju urejevalnika za *NOTES*.

Posamezno konferenco zapustimo s Ctrl–Z, potem pa lahko odpremo naslednjo.[18] Na primer *Conference_on_Conferences*, kjer so v temi *2.∗* zbrane skoraj vse konference, ki jih lahko naročimo. Začnemo torej z ukazom **dir** *2.**, da dobimo seznam odgovorov (naslovi konferenc). O vsaki je napisanih nekaj informativnih stavkov, ki jih lahko preberemo z ukazom **read** oziroma z navedbo številke notice, ali pa beremo vse po vrsti, uporabljajoč tipko Return ali Enter. Če ugotovimo, da nas kaka konferenca zanima, ubogajmo navodilo: brez obotavljanja pritisnemo tipko 7 na stranski tipkovnici ali vtipkamo **select**, in konferenca, o kateri trenutno beremo v *2.∗*, bo naša.

Notes_on_Notes, tretja "prednaročena" konferenca, govori o uporabi, problemih in bontonu na *NOTES*. Ne bo nam škodilo, če si jo bomo ogledali, čeprav je večina prispevkov že starejših.

Ko zapustimo konferenco s Ctrl–Z, lahko izstopimo še iz *NOTES* s ponovnim pritiskom na Ctrl–Z.

[18] Če odpremo novo konferenco, se prejšnja samodejno zapre.

4.

Uporaba storitev omrežja Internet

4.1 Uvod

ARNES je del največjega svetovnega omrežja Internet, ki ga sestavlja množica med seboj povezanih lokalnih in prostranih omrežij. Zato z vključitvijo v ARNES poleg možnosti pošiljanja in sprejemanja elektronske pošte dobimo tudi dostop do vseh omrežnih storitev, ki jih omogoča Internet. Predvsem gre za pregledovanje in dostop do različnih informacij (dokumentov, programske opreme...), ki so shranjene v ogromnem informacijskem prostoru svetovnega omrežja.

Naštejmo nekaj najpomembnejših informacijskih storitev v Internetu:

- delo na oddaljenih računalnikih (Telnet)
- prenos datotek z oddaljenih računalnikov (FTP)
- iskanje datotek v javnih arhivih na oddaljenih računalnikih (Archie)
- izmenjava informacij v elektronskih konferencah (USENET News)
- preprosto preiskovanje omrežja in njegovih podatkovnih baz, dostop do različnih vrst informacij, ne glede na njihovo fizično lokacijo (Gopher, WWW, WAIS)

Telnet in FTP sta osnovni orodji za to, kar (razen dopisovanja) na omrežju pravzaprav želimo početi: prvi omogoča dostop do oddaljenih računalnikov in neposredno delo z njimi, drugi pa je protokol za prenos datotek, ki smo jih našli na oddaljenem računalniku in ki bi jih radi prenesli na svojega (seveda tudi obratno, kar pa nas ta trenutek ne zanima).

Druge storitve, ki jih bomo omenjali, so nekoliko kompleksnejše. Njihov glavni namen je pomagati pri iskanju informacij, ki v tako obsežnem in raznolikem omrežju nikakor niso enotno urejene in katalogizirane v enem samem imeniku. Novejša orodja, kot sta Gopher in WWW, so vedno bolj univerzalna in vključujejo zbiranje in prikazovanje informacij iz različnih koncev sveta hkrati, bolj ali manj zapleteno poizvedovanje v podatkovnih bazah in prenos izbranih informacij (datotek, dokumentov) "domov". Pri tem se uporabniku ni treba ubadati z vzpostavljanjem povezav z različnimi računalniki v omrežju, niti mu ni treba vedeti, na katerem koncu sveta se določena datoteka ali računalnik nahaja; z njegovega stališča je videti, kot da vsa informacija prihaja z enega mesta.[1]

Internet je največji splet računalniških omrežij na svetu, zunaj tega spleta pa so še druga omrežja (BITNET, komercialna omrežja z omejenim dostopom, nacionalna omrežja, ...). Mnoga od teh so prek posebnih računalnikov (*gateway*), ki razumejo jezik obeh udeleženih omrežij, povezana z Internetom. Vendar ta povezava ne dopušča neposredne uporabe vseh storitev, ki jih nudi Internet ali posamezno omrežje, ponavadi pa prepušča elektronsko pošto. S pomočjo elektronske pošte lahko celo naročamo posamezne zgoraj omenjene storitve, tako na Internetu kot zunaj njega.

4.1.1 Strežniki in odjemalci

Delovanje večine omrežnih storitev je zasnovano na tako imenovanem modelu *odjemalec/strežnik* (angl. client/server). *Strežniki* so programi, specializirani za neko storitev, in tečejo na določenih računalnikih v omrežju. Njihova naloga je, da *odjemalcem* na zahtevo poiščejo informacije in jim postrežejo z njimi. *Odjemalec* je program, ki lahko teče na kateremkoli računalniku v omrežju in omogoča uporabniku dostop in delo s strežnikom. Pri tem lahko izkorišča posebnosti lokalnega računalnika. Če je vaš računalnik neposredno povezan z omrežjem, lahko na njem instalirate odjemalce za razne storitve, ki vam delo s strežniki olajšajo z uporabo prijaznih grafičnih vmesnikov (npr. v okolju Windows). Ti omogočajo tudi prikazovanje bolj eksotičnih oblik informacij, ki so dostopne v omrežju, recimo slik ali celo filmov.

Če pa vstopate v Internet z uporabniškega imena na gostiteljskem računalniku (npr. *Stenarju*), s katerim se pogovarjate prek modema, si ne morete pomagati z odjemalci, ki bi npr. izkoriščali zmožnosti okenskega okolja. Ker vaš osebni računalnik v pogovoru z oddaljenim posnema navaden alfanumerični terminal, se boste morali zadovoljiti s tistimi odjemalci na gostiteljskem računalniku, ki informacijo prikazujejo v obliki, primerni za take terminale. To sicer ne pomeni izgube bistvenih informacij,[2] razvajene pa bo motilo, da morajo namesto miške in menijev uporabljati tipke za pomik kazalca in vnašanje ukazov.

Če povzamemo, lahko poteka delo s storitvami Interneta, ki delujejo po modelu odjemalec/strežnik, na naslednje načine:

[1] Seveda nam v resnici mnogokrat pomaga, če vemo, kje se informacija nahaja (ta podatek nam je tudi vedno na voljo). S to poenostavljeno trditvijo želimo le poudariti preprostost uporabe takšnih orodij.

[2] Slik ne vidimo neposredno, lahko pa jih shranimo v datoteko in pogledamo na domačem računalniku.

Neposredno s strežnikom: Povežemo se z (najbližjim) računalnikom,[3] na katerem teče strežnik za določeno storitev, in delamo "neposredno" s strežnikom. Tak pristop je obremenjujoč za gostiteljski računalnik in za omrežje ter v splošnem manj udoben; vendar je v nekaterih primerih edini možen, poleg tega pa strežniki včasih ponujajo bogatejši izbor ukazov kot ustrezni odjemalci (npr. *Archie*).

Z uporabo odjemalca: Ta je lahko nameščen na poljubnem računalniku v Internetu. Imamo dve možnosti:

- Računalnik, na katerem delamo, je del omrežja Internet. Če na njem ni odjemalca za določeno storitev, si ga priskrbimo oziroma poiščemo v omrežju[4] in ga instaliramo (oziroma prosimo usposobljeno osebo, da to stori).
- Povežemo se z (najbližjim) računalnikom v Internetu, na katerem teče odjemalec za določeno storitev.

Ko odjemalca poženemo, mu lahko ob klicu povemo še ime strežnika, s katerim naj se pogovarja. Če tega ne storimo, vzpostavi program stik z vnaprej nastavljenim strežnikom (najbližjim, najboljšim, ali domačim[5]).

S pomočjo elektronske pošte: Mnogi strežniki se odzivajo na ukaze tudi po elektronski pošti. Na naslov strežnika z elektronsko pošto pošljemo zahteve in rezultate dobimo po enaki poti. Ta način je uporaben v posebnih primerih (če gre npr. za storitev na računalniku nekega drugega omrežja, izven Interneta), zato ga ne bomo posebej opisovali. Večkrat pa je moč zaslediti navodila, ki vsebujejo elektronski poštni naslov takšnega strežnika in natančen opis besedila, ki ga moramo poslati v sporočilu, da dobimo željene rezultate.

Večino storitev Interneta navadno uporabljamo s pomočjo odjemalcev, kar je tudi najboljši način. Ta priročnik je namenjen tistim, katerih računalnik ni neposredno povezan v Internet in zato uporabljajo terminalsko povezavo za delo na *Stenarju*. Ta je seveda povezan v Internet in na njem so nameščeni odjemalci za večino storitev, ki jih bomo obravnavali. Navodila za uporabo bomo podali sproti za vsako storitev. Ti odjemalci so namenjeni za delo z navadnimi alfanumeričnimi terminali, kakršno poteka tudi prek modemske povezave.

Vsi primeri v tem poglavju so narejeni na računalniku *Stenar*, zato pred ukazi, s katerimi kličemo posamezne programe, stoji VMSjev poziv $. Naj opomnimo, da je sicer uporaba storitev, ki jih opisujemo, zelo malo vezana na ta ali oni operacijski sistem. Programe kličemo bolj ali manj podobno na vseh vrstah računalnikov, njihovi interni ukazi pa so

[3]Kako to naredimo, je opisano v razdelku "Telnet: delo na oddaljenih računalnikih".

[4]Odjemalci so ponavadi javno dostopni programi.

[5]Če odjemalni program "privlečemo" npr. iz Amerike, se bo verjetno najprej "spomnil" na kak svoj domači strežnik.

bodisi specifični za storitev ali pa razmeroma univerzalno razpoznavni na Internetu.[6] Skoraj vedno si lahko pomagamo z ukazom **help** oziroma **?**.

Naslovi in imena računalnikov v Internetu

Da bi storitve Interneta znali uporabljati, je potrebno povedati vsaj nekaj besed o tem, kako je omrežje organizirano. Rekli smo že, da je Internet omrežje računalniških omrežij. V njem ima vsak računalnik natanko določeno oznako ali *naslov*. Naslov je sestavljen iz štirih števil, med seboj ločenih s pikami, recimo

193.2.1.68

Računalniki imajo poleg naslovov navadno tudi imena, ki jih ravno tako natanko določajo. Ime v splošnem sestavlja nekaj besed, ki so med seboj spet ločene s pikami, na primer

stenar.arnes.si

ki pripada računalniku z zgoraj zapisanim naslovom. Internet omogoča, da računalnike lahko kličemo po imenu ali po naslovu.

Ker si imena lažje zapomnimo in so v splošni uporabi, bomo odslej pozabljali na naslove. Imena so celo bolj zanesljiva, saj so nekatera vezana na določene storitve, ki jih računalnik opravlja. Tako lahko enemu računalniku oz. enemu naslovu pripada več imen. Za primer lahko vzamemo ime FTP strežnika na ARNESu:

ftp.arnes.si

Storitev se sčasoma lahko preseli (skupaj z imenom) na drug računalnik, ki ima tudi drug naslov, vendar uporabniku storitve tega ni treba vedeti.

Tu omenjamo številske naslove le zato, ker se kdaj pa kdaj le zgodi, da jih dobimo v roke namesto imena, in zato, ker uporaba imena kdaj pa kdaj iz različnih vzrokov ne deluje.

Imena so urejena hierarhično po nivojih, imenovanih *domene*, kar nam pomaga uganiti, kam spada kateri računalnik. Hierarhična stopnja raste proti desnemu delu imena, kar pomeni, da je prva beseda pogosto kar ime računalnika ali storitve, vmes je na primer oznaka organizacije, ki ji računalnik pripada, zadnja beseda pa je oznaka države ali pa ene od glavnih domen (*edu, com, gov, mil, org, net*), ki označujejo dejavnost.[7] Vrhnja domena za Slovenijo je *.si*, kar pomeni, da se imena računalnikov pri nas praviloma končajo s to

[6] Navadno bolj spominjajo na ukaze operacijskega sistema Unix, ki je najbolj pogost v omrežju; pogosto pa tudi razumejo več oblik ukazov za isto ali sorodno funkcijo.

[7] Po vrsti kratice za *educational, commercial, government, military, organizations, network*. Računalniki s tako vrhnjo domeno so pretežno iz ZDA, v Evropi je bolj v navadi oznaka države.

oznako.

Poglejmo si za primer še dve imeni:

archie.funet.fi
pegasus.law.columbia.edu

Za prvega bi zdaj že znali uganiti, da označuje strežnik za Archie nekje na Finskem. V ZDA namesto državne končnice *.us* pretežno uporabljajo oznake dejavnosti, zato drugo ime pripada računalniku iz izobraževalnega področja, ki se verjetno nahaja na univerzi Columbia v ZDA in ima najbrž opraviti s pravom. Prišepnimo, da se na njem nahaja knjižnični informacijski sistem pravne knjižnice, kmalu pa se bomo tudi naučili, kako priti do njega.

Bogastvo storitev na Internetu

Težko je na kratko opisati, kaj vse je mogoče najti na omrežju. Poleg možnosti komuniciranja na zares planetarni ravni ponuja Internet predvsem zakladnico informacij. V grobem lahko rečemo, da je moč najti vsaj nekaj podatkov – ponavadi pa veliko več, kot bi pričakovali – o *katerikoli* stvari, ki nam pride na misel (pa morda še marsikaj popolnoma neznanega). Te informacije so lahko preprosti dokumenti, članki, knjige, "pisma bralcev", cele zakladnice programov, slike, zvok ali celo filmi. Vse to je zbrano v raznih arhivih, pravih knjižnicah s katalogi, v bazah podatkov, na oglasnih deskah ali pa je kar "nametano naokrog". Pri tem ne smemo pozabiti še na storitve in informacije drugih omrežij, ki so dostopna prek Interneta vsaj prek elektronske pošte. Celotni Internet tako daje vtis velikanske, ne pretirano urejene knjižnice brez pravega gospodarja, ki jo obiskovalci uporabljajo tudi za družabna srečanja in izmenjavo sporočil.

Čeprav Internet pospešeno naseljujejo ponudniki različnih komercialnih storitev, je večina informacij še vedno zastonj. Potrebujemo pa dostop do Interneta in znanje, kako priti do določene informacije. Potem ko z več ali manj težavami zadostimo prvemu pogoju in se za silo naučimo uporabljati orodja na Internetu, pa ugotovimo, da šele iskanje informacij predstavlja pravi izziv, če želimo bogastvo virov na omrežju res dobro izkoristiti.

Zavedati se moramo, da je informacijski prostor računalniškega omrežja res velikanski. Šele ko nekaj časa na slepo raziskujemo omrežje, zaslutimo njegove razsežnosti, in zazdi se nam, da nekoliko bolje razumemo pojem neskončnosti. Omrežje sestavlja na tisoče in desettisoče računalnikov, za katerimi različne skupine in posamezniki skrbijo za delovanje omrežja in dodajajo nove podatke. Informacije se spreminjajo, selijo in izginjajo. Ker je centralno vodenje take "knjižnice" malodane nemogoče, tudi ne more biti katalogov dostopnih informacij, ki bi obsegali celotno omrežje. Obstajajo pa storitve, ki skušajo s povezovanjem različnih skladišč podatkov pokriti čimvečji del tega prostora. Vedno pogosteje lahko tudi v tiskanih publikacijah (revijah, knjigah) najdemo objavljene vstopne točke v omrežju, kjer lahko najdemo koristne informacije z izbranih področij.

Na omrežju se informacije o dostopnih datotekah in storitvah objavljajo v ustreznih "elektronskih novicah" in drugih v elektronskih publikacijah, zbirajo pa se v splošno znanih informacijskih centrih, v dokumentih FAQ (Frequently Asked Questions), RFC (Request for Comments) in elektronskih priročnikih (npr. "Big Dummy's Guide to the Internet").[8]

Novejša orodja za preiskovanje omrežja in dostop do podatkov (Gopher, WWW) tudi začetnika hitro pripeljejo do takih splošnih informacij. Dostop do takega orodja in nekaj radovednosti je vse, kar potrebujemo pri prvih raziskovalnih korakih.

Kako do njih?

Najprej moramo biti seveda priključeni na Internet (oziroma imeti dostop do njega), o čemer smo nekaj povedali že na začetku. Naučili smo se že tudi, kako pošiljamo elektronsko pošto. Željene podatke lahko torej dobimo tako, da jih naročimo po (elektronski) pošti. Pri tem nam pomagajo avtomatski poštni odzivniki – programi, ki tečejo na mnogih računalnikih in znajo odgovarjati na zahteve (za začetek lahko od njih zahtevamo navodila za uporabo). Kadar nimamo druge možnosti, lahko s pošiljanjem ukazov po elektronski pošti uporabljamo različne storitve, tudi take, ki so sicer zasnovane za interaktivno delo (npr. Gopher). Omenili smo že, da je to edina možnost uporabe storitev Interneta, če je naše omrežje z njim povezano le prek elektronske pošte.

Elektronska pošta je seveda le posredna pot. Omrežje nam omogoča, da se neposredno povežemo z oddaljenim računalnikom in na njem poganjamo programe ali pa prenašamo podatke z enega računalnika na drugega. Za to moramo poleg naslova oddaljenega računalnika v omrežju poznati še uporabniško ime in geslo, s katerim se prijavimo na ta računalnik. Tako se lahko s poljubnega računalnika na Internetu povežemo s katerimkoli računalnikom na svetu, ki je prav tako priključen na Internet, nato pa na njem delamo kot prek terminala ali pa z njega prenašamo podatke. To nam omogočata osnovni orodji na Internetu, programa *telnet* in *ftp*.

Za dostop do javnih podatkov na Internetu uporabljamo javna uporabniška imena in gesla. Tudi druga orodja, ki jih bomo spoznali in ki nam pomagajo iskati informacije na Internetu, uporabljajo isti mehanizem. Za neposredno delo z omrežjem je vselej potrebna prijava na kak oddaljen omrežni računalnik, le da včasih to namesto nas opravi ustrezna aplikacija.

Kako se bomo prijavili oddaljenemu računalniku, kjer so shranjene željene informacije, naj nas torej ne skrbi. Če so podatki namenjeni javni rabi, je tudi znano, kako se do njih pride. Pač pa se moramo pripraviti na to, da bomo pri sprehajanju po različnih računalnikih v omrežju naleteli na različna okolja, ki jih nismo povsem vajeni. Drugačni operacijski sistemi, drugačni ukazi, drugačni odgovori računalnika. A to še ni razlog za preplah. Večina

[8] Vsi pravkar navedeni dokumenti so na voljo tudi na ARNESovem FTP strežniku in zlahka dostopni prek vmesnika na *Stenarjevih* WWW straneh.

komunikacije z oddaljenimi računalniki poteka prek aplikacij, ki uporabljajo takšne nabore ukazov, da so razumljivi večini uporabnikov najbolj pogostih operacijskih sistemov.

Vseeno pa je dobro vedeti, kakšna so na primer imena datotek v različnih operacijskih sistemih, vsaj toliko, da razumemo izpis seznama datotek.

Kako so zgrajena imena datotek v sistemu VMS in kakšne podatke dobimo v izpisu, ki ga v tem sistemu povzroči ukaz **dir**, smo spoznali v prvem delu tega priročnika. Zdaj pa si poglejmo, kako razvozlamo izpis seznama datotek v sistemu Unix, ki je najpogostejši na računalnikih v Internetu. Seznam je videti na primer takle:

```
-rwxr-xr-x  1 619      daemon        790 Apr 13 10:24 .cache
-rw-r--r--  1 root     daemon         57 Mar 17 14:24 .opis
-rw-r--r--  1 619      daemon        523 Nov  9  1993 README
drwxrwxr-x  3 619      30            512 May  4 09:14 arnes
d--x--x--x  2 619      daemon        512 Nov  9  1993 bin
drwxr-xr-x  2 619      daemon        512 Nov  5  1993 dev
drwxr-xr-x  2 619      daemon        512 Apr 11 08:31 etc
-rw-r--r--  1 root     daemon     671671 Jun 11 22:02 ls-lR
drwxrwxr-x 10 619      21            512 Mar 24 08:24 software
```

Kaj moramo vedeti, da se znajdemo v takšnem izpisu? Ime datoteke je zapisano na skrajnem desnem robu. Na prvi pogled je format imena nekoliko svobodnejši kot v DOS ali VMS, saj imena niso vsa v standardni obliki *ime.tip* (v gornjem izpisu ni niti ena taka). Zelo pomembno je tudi, da Unix v imenih datotek *loči med velikimi in malimi črkami*[9]. To si moramo zapomniti in ime datoteke (npr. pri prenašanju) napisati natanko tako, kot ga vidimo v seznamu, upoštevajoč velike in male črke. Nasploh je pametno držati se tega pravila vedno, kadar imamo opraviti z neznanim sistemom. V nasprotnem primeru se nam vse prevečkrat zgodi, da računalnik ne najde datoteke, katere ime smo "čisto prav" napisali.

Tik pred imenom datoteke v izpisu vidimo datum kreiranja datoteke. Pred datumom je velikost datoteke v bytih (v gornjem primeru je najkrajša datoteka dolga 57 bytov, najdaljša pa več kot 600 Kb).

Za nas je zanimiv še prvi, skrajnji levi del vrstice v izpisu, ki ga sestavlja zaporedje desetih znakov – črk oz. pomišljajev, ki nadomeščajo prazen prostor. Prvi znak nam pove, ali gre za navadno datoteko (v tem primeru je prvi znak "**-**") ali pa za imenik (prvi znak je "**d**", za "directory"). Tako lahko v gornjem primeru razberemo, da je *README* "navadna" datoteka, *software* pa je imenik.

Preostalih devet znakov iz tega zaporedja opisuje zaščito datoteke, sestavljajo pa ga trojice znakov **r**, **w** in **x**, ki določajo načine dovoljenega dostopa (read, write, execute) različnim

[9]Enako velja tudi za ukaze v Unixu.

skupinam uporabnikov (user, group, world). Prazno mesto oz. pomišljaj pomeni, da ustrezen dostop ni dovoljen.

Kljub temu, da večino dela na oddaljenih računalnikih opravljamo prek raznih aplikacij in nam ni treba zares poznati tujih operacijskih sistemov, je včasih vseeno potrebno nekaj iznajdljivosti, da se prilagodimo novemu okolju. Pri križarjenju po omrežju se srečujemo z aplikacijami, ki so jih napisali različni ljudje, zato za delo z njimi ni enotnih pravil. Delno se lahko razlikujejo tudi ukazi med izvedbami istega programa na različnih računalnikih. Navodila ali ukazi so lahko tudi v kakšnem drugem jeziku, ne le v angleščini.

Splošna navodila so tale:

- Berite navodila, ki spremljajo aplikacije in uporabljajte pomoč, ki jo nudijo sami programi (vedno lahko poskusite z ukazi, kot sta **help** ali **?**). Ponavadi že kratki napotki povsem zadoščajo za osnovno delo.
- Poskušajte. Aplikacije, s katerimi se srečate v omrežju, so ponavadi dovolj robustne, zato ne škodi, če v zadregi poskusite uporabiti ukaze, ki bi po vašem mnenju lahko delovali. Malo verjetno je, da bi pri tem lahko kaj pokvarili – v najslabšem primeru je treba začeti znova.
- Na začetku si najprej zapomnite ukaz (ali ubežni znak), s katerim končate delo oziroma prekinete delovanje programa. Ta podatek lahko pogosto preberemo takoj po zagonu aplikacije.
- Vedno lahko koga vprašate. Najprej izvedenca v vašem okolju ali skrbnika sistema, na katerem delate (v našem primeru je to kar ARNES, kjer sprejemamo vprašanja na elektronski naslov *helpdesk@arnes.si*). Poleg tega vsebuje v večini primerov opis aplikacije tudi naslov kontaktne osebe, na katero se lahko obrnete v primeru težav. Spodobi pa se, da *pred* spraševanjem preberemo navodila, ki jih imamo pred nosom (bodisi v omari ali na računalniku).

Kdaj pa kdaj se nam zgodi, da ne moremo vzpostaviti zveze s storitvijo na kakem oddaljenem računalniku. Sporočilo, ki ga pri tem dobimo na zaslon, nam lahko izvor težave pojasni ali pa tudi ne. Možnosti, zakaj ne dobimo zveze z oddaljenim računalnikom, je lahko več:

- Najpogosteje je vzrok tipkarska napaka. Preverimo torej, ali smo naslov računalnika (in po potrebi vse, kar sodi zraven) pravilno napisali.
- Včasih računalnik, na katerem delamo, iz različnih vzrokov ne zna pravilno razvozlati sicer pravilnega imena oddaljenega računalnika. V tem primeru lahko namesto z imenom poskusimo z naslovom (številko), če ga poznamo.
- Lahko se zgodi, da oddaljeni računalnik "nima časa" (storitev je preobremenjena) ali začasno ne deluje. Preostane nam le, da poskusimo kasneje.

Če pa ne pomaga ne to ne ono ali pa se problemi ponavljajo, se obrnemo na administratorja našega sistema oz. lokalnega omrežja. Za uporabnike na *Stenarju* velja, naj pošljejo elektronsko sporočilo s čimbolj natančnim opisom težav na naslov HELPDESK (oz. enakovredno na naslov *helpdesk@arnes.si*).

4.1.2 Še beseda o pravilih vedenja na omrežju

Omrežje je skupna dobrina in uporabo storitev na njem moramo jemati kot privilegij, ne kot pravico. Zato velja upoštevati določena pravila vedenja. Mednje najprej sodi upoštevanje norm, ki jih predpisuje samo omrežje, ki ga uporabljate.

Na splošno velja, da omrežja ne smemo zlorabljati (nepotrebno obremenjevati kapacitete omrežja, poskušati vdirati v tuje računalnike).

Kadar uporabljamo storitve na omrežju, jih uporabljajmo tako, da omrežje čim manj obremenjujemo – izberemo strežnik, ki nam je najbližji, podatke, ki jih želimo prenesti na svoj računalnik, poiščemo na čim bližji lokaciji, zlasti požrešne operacije (npr. prenos večjih datotek) opravimo izven delovnih ur, ko so računalniki manj obremenjeni[10] – tako bo tudi naše delo hitrejše in udobnejše.

Varnost omrežja in pravila o dopustni uporabi omrežja

Sama zasnova Interneta je decentralizirana, zato tudi ni ene same organizacije za nadzor nad delovanjem mreže. Vsaka organizacija, ki dostop do Interneta omogoča, zato skrbi, da njen del mreže deluje neprekinjeno (in varno). Skrbnik omrežja, ki je del Interneta, se pri tem ravna po pravilih o dopustni uporabi omrežja. Ta pravila se lahko med seboj delno razlikujejo (npr. glede na zakone države), skupna pa so jim splošna načela, ki smo jih našteli že v uvodu.

Na Internet so priključene organizacije, ki opravljajo različne dejavnosti – od univerz, inštitutov, podjetij, do vladnih služb. Na teh računalnikih so shranjeni podatki vseh vrst. Ti so lahko javno dostopni, lahko pa so tajni in njih izguba lahko organizaciji povzroči večjo ali manjšo škodo, zato so računalniki praviloma zaščiteni pred neavtoriziranim dostopom preko mreže. Če pa niso, to ne pomeni nujno, da jih lahko uporabljamo (povedano z analogijo geslo/ključ: če imamo slabo ključavnico, ali pa smo pustili vrata stanovanja priprta, to ne pomeni, da lahko vstopi vsakdo).

[10] Pri tem moramo upoštevati seveda lokalni čas računalnika, na katerem storitev poganjamo.

Dostop do računalnikov na Internetu

Zvezo z drugim računalnikom lahko na Internetu vzpostavimo na več načinov. Nekatere od teh (WWW, anonimni FTP, Gopher, Archie) lahko uporabimo vedno, ker gre za javno dostopne servise. Pri običajnem *ftp* ali *telnet* dostopu pa moramo vedeti uporabniško ime in geslo na tujem računalniku (nekateri računalniki imajo javna uporabniška imena, recimo knjižnični sistemi).

Nedovoljeno pa je uporabljati tuja uporabniška imena, ali pa tudi poizkušati različne kombinacije imen in gesel. Taki poizkusi se obravnavajo kot vlomi. Skrbnik sistema, ki opazi te poskuse vlomov, bo najverjetneje (ko bo preveril Internet naslov povzročitelja) obvestil skrbnika sistema, s katerega so se ti poizkusi vršili. Le-ta pa mora potem poskrbeti, da do podobnih "motenj" ne prihaja več, kar je najenostavneje storiti tako, da se povzročitelju ukine uporabniško ime.

Tudi piratsko kopiranje avtorsko zaščitene programske opreme je kaznivo in se od leta 1995 v Sloveniji preganja po uradni dolžnosti.

Več o piratstvu, in ustreznih zakonih, ki obravnavajo elektronske komunikacije, lahko najdete na domačih straneh ARNESovega WWW strežnika (takoj ko poženete **www**), sicer pa je teh dokumentov dovolj na mnogih lokacijah Interneta.

4.2 Dostop do oddaljenih računalnikov in prenos datotek v Internetu

V tem razdelku bomo spoznali Telnet in FTP, temeljni orodji za delo na Internetu.

4.2.1 Telnet: delo na oddaljenih računalnikih v omrežju Interneta

Storitev Telnet omogoča, da se povežemo s katerimkoli računalnikom, ki je vključen v omrežje Internet. In če nam ta računalnik to dovoli, na njem tudi neposredno delamo. Poleg očitne, vendar majhne možnosti, da imamo na takem računalniku svoje uporabniško ime, ponujajo mnogi računalniki storitve, ki so dostopne prek javnih uporabniških imen, kjer gesla ne potrebujemo ali pa je javno poznano. Seveda pa je ta storitev ključnega pomena, kadar želimo npr. iz tujine prebirati pošto na svojem domačem sistemu. Če se npr. nahajate v Avstraliji in želite delati na *Stenarju*, vam ni treba plačati gromozanskega telefonskega računa za klic v Slovenijo. Potrebujete le kakršenkoli računalnik, ki ima dostop do Interneta, ter informacijo, kako se na njem uporablja Telnet.

Uporaba Telneta je zelo podobna npr. povezavi osebnega računalnika prek modema in komunikacijskega programa z gostiteljskim računalnikom. Razlika je v tem, da deluje znotraj omrežja Internet, je hitrejša, namesto telefonske številke pa programu povemo ime ali naslov oddaljenega računalnika. Na oddaljeni računalnik se moramo seveda prijaviti. Potem lahko delamo tako, kot da bi bila naša tipkovnica in zaslon neposredno povezana s tem računalnikom. To je v bistvu vsa funkcija Telneta.

Razen za delo na oddaljenem računalniku pod svojim uporabniškim imenom se Telnet najpogosteje uporablja za neposreden dostop do različnih storitev na Internetu: podatkovnih baz, informacijskih strežnikov, knjižničnih kataloških sistemov ali nasploh strežnikov za storitve, za katere nimamo na voljo odjemalca na svojem računalniku. Že v prejšnjem poglavju smo npr. omenili pretvornik elektronskih naslovov, do katerega lahko s *telnetom* dostopamo na naslov *addrconv.arnes.si* in se predstavimo z javnim uporabniškim imenom *addrconv*.

Potek dela

Najprej pokličemo *telnet* in vpišemo ime oddaljenega računalnika, s katerim želimo stopiti v stik:

$ telnet *ime_oddaljenega_računalnika*

Ko se zveza vzpostavi, oddaljeni računalnik zahteva od nas uporabniško ime in geslo.

Javna uporabniška imena najdemo v različnih publikacijah, katalogih, knjigah ali revijah in pa seveda v dokumentih, ki jih najdemo v arhivih omrežja.

Nekatera pogosta javna uporabniška imena so *info, opac, library, gopher, wais* in *www*.

Geslo v mnogih primerih ni potrebno; lahko je enako javnemu uporabniškemu imenu ali pa je javno objavljeno.

Nekatere storitve zahtevajo povezavo prav na določen *vhod* (angl. *port*), njegova številka pa je javno objavljena. V takem primeru podamo to številko z določilom **/port**, kot vidimo v naslednjem primeru[11]

$ telnet *madlab.sprl.umich.edu* **/port=3000**

kar nas poveže z "napovedovalcem vremena" na univerzi v Michiganu. V takih primerih pogosto ne potrebujemo niti uporabniškega imena. Napovedi orkanov na ameriški obali so za nas sicer nekoliko manj uporabne, je pa ta strežnik primeren za spoznavanje dokaj udobne storitve, zato ga lahko preizkusite – če le mogoče, seveda rajši izven (ameriškega!) delovnega časa.

Včasih nas računalnik, s katerim se povežemo, vpraša po tipu terminala. Ker skoraj vsi računalniki prepoznavajo terminal tipa VT100, naš komunikacijski program pa tudi zna oponašati vsaj tega, je nastavitev tipa VT100 kar varna izbira.

Ukaz **telnet** brez imena računalnika nas postavi v ukazni način s pozivom `TELNET>`. Tam imamo na voljo nekaj ukazov (vzpostavitev, prekinitev povezave, nastavitev parametrov delovanja, izhod iz *telneta*), ki so za doslej opisano preprosto uporabo manj pomembni. Pogledate si jih lahko tako, da na gornji poziv odgovorite z **?**, nato pa poskusite še **help** ali **help** *telnet_ukaz*.

Bolj važno je, da v ukazni način lahko pridemo z uporabo *ubežnega znaka* (escape character) kadarkoli med delom s programom *telnet* (recimo, kadar gre kaj narobe in se ne znamo izkopati iz težav na oddaljenem računalniku). Ta znak, ki ga ne smemo zamenjevati s tipko ESC, je odvisen od izvedbe *telneta* in program ponavadi nanj opozori ob začetku delovanja. Na *Stenarju* velja za *telnet* ubežni znak Ctrl-6.[12] Rešilno zaporedje znakov bi bilo torej npr. Ctrl-6 **q** (ubežni znak z ukazom za prekinitev dela).

Kako končamo delo na oddaljenem računalniku? To je pogosta težava, s katero se sreča začetnik, saj odgovor ni en sam. Spet velja, da moramo brati navodila, ki se prikažejo ob prijavi na oddaljeni sistem, ali si pomagati z ukazi za pomoč. Če na ta način ne najdemo napotka, kako zapustiti sistem, lahko poskusimo z enim od naslednjih ukazov: Ctrl-D,

[11] Pri uporabi ukaza **telnet** na nekaterih drugih sistemih (Unix) navedemo številko priključka na koncu ukaza, brez določila.

[12] V navodilih (**help telnet**) piše, da je ubežni znak Ctrl-^; v resnici gre za isti znak, saj v kombinaciji s tipko Ctrl ni razlike med "spodnjo" in "zgornjo" (Shift) funkcijo tipke.

Ctrl-Z, **exit, quit, logout, bye**. Kadar tudi to ne pomaga, lahko še vedno "pobegnemo" v ukazni način *telneta* z ubežnim znakom, kot je opisano zgoraj. Nato *telnet* zapustimo z ukazom **quit**. Kot absolutno zadnji izhod pa lahko izvajanje programa grobo prekinemo s Ctrl-Y.

Izpolnimo še obljubo iz uvodnega razdelka: knjižničnemu informacijskemu sistemu na strežniku *pegasus.law.columbia.edu* se prijavimo z uporabniškim imenom *pegasus*. Ostalo boste zmogli sami. Poskusite. Lahko pa prej preizkusite podoben strežnik v Sloveniji, *izum.izum.si*, javno ime *OPAC*. Po tem, ali bodo v slovenskem besedilu na zaslonu lepo vidni šumniki, boste spoznali kvaliteto posnemanja terminala svojega programa za komunikacijo.

4.2.2 FTP: prenos datotek v Internetu

FTP je ime storitve, ki omogoča prenos datotek med dvema računalnikoma v omrežju Internet, torej opravilo, za katerega uporabljamo znotraj istega računalnika ukaz, kot je **copy**.[13]

Ftp prav tako kot *telnet* vzpostavi zvezo z oddaljenim računalnikom, na katerega se moramo prijaviti. Vendar pa je namenjen le prenosu datotek med tako povezanima računalnikoma, in ne običajnemu delu. Če se s *telnetom* povežemo z oddaljenim računalnikom, še ne pomeni, da znamo tudi prenašati datoteke; FTP nam to delo zelo olajša.

Ftp lahko torej uporabljamo za prenašanje svojih datotek, če imamo uporabniško ime na več računalnikih; prav tako si z njim lahko izmenjujeta datoteke dva uporabnika na različnih računalnikih, vendar mora zato vsaj eden izdati drugemu svoje geslo (prenos seveda lahko poteka v obeh smereh); lahko pa *ftp* uporabljamo tudi za prenos datotek z javno dostopnih virov na svoj računalnik.[14]

Omenimo še, da si lahko znanci datoteke (vsaj v čisti ASCII obliki) izmenjujejo pravzaprav tudi po elektronski pošti. Prednost tega je, da ni treba nikomur izdajati gesla. Vendar velja opozoriti, da nekateri "poštarski" programi včasih dodajajo besedilu kakšne znake ali pa ne prenašajo predolgih sporočil, ki jih moramo zato potem razbiti na več delov. Pošiljanje binarnih datotek je še bolj zapleteno, saj jih je treba najprej kodirati v ASCII obliko, nato pa spet nazaj. *Ftp* brez težav prenaša kakršnekoli datoteke v njihovi izvirni obliki.

Datoteka, ki jo prenašamo, lahko skriva zelo različno vsebino: recimo program v binarni obliki, formatirano besedilo, pa tudi sliko, zvok ali film. Lahko pa je besedilo v čisti ASCII

[13] Pustimo ta hip ob strani dejstvo, da lahko ukazi **copy** pogosto prenašajo tudi datoteke med *istovrstnimi* računalniki.

[14] Pri tem ne pozabimo: če na računalniku v omrežju, s katerega poganjamo *ftp*, le gostujemo (npr. prek modemske povezave na *Stenarju*), moramo datoteko nato prenesti še z gostiteljskega računalnika na svojega (s svojim komunikacijskim programom), preden je resnično "doma" in jo lahko npr. izpišemo na tiskalnik.

obliki, brez dodatne kode za oblikovanje besedila. To moramo upoštevati, ko nastavljamo način prenosa datotek: **ascii** (za "čista" besedila) ali **binary** (za vse drugo).

Potek dela

Ftp skoraj na vseh vrstah računalnikov poženemo na enak način:

$ **ftp** *ime_oddaljenega_računalnika*

Lahko pa ime računalnika izpustimo in po pričakovanju se oglasi FTPjev tolmač ukazov s pozivom `FTP>`. Zvezo zdaj vzpostavimo z ukazom

`FTP>` **open** *ime_oddaljenega_računalnika*

Oddaljeni sistem od nas najprej zahteva uporabniško ime in geslo, nato pa se odzove s pozivom

`FTP>`

ali (npr. če poganjamo *ftp* s *Stenarja*),

`IME_ODDALJENEGA_RAČUNALNIKA>`

ali kaj podobnega. Že ta poziv nas opozarja, da tu ne gre za običajno delo na oddaljenem računalniku, kot je to pri *telnetu*, ampak da vse ukaze interpretira *ftp*. Na razpolago imamo omejen nabor ukazov, potrebnih za prenos datotek v obe smeri:

?
: Pokaže seznam ukazov.

help
: Pomoč. "**help** *ukaz*" opiše *ukaz*.

user *ime*
: Prijavi se na oddaljeni računalnik pod uporabniškim imenom *ime*. To sicer lahko zahtevamo že na začetku, če napišemo **ftp** *ime_oddaljenega_računalnika*/**user**=*ime*.

dir
: Seznam datotek v trenutnem imeniku na oddaljenem računalniku.

cd *imenik*
: (Change Directory). Spremeni delovni (trenutni) imenik na oddaljenem računalniku v *imenik*. Sintaksa ukaza je podobna npr. DOSovi (v podimenik z imenom *bin* se premaknemo z ukazom **cd** *bin*).

cdup
Vrnitev v imenik, ki je za eno raven nad trenutnim (včasih deluje tudi ukaz **cd ..**, kot ga poznamo iz DOSa).

pwd
(Print Working Directory). Pokaže trenutni (delovni) imenik datotek na oddaljenem računalniku.

binary
Nastavi ustrezni način za prenos binarnih datotek.

ascii
Nastavi ustrezni način za prenos datotek v obliki ASCII (ta način je navadno privzet ob začetku dela s *ftp*).

get *ime_datoteke*
Prenese datoteko *ime_datoteke* z oddaljenega računalnika na lokalnega. Datoteka se shrani pod istim imenom v trenutni imenik računalnika, na katerem poganjamo *ftp*. Če jo želimo shraniti pod drugim imenom, uporabimo ukaz **get** *ime_datoteke novo_ime_datoteke*.

put *ime_datoteke*
Prenese datoteko *ime_datoteke* z lokalnega na oddaljeni računalnik. Datoteka se shrani pod istim imenom v trenutni imenik oddaljenega računalnika. Če jo želimo shraniti pod drugim imenom, uporabimo ukaz **put** *ime_datoteke novo_ime_datoteke*.

mget *datoteke*
Prenese skupino datotek, katerih imena ustrezajo vzorcu *datoteke* (npr. $*$.*txt* pomeni vse datoteke s podaljškom *txt*) z oddaljenega računalnika na lokalnega.

mput *datoteke*
Prenese skupino datotek, katerih imena ustrezajo vzorcu *datoteke* z lokalnega na oddaljeni računalnik.

Ctrl-A
Pokaže stanje med prenašanjem datoteke.

Ctrl-G
Prekine prenos datoteke in se vrne v ukazni način *ftp*.

close
Prekine povezavo z oddaljenim računalnikom.

quit
Konča delo s *ftp* in izstopi iz programa.

Ukazov je seveda še veliko. Mnogi med njimi so sinonimi, kar olajšuje delo uporabnikom, ki so vajeni različnih sistemov. Pogosto srečamo ukaze za delo z datotekami na lokalnem ("našem") računalniku – navadno imajo predpono **l**, npr. **lcd** za **l**ocal **c**hange **d**irectory, pa **ldir**, **lpwd**, ... Nabor ukazov je seveda odvisen tudi od posameznega programa. Zgoraj našteti ukazi, ki zadoščajo za osnovno delo, so enaki pri skoraj vseh izvedbah, zagotovo pa veljajo za *ftp* na računalniku *Stenar*.

Javno dostopne datoteke

Veliko informacij v omrežju Internet je javno dostopnih. Mnoge organizacije, povezane v Internet, ponujajo na svojih računalnikih posebno storitev, imenovano *FTP strežnik z anonimnim dostopom* (anonymous FTP server). Gre za posebno področje na računalniku, kjer so shranjene datoteke, namenjene javni uporabi. Ta področja lahko dosežemo z uporabo programa *ftp* in si z njih na svoj računalnik prenesemo datoteke, ki nas zanimajo.

Za dostop do teh storitev uporabljamo javno uporabniško ime *anonymous* (včasih tudi *ftp*). Ustrezen ukaz na *Stenarju* je torej

$ ftp *ime_oddaljenega_računalnika* **/user=anonymous**

Namesto gesla vpišemo svoj elektronski naslov (ponavadi tudi sam strežnik na oddaljenem računalniku ob uporabi imena *anonymous* pove, kakšno geslo zahteva[15]).

Te storitve ne ponujajo vsi računalniki; trenutno jih je približno 1500, z več kot 3 000 000 razpoložljivimi datotekami. Med njimi so članki, elektronske knjige, programska oprema v javni lasti, slike in še kaj. Obstajajo seznami vseh računalnikov, do katerih lahko s *ftp* pridemo kot "anonimneži". Vendar bomo kmalu spoznali tudi učinkovitejše načine, da poiščemo tako dostopne datoteke, ki so za nas zanimive.

Kadar uporabljamo *ftp* za dostop do javnih datotek, veljajo nekatere dodatne omejitve. Normalno je npr., da nam ukaz za spreminjanje imenikov ne dovoli "zaiti" z javnega področja. Datoteke ponavadi lahko le "jemljemo" (**get**), ne moremo pa jih s svojega računalnika prenesti na javno dostopno področje. Včasih taka storitev ni na voljo ves dan, ampak samo takrat, ko oddaljeni računalnik ni obremenjen z drugimi rednimi opravili.

Ko se prijavimo na računalnik z javno dostopnimi datotekami, je koristno najprej prenesti in prebrati navodila. Ponavadi jih najdemo v glavnem imeniku na datoteki, katere ime je lahko *README*, *read.me*, *INDEX*, *INFO* ali kaj podobnega. Ta vsebuje nekaj podatkov o razpoložljivih datotekah in napotke za prenos. Datoteko *README* prenesemo na svoj računalnik z ukazom **get** *README*, nato jo lahko preberemo. Ne pozabimo na pravilo, da imena datotek pišemo tako, kot jih vidimo napisane v imeniku, saj mnogi sistemi ločijo velike in male črke.

[15]Zavedati se moramo, da dostop do takega računalnika v resnici ni anonimen; strežnik se zaveda, kdo je vzpostavil zvezo, običajno se ta informacija tudi zabeleži.

Da bi mogli pravilno nastaviti način prenosa (**binary** ali **ascii**), moramo vedeti nekaj o datotekah, ki jih prenašamo. Datoteke so pogosto tudi *zgoščene* (*compressed*, torej obdelane s posebnimi programi, ki podatke zgostijo, da datoteka zavzame manj prostora. Še posebej programska oprema je običajno na voljo v zgoščeni obliki. Take datoteke so v binarni obliki, torej moramo pred prenosom uporabiti ukaz **binary**. Pred uporabo na svojem računalniku pa jih moramo še predelati v normalno obliko (*dekomprimirati*), ponavadi z uporabo enakih programov, kot so bili uporabljeni za zgoščanje. Ti programi so večinoma v javni lasti.

Pri določanju tipa datoteke oziroma tipa zgoščanja, ki je bilo na njej uporabljeno, si pomagamo s standardnimi podaljški v imenih datotek. V naslednji tabeli je predstavljenih nekaj najpogostejših podaljškov, ki jih srečamo v imenih datotek. V tabeli so tudi oznake računalnikov, katerim so datoteke s takimi podaljški namenjene , tip datoteke oziroma ime ustreznega programa za zgoščanje ter način prenosa datoteke (binarni ali ASCII).

Pogosti tipi datotek na Internetu				
podaljšek	vsebina	sistem	naziv/program	način prenosa
.txt	besedilo	splošno	besedilo	ASCII
.gif	grafika	splošno	GIF	binary
.jpg/.jpeg	grafika	splošno	JPEG	binary
.tif	grafika	splošno	TIFF	binary
.mpg/.mpeg	video	splošno	MPEG	binary
.ps	dokument	splošno	PostScript	ASCII
.uue	kodirano	splošno	uuencode in uudecode	ASCII
.arc	zgoščeno	PC	ARC, PKPAK	binary
.arj	zgoščeno	PC	ARJ	binary
.lzh	zgoščeno	PC	LHArc	binary
.pak	zgoščeno	PC	PAK	binary
.voc	zvok	PC	Soundblaster	binary
.wav	zvok	PC	WAVE	binary
.qt	video	PC/Mac	Quicktime	binary
.zip	zgoščeno	PC	PKZIP, InfoZIP	binary
.zoo	zgoščeno	PC	zoo	binary
.au	zvok	Unix	Sun Ulaw	binary
.snd	zvok	Unix	NeXT avdio datoteka	binary
.Z	zgoščeno	Unix	compress	binary
.gz	zgoščeno	Unix	gzip, gmzip	binary
.tar	arhiv (več datotek)	Unix	tar	binary
.bin	kodirano	Mac	MacBinary	binary
.cpt	zgoščeno	Mac	Compact Pro	binary
.hqx	kodirano	Mac	BinHex 4.0	ASCII
.sit	zgoščeno	Mac	StuffIt	binary

4.2.3 Iskanje javno dostopnih datotek s programom *Archie*

Do podatka, ali nam je ta ali ona za nas zanimiva datoteka dostopna prek FTP, oziroma kje je shranjena, lahko pridemo na več načinov:

- Zvemo za to od znanca.
- Podatek kje preberemo (knjiga, revija, USENET News).
- Pomagamo si s storitvami za iskanje informacij na Internetu.

Javno dostopnih datotek je v omrežju preveč, da bi imeli stalno pregled nad vsemi. Kadar pa želimo izvedeti, ali je neka datoteka (npr. program v javni lasti) dostopna prek anonimnega FTP, lahko uporabimo program *Archie*. Ta program preiskuje skladišče podatkov, kjer so shranjena vsa imena datotek v vseh znanih javno dostopnih arhivih. Tako nam ni treba "peš" preiskovati različnih računalnikov, da bi našli kako datoteko. Poleg tega za iskanje zadostuje delno ime datoteke, kar nam delo dodatno olajša. Ustrezni strežniki za Archie delujejo na različnih računalnikih po svetu, skladišče podatkov, ki ga pregledujejo, pa se redno obnavlja. Do teh strežnikov lahko pridemo z uporabo *telneta*, javno uporabniško ime pa je običajno *archie*. Priporočljiva je uporaba odjemalca, če nam je le na voljo, saj to precej pohitri delo in zmanjša obremenitev omrežja.

Na računalniku *Stenar* tak program obstaja. Poženemo ga z ukazom **archie**, ki mu sledijo parametri in določila. V njih navedemo ime iskane datoteke in podrobneje določimo način iskanja. Ukaz **archie** brez parametrov izpiše navodila za uporabo. Določila podamo z ustreznimi črkami, pred katerimi stoji znak '-'. Ta sintaksa je podobna kot na sistemu UNIX in pri nekaterih programih za PC (npr. PKZIP). Oglejmo si nekaj najbolj uporabnih določil:

-s	Išče ime datoteke, ki vsebuje podani niz; pri tem ne ločuje med malimi in velikimi črkami.
-c	Pri iskanju podnizov loči med malimi in velikimi črkami.
-e	Ime datoteke se mora natanko ujemati s podanim nizom (to določilo je privzeto, če ne navedemo drugega).
-l	Izpiše en zadetek na vrstico (primerno za preiskovanje rezultatov s programom).
-m *število*	Določi največje število zadetkov, ki jih izpiše (privzeta vrednost je 98).
-o *ime_datoteke*	Določi ime datoteke, na katero naj shrani rezultate iskanja; če ime ne vsebuje imenika, se datoteka shrani na trenutni imenik.

Za primer sestavimo ukaz, ki bo v arhivih, javno dostopnih prek FTP, poiskal največ 30 datotek, katerih imena vsebujejo niz "kermit", rezultate pa naj zapiše na datoteko *kermit.kje*. Uporabili bomo določila **s** (ne želimo razlikovati med malimi in velikimi črkami),

o (za izhodno datoteko) in **m** (za določanje števila zadetkov):

$ **archie -sm** *30* **-o** *kermit.kje kermit*

Iskanje bo hitrejše, če ime datoteke natančno poznamo. V tem primeru določilo **s** izpustimo oziroma uporabimo določilo **e**.

Tu je del rezultatov zgornjega iskanja:

```
Host ftp.vmars.tuwien.ac.at

    Location: /pub/minix/updates/1.5-1.6.25/pc
           FILE -rw-r--r--       3331  Jun  9 1993  kermit.sh.Z

Host nic.switch.ch

    Location: /mirror/msdos
      DIRECTORY drwxrwxr-x       1024  Jun 18 07:27  kermit
    Location: /software/sources/network
           FILE -rw-rw-r--       2539  Aug 22 1991  kermit

Host slopoke.mlb.semi.harris.com

    Location: /pub/mac/util
           FILE -rw-r--r--     140039  Jan  7 1991  mackermit-098.hqx

Host faui43.informatik.uni-erlangen.de

    Location: /mounts/epix/public/pub/amiga/aminet/comm/term
           FILE -rw-r--r--      71422  Apr  2 01:26  xprkermit.2.lha
           FILE -rw-r--r--        499  Apr  2 01:26  xprkermit.2.readme
    Location: /mounts/epix/public/pub/pc/msdos/communication
      DIRECTORY drwxr-xr-x       1024  Jun  5 03:04  kermit
```

Iz rezultatov vidimo, da smo našli programe za različne računalnike. Med njimi poiščemo takega, ki nam ustreza, pri tem pa pazimo, da se nahaja na čim bližjem računalniku, da ne bomo po nepotrebnem obremenjevali omrežja. Nato z uporabo *ftp* prenesemo datoteko na svoj računalnik.

4.3 Sredstva množične komunikacije med uporabniki računalniških omrežij

Eden glavnih namenov omrežja je seveda komuniciranje med njegovimi uporabniki. Načinov in vrst komuniciranja je več. Spoznali smo že elektronsko pošto, ki je vsekakor najosnovnejša in v nekem smislu tudi najbolj univerzalna oblika.

V osnovi je elektronska pošta namenjena izmenjavi sporočil med dvema uporabnikoma. Računalniško omrežje pa ponuja tudi možnosti množične komunikacije, torej izmenjave sporočil med večjim številom ljudi.

Prva oblika množičnega komuniciranja v omrežju, ki nam pride na misel, je "skupinska diskusija po pošti" ali *mailing list*. Gre preprosto za posebej sestavljen in upravljan seznam elektronskih naslovov članov neke skupine, ki omogoča, da vsi člani dobijo sporočilo, ki ga pošlje katerikoli od njih.

Seveda lahko v majhnih skupinah problem rešimo preprosto tako, da vsako sporočilo pošljemo večim naslovnikom hkrati.[16] Za večje in razpršene skupine naslovnikov pa postane vzdrževanje naslovov naporno (vsak potrebuje svojo datoteko) in neučinkovito, zato je prava rešitev uporaba centralnega seznama naslovov (*mailing list*), ki se vzdržuje s pomočjo posebne programske opreme. Tak seznam lahko naredi *postmaster* sistema, na katerem uporabljamo elektronsko pošto.

Seznami elektronskih naslovov se ne uporabljajo le za žive diskusije, temveč tudi za objavljanje in razširjanje informacij, ki so zanimive za vse člane skupine, torej nekakšne elektronske časopise. Take časopise lahko naročimo, navadno tako, da pošljemo na naslov vzdrževalca seznama ustrezno sporočilo. Vzdrževanje seznama je lahko tudi avtomatsko.

Ker na Internetu obstaja vrsta drugih storitev tako za komuniciranje kot tudi za dostop do informacij, je takšen dostop do informacij zanimiv zlasti za uporabnike drugih omrežij, ki jih z Internetom povezuje le elektronska pošta. Tak primer je BITNET in njegov znameniti LISTSERV, ki avtomatsko vzdržuje veliko število tovrstnih seznamov. Mnoge diskusije so bogate z informacijami, nekatere pa so le kopije drugih virov na Internetu (npr. diskusij na globalni elektronski oglasni deski USENET News).

Očitna prednost tega, da sporočila dobivamo v svoj poštni predal, namesto da bi jih iskali po omrežju ali celo zamudili del diskusije (kot se to lahko zgodi na USENETu, kjer je rok trajanja sporočil omejen), je hkrati tudi slabost, saj je poštni predal lahko hitro prenapolnjen. Če imamo možnost, izberemo rešitev, ki nam bolj ustreza.

Od sredstev množične komunikacije na omrežju smo doslej spoznali že konferenčni sistem NOTES na omrežjih DECnet, nekateri pa poznate tudi BBS, elektronske oglasne deske na računalnikih, ki so med seboj povezani preko modemov.

---

[16] Naslove preprosto ločimo z vejicami, lahko pa jih tudi napišemo na datoteko. Konkretna navodila za to najdemo npr. v programu **mail** z ukazom **help send**.

Vsi omenjeni načini komuniciranja so v grobem nekakšne dopisne diskusije. Poleg tega načina se lahko prek omrežja pogovarjamo tudi "v živo", z enim ali več sogovorniki. Za lokalne pogovore te vrste najdemo na VMS sistemih program *PHONE*, na Internetu pa je razširjen Internet Relay Chat (IRC). Pojavljajo se tudi storitve oz. programi, ki omogočajo multimedijsko izmenjavo sporočil (glas, slika), vendar so zaradi premajhne zmogljivosti omrežij še precej v povojih.

V nadaljevanju bomo podrobneje predstavili le najpopularnejši medij za množično izmenjavo sporočil: USENET News, oglasno desko globalnega računalniškega omrežja. Diskusije na tem mediju segajo še preko meja Interneta.

4.3.1 Pravila vedenja v medsebojni komunikaciji na Internetu

Pri interakciji tako velikega števila ljudi, ki lahko izhajajo iz zelo različnih okolij, lahko pride do neželjenih konfliktov, kadar ne vemo, kaj se na omrežju smatra za sprejemljivo in kaj ne.

Tu skušamo podati hitri opis etike omrežja (pogovorno: *netiquette*), oziroma skuša razložiti, kakšno obnašanje je na mreži dopustno in kakšno ne.

Uporabniki Interneta si lahko izmenjujejo informacije na več načinov, najbolj priljubljena pa sta elektronska pošta in USENET konferenčni sistem. Pri prvem ponavadi pišemo eni osebi, medtem ko pri drugem tisto, kar napišemo, včasih prebere nekaj (deset-, sto-) tisoč ljudi (včasih manj, včasih več, odvisno od popularnosti konference).

Pri elektronskem načinu komunikacije se lahko zgodi, da nas zanese občutek neosebnosti, težko si je tudi predstavljati, koliko bralcev lahko prebere naš prispevek. Pri tem načinu komunikacije tudi manjka govorica telesa in ton glasu, zato moramo tudi paziti, da nas bodo razumeli na pravi način (sarkazem lahko kdo od bralcev vzame čisto zares, če tega dodatno ne označimo).

Zaradi vsega navedenega je koristno, da se tudi tu držimo pravil dialoga, ki smo jih navajeni v medosebni komunikaciji, se posebej pa se moramo navaditi na naslednje:

- ko začnemo brati konferenco, si vzemimo najprej čas samo za branje – s tem bomo dobili občutek o tem, kakšni ljudje na njej debatirajo, o čem govorijo in če morda obstajajo kakšna "interna pravila" v konferenci,
- preden odgovorimo (ali napišemo prispevek na novo temo), dobro razmislimo, kaj sploh želimo povedati,
- velike večine bralcev ne bo zanimal prispevek, v katerem bo pisalo samo: "Strinjam se." – konference so namenjene izmenjavi informacij in polemiziranju (če podamo neke nove argumente in razloge za (ne)strinjanje z nečim, je to povsem drugačen primer),

- če pišemo odgovor, vedno citirajmo del originalnega prispevka, na katerega odgovarjamo, da bo povezava jasna (čeprav po drugi strani nima smisla citirati celotnega originala...),
- kadar postavljamo vprašanje, dobro opišimo svojo situacijo; pri vprašanjih, ki se nanašajo denimo na določen program, napišimo na katerem tipu računalnika delamo (in operacijskem sistemu), katero verzijo programa uporabljamo in kaj želimo storiti,
- pri takih temah ne zastavljajte vprašanj, za katera so podane rešitve v priročniku programa,
- ne pričakujte odgovora vedno v roku enega dneva, zakasni lahko zaradi tehničnih problemov (obremenitev omrežja), ali pa preprosto zaradi tega, ker je ta, ki ve odgovor na naše vprašanje, trenutno "zasut" z elektronsko pošto ali delom, in še ni našel časa, da bi napisal odgovor.

Na večini konferenc se občasno (recimo vsak mesec) objavljajo odgovori na pogosto zastavljana vprašanja ("Frequently Asked Questions" ali s kratico: FAQ). Tam se zbirajo odgovori na vprašanja, ki so se velikokrat do sedaj pojavila v konferenci. Možno je, da je odgovor na vprašanje, ki ga želite zastaviti, že tam. Tule si lahko preberete nekaj o konferencah na splošno, na voljo pa vam je seznam FAQ dokumentov v hipertekst obliki, ali pa preko servisa FTP.

Pri elektronski pošti pa velja omeniti to, da nimate nadzora nad tem, kaj bo s pismom naredil naslovnik. Prav tako ne morete vedeti, kako varen je računalnik, kjer bo nekdo prejel vašo pošto – tam bo recimo lahko vašo pošto prebral še kdo drug kot tisti, ki ste mu pisali, kdo ve. V resnici se seveda elektronske pošte ne prebira kar poprek. Vendar se je dobro potencialne nevarnosti zavedati, kadar pošiljamo sporočila, za katera ne želimo, da bi jih prebral kar kdorkoli.

4.3.2 USENET News

USENET News (ali Network News, ali Netnews) je sistem množične komunikacije za izmenjavo sporočil v standardnem formatu preko računalniškega omrežja. Sporočila so glede na tematiko razvrščena v diskusijske *skupine* (*newsgroups*), včasih imenovane tudi *elektronske konference*. Prenašajo se od računalnika do računalnika preko različnih omrežij, ne le v Internetu. USENET News je torej največja (elektronska) oglasna deska na svetu, ki jo prebirajo milijoni ljudi. Na njej najdemo zadnje novice iz stroke, vietnamske kuharske recepte, potovalne nasvete ter komentarje aktualnih dogodkov po svetu "iz prve roke". Z ljudmi po vsem svetu se lahko pogovarjamo o svojem delu ali o svojem najnenavadnejšem hobiju, debatiramo o politiki in navezujemo stike.

Po namembnosti so USENET News sorodne sistemu *NOTES*, ki smo ga omenili v prvem delu priročnika, po organiziranosti sporočil pa še bolj spominja na BBS (Bulletin Board

System), ki ga poznamo iz sveta osebnih računalnikov. USENET News je seveda globalna različica tega sistema neprimerno večjih razsežnosti.

Obstaja nekaj tisoč skupin novic. Nekatere so bolj "uradne" kot druge, nekatere so zgolj lokalnega značaja, nekatere imajo moderatorje,[17] druge pa ne. Imena skupin odražajo določeno hierarhijo obravnavanih tem, ki so razdeljene na splošne skupine in podskupine. Oblika imena je uporabnikom Interneta že domača: tako na primer

rec.games.go

predstavlja ime skupine, ki spada v širšo domeno "*rec*reation", kjer se pogovarjamo pretežno o stvareh, s katerimi se ukvarjamo v prostem času. Konkretneje gre za diskusijo o eni izmed iger (*games*), in sicer o igri *go*.

Poznamo sedem "uradnih" glavnih hierarhičnih skupin:

comp	-	računalniške teme
sci	-	znanost
soc	-	družba
news	-	prispevki v zvezi s samimi USENET News
rec	-	prosti čas (hobiji, umetnost ...)
misc	-	razno
talk	-	debatne skupine

Obstajajo pa še druge glavne skupine, ki jih ne prenašajo nujno vsi računalniki v sistemu USENET. Veliki glavni skupini sta na primer *alt* (različne alternativne skupine) in *bionet* (biologija). Nekatere so geografsko lokalne, lahko so povezane z določenimi organizacijami (npr. *ieee*) ali pa so komercialne (*clarinet*).

Za razliko od NOTES se tu novice/prispevki ne shranjujejo le na enem računalniku v omrežju, ampak se skupine novic v velikih datotekah prenašajo od računalnika do računalnika, kjer jih ustrezni strežniki (*news server*) nudijo uporabnikom. Za dostop do sistema USENET News torej potrebujemo dovoljenje nekega strežnika in ustrezno programsko opremo – *bralnik novic* (*news reader*). Upravitelj računalnika določa, katere skupine novic posreduje določen strežnik. Pri tem se navadno ozira na potrebe uporabnikov in razpoložljiv prostor na disku za hranjenje novic.

Prispelih novic strežnik ne shranjuje trajno, saj jih dnevno prihaja ogromna količina, ampak le za določen čas. Tudi ta čas določa upravitelj sistema po že omenjenih kriterijih. Določeni pomembnejši prispevki se zato objavljajo periodično. Tak primer so npr. odgovori na pogosto zastavljena vprašanja v neki skupini. V večini aktivnejših skupin se oblikuje seznam takih odgovorov, pogosto imenovan FAQ (Frequently Asked Questions), ki se v skupini redno pojavlja, shranjen pa je tudi v arhivih podobnih dokumentov (FAQ archive) in objavljen v skupini *news.answers*.

[17]Osebe, ki izbirajo in urejajo prispevke v tako skupino.

Spremljanje USENET News na *Stenarju*

Bralnik novic na računalniku *Stenar* se imenuje *ANU-NEWS*. Ta odjemalski program spremlja čez 6000 skupin novic na ARNESovem strežniku, zato si ob zagonu vzame malce časa. Ker ne moremo (niti ne želimo) spremljati vseh skupin novic, se lahko na določene skupine "naročimo" (ukaz **register** v *ANU-NEWS*). Za te skupine program potem vodi evidenco o že prebranih novicah in nam s tem olajša pregledno branje. Poleg tega pa so nam ves čas na voljo tudi vse ostale, neregistrirane skupine novic.

Seznam vseh skupin s kratkim komentarjem vsebine oz. tematike je redno objavljen v *news.announce.newgroups*, zato ga je vredno pogledati (lahko ga npr. shranimo v datoteko in v miru pregledujemo z urejevalnikom besedil), da se med mnogimi temami lažje znajdedemo.

Delo z *ANU-NEWS*

ANU-NEWS poženemo z ukazom **exe anu-news** ali na kratko

$ anu

Nalaganje novic običajno zahteva kar nekaj časa, včasih nas program še vpraša, ali želimo registrirati nove skupine novic. Nato na zaslonu dobimo približno takole sliko:

```
   NEWSGROUPS  [NEW, 24 Newsgroups]

           Newsgroup                                           Count    Unread
   1       alt.usage.english                                     422       422
-> 2       bionet.announce                                        10        10
   3       comp.answers                                          314       314
   4       comp.internet.net-happenings                           24        24
   5       comp.security.announce                                  2         2
   6       gnu.announce                                            2         2
   7       misc.answers                                           69        69
   8       misc.kids.consumers                                    37        37
   9       misc.news.bosnia                                       54        54
   10      news.announce.conferences                             576       576
   11      news.announce.newusers                                 18        18
   12      news.answers                                         1190      1190
   13      news.newusers.questions                               486       486
   14      rec.answers                                           427       427
   15      rec.food.preserving                                    23        23
   16      rec.music.celtic                                      174       174
   17      sci.answers                                            85        85
NEWS>
```

V prvi vrstici lahko vidimo izbirni kriterij za prikaz skupin novic ("NEW" velja za *registrirane* skupine, ki *vsebujejo neprebrane novice*). Sledijo imena skupin z zaporednimi številkami in podatka o številu prispevkov v skupini ("Count") ter številu neprebranih prispevkov ("Unread"). Kazalec (`->`) kaže na trenutno izbrano skupino. Po pričakovanju lahko ta kazalec premikamo s pomočjo tipk-puščic, s pritiskom na Return pa potrdimo izbiro. Ko izberemo neko skupino, se prikaže seznam prispevkov v njej:

```
news.announce.newusers: 18 Items (#520 - #554) Reg:1 Prot:RW
        -<Explanatory postings for new users. (Moderated)>-
         Title                                From                Lines  Date
->520    Introduction to the *.answers newsgrou news-answers-reques   313 19-Sep
  521    Introduction to the *.answers newsgrou news-answers-reques   313 19-Sep
  539    Anonymous FTP: Frequently Asked Questi Perry.Rovers@kub.nl  1023  1-Oct
  540    How to Create a New Usenet Newsgroup   tale@uunet.uu.net (   158  3-Oct
  541    Emily Postnews Answers Your Questions  netannounce@deshaw.   405 13-Oct
  542    A Primer on How to Work With the Usene netannounce@deshaw.   421 13-Oct
  543    Introduction to news.announce          netannounce@deshaw.   182 13-Oct
  544    How to Get Information about Networks  netannounce@deshaw.   231 13-Oct
  545    Answers to Frequently Asked Questions  netannounce@deshaw.   854 13-Oct
  546    Rules for posting to Usenet            netannounce@deshaw.   334 13-Oct
  547    A Guide to Social Newsgroups and Maili taylor@netcom.com (   283 13-Oct
  548    USENET Software: History and Sources   netannounce@deshaw.   534 13-Oct
  549    Hints on writing style for Usenet      netannounce@deshaw.   112 13-Oct
  550    What is Usenet?  A second opinion.     emv@msen.com (Edwar   471 13-Oct
  551    What is Usenet?                        netannounce@deshaw.   353 13-Oct
  552    Usenet Newsgroup Creation Companion    rdippold@happy.qual   492 14-Oct
  553    How to find the right place to post (F buglady@bronze.lcs.   347 15-Oct
  554    How to become a USENET site            clewis@ferret.ocuni   880 15-Oct
NEWS>
```

Iz prve vrstice razberemo ime skupine in število tekočih prispevkov. Številka neprebranega sporočila je poudarjena, če gre za seznam notic v registrirani skupini. Neprebrani naslovi so navadno poudarjeni. Poleg naslova posameznega prispevka (označenega s "*Title*" ali "*Subject*", kot pri elektronski pošti) lahko vidimo še elektronski naslov pošiljatelja, velikost prispevka in datum njegovega prispetja na oglasno desko.[18]

Kot že rečeno, je organizacija prispevkov drugačna kot v *NOTES*. Prispevki v skupinah (v *NOTES*: "konferencah") niso nadalje razdeljeni po temah (v *NOTES*: "topic"), temveč skupno temo razberemo iz naslova prispevka. Odgovori na določeno "temo" so označeni z "RE: tema", kot pri elektronski pošti.

Če pogledamo primer prispevka v USENET News, opazimo podobnost z elektronsko pošto:

[18] V izpisu so vsa večja polja nekoliko "porezana".

```
Group: news.newusers.questions, Item 24849   (Range: #22857 - #25352, Unread: 52
Subject: Why is Earth oval?
From: larrymor@crl.com (Laurence A. Moore), CRL Dialup Internet Access       (4
Date: 18 Oct 94 02:59:42 GMT

train2@iserver.shasta.forestry.state.ca.us (train2) wrote:

> I have never had a real answer to the question of why the earth
> is the exact shape it is.

Even today, Earth is not a solid object; originally, Earth was
mainly molten.

Under its very thin crust, Earth is still mainly molten.

If you spin a sphere in space, it will shrink from pole to pole
and spread out at the equator.

Earth is an oblate spheroid -- shrunk from pole to pole and
spread out at the equator.

NEWS>
<RETURN for more - 16/22 Lines (87%)>
```

Glava prispevka vsebuje poleg imena skupine in njegove zaporedne številke v njej še predmet (`Subject:`), naslov pošiljatelja in datum. V besedilu je z “>” na začetku vrstice označen citat, na katerega se nanaša odgovor.

Osnovni ukazi

Ukazi za delo z *ANU-NEWS* v mnogočem spominjajo na *NOTES*.

Ponovno lahko najprej spomnimo na ukaz **help**, ki nam pomaga podrobneje odkriti delovanje ostalih ukazov. Pričujoče besedilo je namenjeno bolj ali manj le uvajanju v uporabo različnih programov, za udobnejše delo pa je nujno, da si na ta način razširite poznavanje njihovih zmožnosti.

Za osnovno delo potrebujemo naslednje ukaze:

$\boxed{\uparrow}$ (ali tipka $\boxed{\mathbf{5}}$ na stranski tipkovnici)
Pomika izbirni kazalec `->` navzgor po seznamu.

$\boxed{\downarrow}$ (ali tipka $\boxed{\mathbf{2}}$ na stranski tipkovnici)
Pomika izbirni kazalec `->` navzdol po seznamu.

$\boxed{\textbf{Return}}$ ali $\boxed{\textbf{Enter}}$
Odpre izbrano skupino novic, ali prikaže izbrani prispevek v skupini, ali nadaljuje z branjem (odvisno od konteksta).

open *ime_skupine_novic*
Odpre skupino novic za branje (lahko je tudi neregistrirana, vendar je s tem ne registriramo).

close
Zapre tekočo skupino in znova prikaže seznam skupin. Če ukaz uporabimo med branjem, zaključi branje in nas vrne v seznam prispevkov tekoče skupine.

dir
Prikaže seznam skupin novic ali pa seznam prispevkov v izbrani skupini. **dir/all** pokaže vse dostopne skupine novic.

register *ime_skupine_novic*
Registrira ("naroči") skupino novic. Nasproten ukaz je **deregister**. Če na začetku nenamenoma registriramo cel kup skupin, je najpametneje uporabiti ukaz **deregister/all** in nato naročiti le izbrane.

read *številka*
Prikaže prispevek z ustrezno zaporedno številko.

read/new ali **next unseen**
Prikaže naslednji neprebran prispevek v tekoči skupini.

read/followup
Prikaže naslednji prispevek z isto temo.

extract *ime_datoteke*
Vsebino branega sporočila shrani v datoteko *ime_datoteke*. Namesto ukaza **extract** lahko uporabimo ukaz **save**.

reply
Pošlje odgovor na tekoči prispevek njegovemu avtorju (po elektronski pošti).

followup
Pošlje odgovor na trenutni prispevek v izbrano skupino novic, kjer ga lahko vsakdo prebere.

write ali **post**
Pošlje prispevek v izbrano skupino novic.

exit
Izstop iz *ANU-NEWS*.

Večina ukazov ima svoje ekvivalente na stranski tipkovnici. Preberite si "**help keypad**" in preizkusite, če vaš program za komunikacijo dobro oponaša VT100 tip terminala.

Pri nekaterih prispevkih želimo preprečiti, da bi jih po naključju prebral nekdo, ki bi mu bilo zaradi tega žal ali bi ga prispevek lahko užalil.[19] Navada je, da se take prispevke kodira z vgrajenim algoritmom *rot13* (avtomatsko z uporabo določila **/rot13** pri ukazu **post** oz. **followup**) ter ustrezno označi[20].

Na splošno se je treba pri pisanju "novic" držati določenega bontona. Zato najprej toplo priporočamo branje ustreznih dokumentov za začetnike in splošnih obvestil, ki jih najdemo v skupinah *news.newusers.questions*, *news.announce.**, *news.groups.**, *news.answers*, pa tudi drugod. Posebej skupini *news.newusers.questions* in *news.announce.newusers* predstavljata obvezno branje za vse novince v USENET News, saj poleg več besedil o etiki omrežja najdemo tudi odgovore na vsa splošna vprašanja o USENETu. Ko preberemo ključne prispevke v teh skupinah, pa je morda najbolje, da si shranimo seznam aktivnih diskusijskih skupin, ki je objavljen v *news.announce.newgroups*, ga v miru pregledamo in se odločimo, katere skupine želimo redno spremljati.

Preden se vključimo v diskusijo neke skupine, je dobro nekaj časa spremljati prispevke, da dobimo vpogled v tematiko in se izognemo ponavljanju. Preberimo dokument FAQ, če ga skupina ima. Kadar je odgovor namenjen le avtorju nekega prispevka, uporabljamo **reply**. Ne pozabimo, da se s svojimi prispevki javno izpostavljamo veliki publiki, ki je lahko zelo občutljiva. Nepotrebna pa bi morala biti opomba, da se izogibajmo žaljivim prispevkom.

[19] Take so lahko šovinistične šale ali pa prispevki, ki npr. razkrivajo razplet nekega filma.
[20] Npr. oznaka "spoiler" za "izdajanje skrivnosti".

4.4 Križarjenje po svetovnem omrežju

Prehajamo na novejša orodja, ki so namenjena predvsem raziskovanju obsežnega informacijskega prostora omrežja Internet. Ta orodja omogočajo udobno sprehajanje po omrežju in pregledovanju dokumentov ter uporabljanje storitev, ki so na voljo na različnih računalnikih. Vse "tehnične podrobnosti", ki se skrivajo v ozadju tega dogajanja – prenos datotek, imena računalnikov, prijavljanje na oddaljene računalnike, gesla, specifični ukazi tujih računalnikov in podatkovnih baz – vse to je uporabniku kar v največji meri prihranjeno. Pri uporabi teh orodij dobimo občutek, da brskamo po eni sami veliki knjižnici podatkov, ne da bi nam bilo zares treba vedeti, na katerem koncu sveta se neki podatki dejansko nahajajo. Delo se reducira na prelistavanje velikanskega, vendar nepopolnega in mestoma nekoliko zmedenega in nezanesljivega elektronskega kataloga z mnogo navzkrižnimi referencami.

Zdaj se ne srečujemo več z začetniškimi težavami, ki nas pestijo zato, ker ne poznamo množice različnih ukazov za delo s storitvami na različnih računalnikih, ampak s pravimi problemi omrežja.

Prvi je vsebinski: pri tako veliki količini podatkov je nujno prisoten šum – po domače rečeno, veliko podatkov je odvečnih oziroma jih ne moremo uporabiti. Poleg tega so ti podatki precej neurejeni, kar otežkoča iskanje.

Drugi problem je tehnični: omrežje – zlasti izven računalniško najrazvitejših področij – ne zmore zadovoljevati naraščajočih zahtev po pretoku podatkov. Razvoj orodij, ki omogočajo lahek dostop do velikih količin podatkov tudi povprečnemu uporabniku, je hitrejši od rasti zmogljivosti omrežja. Stvari otežuje dejstvo, da se mnogo podatkov prenaša po nepotrebnem. Zato skušajmo s temi orodji ravnati tako, da ne preobremenjujemo omrežja.

4.4.1 Gopher

Gopher je ime storitve, ki omogoča dostop do različnih virov na Internetu prek enotnega uporabniškega vmesnika, ki je preprost za uporabo. Delo opravlja Gopherjev strežnik, ki na svojem računalniku hrani potrebne podatke, poleg tega pa je povezan s sorodnimi strežniki v omrežju. Globalni informacijski prostor, ki ga ti strežniki pokrivajo, imenujemo *gopherspace*. Uporabnik lahko preiskuje vire v tem prostoru in dostopa do podatkov, ki so shranjeni na različnih krajih v omrežju.

Ko pregledujemo podatke z *gopherjem*, se nam prikazujejo kot zaporedje gnezdenih menujev. Ta struktura spominja na imenik datotek z mnogimi podimeniki in datotekami. Podimenike in datoteke lahko najdemo na računalniku lokalnega strežnika ali na računalnikih oddaljenih strežnikov. Z gledišča uporabnika se zdi, kot da vsi podatki v menujih prihajajo z enega mesta.

Podatek v menuju je lahko besedilo ali binarna datoteka, imenik podatkov (naslednji menu), slika ali zvok. Izbira v *gopherjevem* menuju lahko poleg tega sproži tudi iskanje po indeksirani bazi podatkov ali pa namesto nas odpre povezavo z oddaljenim računalnikom (*telnet*), ne da bi nam bilo treba vtipkati (ali sploh poznati) naslov tistega računalnika. Če menu prikazuje seznam datotek v FTP arhivu, sproži izbira datoteke prenos te datoteke v naš direktorij (na *Stenarju*). Zato je *gopher* pogosto primernejši od neposredne uporabe *ftp* za pregledovanje javno dostopnih arhivov in snemanje datotek.

Kako uporabljamo Gopher

Do Gopherjevih strežnikov dostopamo prek lokalnega odjemalca ali, če tega nimamo, z dostopom (*telnet*) do oddaljenega javnega Gopherjevega odjemalca.

Podrobnosti izvedbe posameznih odjemalcev se razlikujejo glede na zmogljivost okolja (v oknih npr. za izbiranje v menujih uporabljamo miško), osnovne funkcije in ukazi pa so enaki. Ogledali si bomo uporabo *gopherja* prek navadnega terminala (oz. terminalske emulacije) na računalniku *Stenar*.

Gopher na *Stenarju* poženemo z ukazom

$ gopher

Program vzpostavi zvezo z vnaprej določenim strežnikom – običajno najbližjim.[21] Prikaže se glavni menu:

```
                 Internet Gopher Information Client 1.2VMS pl0

                       Root gopher server: kanin.arnes.si

 -->  1.  Computer Documentation, Software, and Information/
      2.  Information about Gopher/
      3.  Internet File Server (ftp) Sites/
      4.  Libraries and Reference Information/
      5.  Newspapers, Newsletters, and Weather/
      6.  Other Gopher and Information Servers/
      7.  Phone Books (ph)/
      8.  Test tree/
      9.  Welcome to the ARNES Gopher.



Press ? for Help, q to Quit, u to go up a menu                     Page: 1/1
```

V menuju izbiramo tako, da vtipkamo njegovo zaporedno številko ali pa premaknemo kaza-

[21] Če se želimo takoj povezati na nek oddaljeni strežnik, ga podamo kot parameter v ukazu: **gopher** *ime_oddaljenega_strežnika*.

lec (`-->`) na izbiro s tipkami za premikanje (puščici ↓ in ↑) ter jo potrdimo s pritiskom na Return. Poskusimo izbrati 3. možnost. Prikaže se naslednji menu:

```
                Internet Gopher Information Client 1.2VMS pl0

                        Internet File Server (ftp) Sites

 -->  1.  ARNES FTP server/
      2.  About-Anonymous-FTP.
      3.  About-this-directory.
      4.  Boombox at Minnesota, Home of the Gopher and POPmail/
      5.  FTP sites that start with 'a'/
      6.  FTP.CSO: University of Illinois CCSO's Main FTP Server/
      7.  FTP.NCSA: University of Illinois NCSA's Main FTP Server/
      8.  Keyword Search of Entries in FTP Menus <?>
      9.  Popular FTP Sites via Gopher/
      10. Type in the ftp site name for direct access <?>
      11. UXC: University of Illinois/
      12. Wuarchive.wustl.edu 128.252.135.4   GNU, X.11R3, GIF, IEN, RFCs,/
      13. b/
      14. c/
      15. d/
      16. e/
      17. f/
      18. g/

Press ? for Help, q to Quit, u to go up a menu                       Page: 1/3
```

Uporabljati znamo že puščici ↓ in ↑ ter tipko Return, zadnja vrstica v primeru pa nam pove, kako se vrnemo v prejšnji menu (**u**), kako iz programa izstopimo (**q**) in kako si ogledamo še preostale ukaze (**?**). Kratka, a lahko razumljiva navodila, ki jih dobimo z uporabo ukaza **?**, preberemo s pomočjo tipk za pomikanje kazalca, branje navodil pa končamo s Ctrl–Z, kakor nam prijazno svetuje osvetljena vrstica v spodnjem delu ekrana. Nasploh program vodi naše delo s podobnimi kratkimi napotki.

Ker opazimo, da je brana stran šele prva od treh, v navodilih preberemo, da se po straneh lahko premikamo z znakoma **+** in **-**. Prav tako izvemo, da smemo za vrnitev v prejšnji menu uporabljati tudi ←, za potrditev izbire pa →, če si to laže zapomnimo.

Posamezne izbire imajo na koncu oznake. Če oznake ni (npr. izbiri pod 2. in 3.), gre za datoteko. Oznaka "/" pomeni podimenik (podmenu), "<Picture>" je seveda slika, "<?>" pa označuje indeksno iskanje po besedilu.

Potrditev izbire sproži akcijo, ki je odvisna od tipa izbire. Če izberemo podmenu, se le-ta pač prikaže, potem pa lahko v njem izbiramo naprej.

Kadar izberemo datoteko, ki vsebuje navadno (ASCII) besedilo, se vsebina datoteke prikaže na zaslonu (na enak način kot navodila za pomoč pri ukazu **?**). Po besedilu se zopet lahko pomikamo s pomočjo puščic, iščemo v njem nize znakov, na koncu pa izstopimo s Ctrl–Z. Ob izstopu se nam ponudi izbira, da z ukazom (**save**) zahtevamo, naj se datoteka shrani na računalnik, s katerega poganjamo Gopherjeve odjemalce (lahko pa jo tudi z elektronsko pošto pošljemo poljubnemu naslovniku).

Če je izbrana datoteka v binarni obliki, je na terminalu ne moremo videti, *gopher* pa nam ponudi, da jo shranimo na disk.

Tako smo dokument oz. datoteko, ki smo jo z *gopherjem* našli nekje na omrežju, "prinesli domov", v tem primeru na računalnik *Stenar*, ne da bi pri tem morali poznati FTP ali sploh vedeti, kje v omrežju je ta datoteka shranjena.

To tudi pomeni, da lahko *gopher* uporabljamo namesto programa *ftp*. Ko pri raziskovanju v menujih naletimo na ime FTP strežnika, ki se nam zdi zanimiv, se ves postopek prijave na strežnik, ki smo ga opisali v poglavju o FTP (zagon programa *ftp* z imenom računalnika, prijava na uporabniško ime "anonymous", vnos lastnega elektronskega naslova za geslo) zreducira na pritisk tipke za izbiro. Namesto nas vse opravi *gopher*. Tudi od tod naprej ne potrebujemo novih ukazov. S puščicami in z izbiranjem se sprehodimo po imenikih, ki se nam kažejo kot *gopherjevi* menuji. Ko pridemo do zanimive datoteke, jo samo izberemo, nato pa jo lahko shranimo na disk.

Ta način "zbiranja" datotek je zelo primeren, kadar nanje naletimo šele pri raziskovanju virov omrežja. Kadar pa natanko vemo, na katerem FTP strežniku je iskana datoteka shranjena, je seveda bolje uporabiti neposredno program *ftp*, kot pa iskati ta strežnik po Gopherjevih menujih.

Kadar sprožimo iskanje po kaki bazi podatkov (oznaka "<?>"), nas program vpraša za eno ali več ključnih besed, ki jih lahko ločimo z besedicami **and**, **or** in **not**, na primer:

terminal **and** *setting* **or** *tset*

To bo povzročilo iskanje dokumentov v skladišču, ki vsebujejo bodisi niza *terminal* in *setting* ali pa niz *tset* (lahko tudi vse tri). Rezultat iskanja je običajen Gopherjev menu, sestavljen iz seznama datotek, ki vsebujejo iskani niz.

Na enak način na nekaterih strežnikih deluje tudi vmesnik do programa *Archie*, tako da lahko kar iz Gopherja sprožimo iskanje datotek po FTP arhivih.

FTP arhivi seveda sestavljajo le del domene, ki jo pokriva t.i. *gopherspace*. *Veronica* je orodje, ki preiskuje celoten *gopherspace* podobno, kot *Archie* preiskuje FTP arhive. *Veronica* išče ključne besede v vseh naslovih, ki se pojavljajo v menujih (vmes so tudi imena datotek v arhivih!).

Veronico najdemo na več Gopherjevih strežnikih, navadno pod izbiro "Other Gopher and

Information Servers", ki je vedno nekje v glavnem menuju ali blizu njega.

Zakaj ne bi poskusili sami raziskovati naprej? Z *gopherjem* je veliko laže delati kot pa o njem brati priročnike. Vsa potrebna pomoč vam je na voljo spotoma. Pred tem pa še zadnji napotek: ko po napornem sprehajanju po menujih odkrijete kakšnega posebej zanimivega, kamor bi se radi večkrat vrnili, si njegov položaj lahko shranite kot *zaznamek* (*bookmark*). Pomoč (**?**) nam pove, da nov zaznamek shranimo z ukazom **a**. Ko se torej odločimo, da bi se v trenutni menu še kdaj radi vrnili po bližnjici, preprosto pritisnemo tipko $\boxed{\text{a}}$. Zdaj se lahko sprehajamo naprej, Gopher zapustimo in se vrnemo naslednji dan. Ukaz **v** ("view bookmark file") kadarkoli prikaže vse zaznamke, ki smo jih dotlej nabrali, v običajni obliki menuja. Izbira zaznamka nas postavi naravnost v označeni menu.

4.4.2 World Wide Web

Pod tem imenom se skriva trenutno najbolj univerzalno orodje za neomejeno križarjenje po omrežju Internet. World Wide Web, ali na kratko WWW, na nov način združuje vse različne oblike dostopnih informacij in vse doslej obravnavane storitve na Internetu. Zaradi preprostosti uporabe je najbolj privlačno orodje tudi za začetnika. Hkrati pa je treba takoj povedati, da tudi relativno najbolj obremenjuje omrežje.

World Wide Web je poskus, da bi različne vrste dostopnih informacij na Internetu, z vašimi lokalnimi podatki vred, organizirali kot enotno množico dokumentov[22], dostopno prek enotnega uporabniškega vmesnika in med seboj povezanih z navzkrižnimi referencami v obliki *hiperteksta* (*hypertext*).

Hipertekst je (večdimenzionalno) besedilo, v katerem določene besede ali skupine besed predstavljajo povezavo z drugim dokumenti v omrežju. Tako besedilo pregledujemo s posebnim *pregledovalnikom za hipertekst* (angl. *hypertext browser*). Če med branjem besedila naletimo na posebej označeno besedo, ki skriva povezavo z drugim dokumentom, lahko povezavi sledimo (s pritiskom na tipko). Pregledovalnik nam nato prikaže dokument, na katerega kaže ta povezava. To je lahko del slovarja ali enciklopedije, ki podrobneje pojasnjuje označeno besedo, lahko je članek, poglavje v knjigi, sporočilo na elektronski oglasni deski USENET News, lahko pa tudi slika, zvok ali celo film (če ga pregledovalnik lahko prikaže). Sedaj lahko nadaljujemo z branjem novega dokumenta in naprej sledimo povezavam, ki jih najdemo v njem. Tako kot v *gopherju* si lahko pri tem shranjujemo knjižne zaznamke (*bookmarks*), da lahko naslednjič hitreje najdemo kak posebej zanimiv dokument. Nepotrebno je poudarjati, da lahko najdene dokumente (tudi npr. slike) s preprostim ukazom shranimo na računalnik, kjer poganjamo pregledovalnik hiperteksta.

Za osnovno uporabo pregledovalnika za WWW (enako bi lahko trdili za Gopher) pravzaprav ne potrebujemo priročnika. Vedeti moramo le, kako se ga na kakem računalniku požene, od tod naprej se brez večjih težav znajdemo sami. Osnovno pomoč dobimo z ekrana, potem pa ni težko poiskati dokumentov, ki nam o uporabi WWW povedo vse in še več.

Poskusite! Na *Stenarju* poženemo WWW pregledovalnik[23] z ukazom

$ www

[22] V WWW kontekstu pomeni "dokument" lahko besedilo ali binarno datoteko s programom, sliko, zvokom, filmom ...

[23] Gre za pregledovalnik z imenom *Lynx*, ki je prilagojen delu s terminali tipa VT, kakršnega emulira naš komunikacijski program.

Na zaslonu se prikaže nekaj takega:

```
                              Informacije za uporabnike Stenarja

          ARNES - AKADEMSKA IN RAZISKOVALNA MREZA SLOVENIJE

 Informacije za uporabnike Stenarja

      * Kako se tale rec sploh uporablja? (pritisni Return) (zelo
        zacetniski napotki)
      * Nekaj popularnih prirocnikov o Internetu in WWW
      * Delo uporabnikov racunalnika STENAR
      * Slovenija na WWW - kratka, a zanimiva predstavitev
      * Informacijski strezniki v Sloveniji
      * Nekatere priporocljive izhodiscne tocke za raziskovanje Interneta
      * Pravno varstvo racunalniskih programov v Sloveniji
      * Pogosti ukazi na racunalniku Stenar s kratko obrazlozitvijo
      * Kako postaviti svojo domaco stran na WWW

     ARNES home page (ARNESova domaca oz. vrhnja stran)




Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<-' to go back
  Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
 H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

Vidimo, da so nekatere besede ali fraze poudarjene (obarvane, podčrtane ali kako drugače). Kazalec čaka na prvi od teh besed. V spodnjih dveh vrsticah na zaslonu najdemo nekaj najpogostejših ukazov.

Poudarjene besede predstavljajo povezave z drugimi dokumenti. S puščicama $\boxed{\downarrow}$ in $\boxed{\uparrow}$ premikamo kazalec po teh povezavah v tekočem dokumentu. S pritiskom na $\boxed{\rightarrow}$ (tudi

Return) sledimo povezavi v nov dokument. Če bi npr. radi zvedeli kaj več o Sloveniji, se torej premaknemo na poudarjeno frazo "Slovenija na WWW" (uporabljamo le puščici ↓ in ↑), nato pritisnemo → ali Return, in pred nami se prikaže nova stran:

```
                                                       Slovenija (p1 of 4)


    See the  sensitive map of Slovenia

    and a list of  information sources in (or about) Slovenia.


                             [IMAGE] SLOVENIJA


    This is Slovenia, on the sunny side of the Alps.  Young state, old

    culture, fantastic scenery.


    A nation of two million, with a distinctive and clear identity, which

    has preserved its individuality in this treacherous sub-alpine


-- press space for next page --
  Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
 H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

Ta dokument vsebuje podatke o Sloveniji, ki so še naprej strukturirani v obliki hiperteksta. Pod povezavo "Young" se npr. skriva kratka zgodovina mlade slovenske države. Puščica ← nas vrne v dokument, iz katerega nas je pripeljala (zadnja) uporabljena povezava. Prepuščamo vam, da sami raziščete še ostale povezave[24].

Vsak dokument na Internetu, ki ga lahko dosežemo preko WWW, ima svoje natanko določeno ime, ki ga označujemo s kratico URL (Universal Resource Locator). Sintaksa teh imen je preprosta, a se tukaj vanjo ne bomo spuščali, saj se o tem lahko poučimo kar sami s pomočjo našega novega orodja, če le uporabimo ukaz **h**. Za pokušino le povejmo, kakšen je URL pravkar prikazanega uvodnega dokumenta o Sloveniji: *http://www.ijs.si/slo.html*. Razberemo lahko ime protokola, ki v bistvu pove, da moramo za dostop do tega dokumenta kontaktirati WWW strežnik, sledi običajen "Internet naslov" strežnika, temu pa ime datoteke. Tip datoteke označuje HTML (HyperText Markup Language), jezik, s katerim na WWW opisujemo hipertekst.

[24]Med drugim boste nagrajeni z izčrpnim zapisom o vinih na slovenskem.

Hipertekstovni dokument, ki ga prikaže pregledovalnik, pogosto omenjamo kar kot *stran* v navidez neskončni knjigi WWW (*WWW page*). Pregledovalniku ob zagonu lahko povemo naslov WWW strani, ki naj jo prikaže kot prvo. Če torej poznamo URL nekega dokumenta (npr. *http://www.whitehouse.gov/*), lahko ta dokument dosežemo direktno tako, da namesto običajnega ukaza **www** vtipkamo **www** *http://www.whitehouse.gov/*. Če tega ne storimo, se podobno kot pri Gopherju najprej prikaže privzeti dokument (v našem primeru ARNESova "domača stran za uporabnike *Stenarja*").

Ukazi za terminalsko orientirani pregledovalnik *Lynx*, ki ga uporabljamo na *Stenarju*, so okrajšani na eno samo črko oz. tipko, ki jo moramo pritisniti, podrobneje pa si jih lahko ogledamo, če uporabimo ukaz **h** za pomoč, kot nam kaže spodnja vrstica ekrana. Omenimo še nekaj bolj uporabnih ukazov:

a	–	doda tekoči kazalec (link) med osebne zaznamke (bookmarks)
g	–	direktni skok na podani URL
/	–	išče niz v tekočem dokumentu
v	–	pokaže beležko z osebnimi zaznamki (od tod lahko "skočimo" na označene dokumente)
=	–	pokaže URL trenutnega kazalca
z	–	prekinemo prenos dokumenta (kadar npr. traja brezupno dolgo)

Prekinitev dosežemo tudi s Ctrl–G, poznamo pa že ukaz Ctrl–Z za izstop iz pregledovalnika in tudi ukaz Ctrl–W za osvežitev zaslona smo srečali že vsaj v urejevalniku besedila.

Hipertekst lahko poleg besedila vsebuje tudi sliko in zvok (izraz "tekst" je torej tu že nekoliko preozek). Žal tega ne moremo videti na zaslonu navadnega terminala. Kot vidimo na zgornjem primeru, je pregledovalnik tisto mesto v besedilu, kjer bi morala biti sličica Slovenije, nadomestil z oznako [IMAGE]. Da bi sliko lahko videli neposredno na zaslonu, bi moral biti naš računalnik, ki podpira grafični uporabniški vmesnik (npr. okolje Windows) *neposredno* vključen v Internet, ne pa prek modemske povezave na gostitelja. Kadar je naš računalnik zares del Interneta, lahko uporabljamo veliko "lepše" grafične pregledovalnike (najbolj znana sta *Mosaic* in *Netscape*).

Kadar v navadnem terminalsko zasnovanem pregledovalniku izberemo povezavo, za katero se skriva slika, zvok ali film, nas pregledovalnik opozori, da tega ne more prikazati, in vpraša, ali želimo namesto tega ustrezno datoteko shraniti na disk (**d**ownload). Če delamo to na *Stenarju*, lahko potem prenesemo datoteko na svoj računalnik in si npr. sliko pogledamo s svojim prikazovalnikom[25].

WWW se zna pogovarjati tudi z drugimi strežniki – pravimo, da pozna različne protokole. Tako lahko s pregledovalnikom za WWW pregledujemo tudi Gopherjeve menuje, vzpostavljamo *telnet* povezave in "nabiramo" datoteke po FTP arhivih – kar je pogosto udobneje, kot z uporabo originalnih programov.

[25]Ustrezne programe lahko seveda najdemo v FTP arhivih omrežja.

World Wide Web tako postaja univerzalna informacijska storitev, ki je v navezi z uporabniku prijaznimi vmesniki–odjemalci iz Interneta v kratkem času naredila popularni hit. Začetnika pritegne preprostost uporabe in raznolikost na tako enostaven način dostopnih informacij. Pri tem niti ni tako bistveno, ali imamo možnost uporabe izpopolnjenih grafičnih vmesnikov, čeprav je seveda multimedijski prikaz, vključno z informacijami, ki se morda komu ne zdijo nepogrešljive,[26] tisti magnet, ki privlači nove množice uporabnikov ter javne medije. Čar WWW je namreč tudi v tem, da je mogoče informacijo v lepo obvladljivi obliki spremljati na različno zmogljivih platformah. Bolj važno je to, da ta popularnost spodbuja bliskovito širitev ponudbe informacij in storitev, dostopnih prek WWW strežnikov. Zato kaže slediti dogajanju na World Wide Web, začenši z dokumenti, ki se objavljajo na ARNESovem strežniku za uporabnike.

[26]Na primer slika Clintonovega mačka in posnetek njegovega glasu.

4.5 Kako na Internetu najdemo, kar nas zanima?

To vprašanje je, potem ko smo Internet za silo spoznali in vsaj zaslutili njegove razsežnosti, vsekakor popolnoma naravno. Seveda s tem dregnemo v bistvo, v smisel samega obstoja Interneta, saj se dela z vsemi orodji, ki jih spoznavamo ob tem priročniku, ne učimo kar tako, le da bi jih obvladali. Kmalu (v primeru "prijaznih" storitev, kot je WWW, pa kar takoj) si začnemo postavljati prava vprašanja: "Kako si lahko s tem pomagam? Kje v tej kopici informacij lahko najdem kaj zase?"

Odgovora, ki bi na mah potolažil vse, preprosto ni, kar seveda ni presenetljivo. Dejstvo je namreč, da je kopica informacij nekoliko razmetana. To je cena za to, da je Internet tako velik, poceni (za nekatere celo zastonj), demokratičen in končno, da kot tak sploh deluje. Ogromno dela v Internetu je opravljenega prostovoljno, kar zagotavlja tolikšno količino javno dostopnih podatkov, seveda pa to delo ni sistematično usmerjano niti nujno profesionalno. Centralnega nadzora nad ponudbo informacij torej ni in tudi ni mogoč.

Problem katalogiziranja podatkov je trenutno ena najbolj vročih tem med ponudniki in iskalci podatkov v omrežju. Situacija se nenehno spreminja in priročnik nikakor ni pravšnji medij za njeno podrobno obdelavo. Zato bomo tu podali le nekaj splošnih nasvetov, kako se problema lotimo.

Pri tem se lahko brez škode omejimo na iskanje informacij v WWW. Ne samo, da je to orodje, ki ga največ uporabljamo, ampak pokriva tudi dostop do različnih virov informacij izven samih WWW strežnikov – FTP arhive, *gopherspace*, elektronski imeniki, WAIS indeksirane baze podatkov...

Deloma smo se z iskanjem informacij pravzaprav že seznanili: *Archie* za nas preiskuje sezname datotek po FTP arhivih, *Veronica* pa obvladuje t.i. *gopherspace*. Mimogrede omenimo, da oba lahko na preprost način uporabljamo tudi skozi WWW.

Drug vir informacij so seveda oglasne deske. Na tisoče diskusijskih skupin obdeluje toliko različnih tem, da je med njimi skoraj gotovo tudi predmet našega trenutnega zanimanja. Tam se tudi izmenjujejo izkušnje o ustreznih virih informacij na omrežju. Seznami skupin z opisom tematike se redno objavljajo v *news.announce.newgroups*. Tudi če primerne teme ne najdemo, morda lahko sami sprožimo debato ali vsaj postavimo vprašanje. Vendar s tem nikar ne hitimo, dokler se najprej malo ne razgledamo. Pregledovanje arhivov diskusijskih skupin pa se že spet prepleta z uporabo *ftp* ali kar WWW.[27]

Kako torej raziskujemo Word Wide Web?

[27]Na ARNESovem WWW strežniku lahko npr. najdemo indeksiran arhiv *news.answers* – precejšnja zbirka FAQ dokumentov iz različnih diskusijskih skupin.

4.5.1 Prvi sprehodi po omrežju

Najlepše je, če si lahko na začetku za to vzamemo malce časa. Tudi v klasični knjižnici, ki je dobro urejen vir informacij, se ne bomo znašli prvo minuto, dokler ne vemo, kaj lahko tam pričakujemo, kaj bi pravzaprav radi in kako znamo to izraziti (povedati knjižničarju, uporabljati indekse).

Začnemo se torej "sprehajati" po WWW. Tudi če uporabljamo navaden tekstovno orientiran pregledovalnik, nas večdimenzionalnost hiperteksta, povezana s pestrostjo Interneta, hitro navduši. Med potjo si shranjujemo v beležko (bookmark file) kazalce na zanimive lokacije, dobivamo občutek, kakšne vrste informacij so na voljo in kako so strukturirane.

Že pri tem brskanju naletimo na kup zanimivih reči; odprejo se nam možnosti, ki nanje morda nismo pomislili, hitro naletimo na popularne dokumente, ki se množično uporabljajo kot izhodišča za nadaljnje iskanje.

Začnemo preprosto tako, da sledimo povezavam s svoje domače strani (gl. razdelek o WWW). V enem seznamu je zbrana npr. vsa ponudba informacij, ki jo lahko na ta način trenutno dosežete v Sloveniji, drug seznam ponuja najpopularnejša in najbolj uporabna svetovna izhodišča za začetno in za sistematično preiskovanje WWW in Interneta.

Vsi opisani dokumenti so dovolj razumljivo napisani, da nas smiselno vodijo in nam pomagajo pri izbiri nadaljnjih povezav. Najprej se lahko lotimo bolj splošnih dokumentov, ki so tam našteti: Internet Web Text, Planet Earth Home Page, Global Network Navigator... Vsi ti dokumenti ponujajo obširne uvodne predstavitve Interneta, WWW in razpoložljivih možnosti in storitev, skupaj z izčrpno hierarhijo povezav na najbolj zanimive točke v svetovnem omrežju. Za pokušino pa lahko začnemo tudi z ogledom predstavitve Slovenije na WWW (priporočamo tudi ljubiteljem jedač in pijač ;-) ali pa si ogledamo nekaj priročnikov o Internetu, katerih kopije so na ta način dostopne tudi pri nas.

Spotoma naletimo na dokumente, ki podrobneje opisujejo sam World Wide Web, obstoječe pregledovalnike in jezik HTML, v katerem so opisani dokumenti v hipertekstu.

Nato nas počasi začnejo zanimati konkretne informacije, na katere nismo uspeli naleteti po v svojih prvih sprehodih po omrežju. Iskanje je treba nekoliko sistematizirati.

4.5.2 Načrtno iskanje

Tudi sistematično iskanje lahko začnemo v dokumentih, omenjenih v prejšnjem razdelku, saj vsebujejo obsežne sezname referenc, navadno urejenih po tematiki vsebine.

Poleg doslej naštetih obstaja še množica seznamov informacijskih virov na Internetu, ki zajemajo dokumente na WWW strežnikih, arhive FTP strežnikov, različne podatkovne baze, informacijske sisteme, relevantne diskusijske skupine v USENETu, distribucijske sezname za elektronsko pošto in še kaj. Tudi ti seznami so (zaenkrat) pogosto rezultat prostovoljnega dela in so različno organizirani in seveda pokrivajo nekoliko različne množice informacijskih virov, čeprav velja, da se dobri in popularni viri informacij prej ali slej pojavijo v vseh redno vzdrževanih seznamih.

Seznami so urejeni so glede na vsebino, način dostopa ali geografsko.

Nekaj najpopularnejših izhodišč je zbranih v dokumentu, ki ga opisujemo ("izhodiščne točke ..."), med njimi WWW Virtual Library, Yahoo Guide, Virtual Tourist in Yanoff's List.

Ker so seznami različno urejeni, izbira pa se zaradi hierarhične ureditve kar hitro zoži, se lahko "sprehodimo" po nekaterih bolj obetavnih drevesih in poskusimo najti željeno informacijo.

Če iščemo bolj konkretno (in znamo svoje želje izraziti), so nam na voljo različni iskalni avtomati, ki opravljajo podobno funkcijo kot *Archie* in *Veronica* (vključno z njima, seveda). Ti programi tako ali drugače zbirajo informacije o dokumentih v omrežju in ustvarjajo zbirke podatkov. Takšno zbirko podatkov lahko preiskujemo s pomočjo ključev oz. ključnih besed (keywords). Uporaba je navadno zelo preprosta: ko izberemo aplikacijo, na zahtevo podamo enega ali več ključev in sprožimo iskanje. Podrobnejaša navodila, kako sestaviti učinkovito "vprašanje" za tak avtomat, so navadno zraven ustreznega pogovornega okenca oz. na domači strani aplikacije.

Tudi iskalni avtomati se po metodah in učinkovitosti med seboj razlikujejo, trenutno morda najpopularnejša sta *Lycos* in *WebCrawler*, prihajajo pa že nove generacije tovrstnih programov (*Harvest*, *WHOIS++*). *Lycos* in *WebCrawler* najdemo med "izhodišči" skupaj z nekaj drugimi avtomati ter kazalcem na seznam cele kopice podobnih orodij, zbrane na enem mestu.

Vse to so seveda zgolj – bolj ali manj učinkovita – orodja. Kot povsod pa velja, da pridobimo učinkovitost iskanja relevantnih informacij z izkušnjami ter sprotnim spremljanjem tako omrežne informacijske ponudbe za ustrezna področja kot tudi razvoja orodij za dostop do teh informacij. Živimo v času, ko se oboje spreminja s hitrostjo, ki ji je kar težko slediti in ko nekatere informacije lahko hitro zastarijo.

4.6 Epilog: kako se znajdemo od tu naprej?

S priročniki na papirju je nemogoče zasledovati tako hitro spreminjajoč se organizem, kot je Internet. Zato bo v prihodnje več pozornosti namenjeno elektronskemu obveščanju. Ta priročnik vam – upamo – nudi vsaj toliko osnovnega znanja, da orodja na omrežju začnete uporabljati, raziskovati, da lahko berete elektronska obvestila in druge dokumente v elektronski obliki (hipertekst), nanje reagirate, sprašujete in se konstruktivno vključite v izboljšanje informacijske ponudbe na omrežju.

Priporočamo vam, da ne prezrete elektronskih obvestil, ki se kdaj pa kdaj pojavijo ob priključitvi, da redno berete prispelo elektronsko pošto, predvsem pa redno spremljate dogajanje na ARNESovih domačih dokumentih, saj bodo vse zanimivosti in novosti, o katerih vas bomo hoteli obvestiti, prej ali slej objavljene na našem skupnem strežniku, kjer lahko tudi sami predstavite svoje delo.[28]

Podrobnejše napotke za uporabo posameznih storitev s pomočjo programov na *Stenarju* ter dodatnih ukazov sistema VMS lahko poiščete z izkoriščanjem pomoči, ki jo nudijo sami programi oz. operacijski sistem, ali pa poiščete ustrezno literaruro.

O Internetu in njegovih storitvah je napisano mnogo knjig. Za zelo poljuden in poučen uvod v Internet in njegovo kulturo, ki pa nam daje tudi dovolj splošne osnove za začetno uporabo, bi lahko izbrali npr. *The Internet Companion* avtorice Tracy LaQuey (založba Addison-Wesley, trenutno najnovejša je druga izdaja iz l. 1994). Ena najboljših knjig o Internetu pa je *The Whole Internet – User's Guide and Catalog* (Ed Krol, založba O'Reilly, trenutno druga izdaja, l. 1994). V tem besedilu je imenitno združen lahko berljiv poljuden pristop za začetnike s podrobno obravnavo storitev Interneta, konkretnih opisov popularnih odjemalcev ter kopico koristnih nasvetov, hkrati pa knjiga vsebuje izbor zanimivih virov informacij na Internetu. Ta knjiga bi morala biti dostopna vsakomur, ki želi spoznati in uporabljati Internet.

Seveda pa je množica besedil, uvodnih esejev in priročnikov, na voljo v različnih virih na omrežju. Nekaj znanih priročnikov o Internetu, kompletni seznami FAQ (Frequently Asked Questions), dokumenti, ki definirajo delo na Internetu ter arhivi USENET News so na voljo kar na ARNESovem FTP strežniku, najdemo pa jih tudi na WWW straneh, kjer jih lahko preiskujemo s pomočjo ključnih besed. Že ti dokumenti več kot zadoščajo za zavidljiv nivo splošnega razumevanja Interneta in uporabe njegovih storitev.

Ta priročnik je v nečem soroden Internetu: široko zastavljen, ne da bi bil kakorkoli popoln, nekaj povedanega o tem, nekaj o onem, v upanju, da bo komu prišlo prav. Veseli bomo, če nam boste opozorila na morebitne napake v priročniku ali pripombe glede manjkajočih informacij poslali na naslov *prirocnik@arnes.si*.

[28] Več o tem pa v dokumentu "Kako postaviti svojo domačo stran na WWW", ki ga najdete na dnu domače strani za uporabnike *Stenarja*.