

CALCULATOARE

PERSONALE

**1 / 91
(4)**

IANUARIE

39 Lei

Revistă lunară editată de Micro ATCI Tîrgu Mureş

Imprimanta - sau fără hîrtie nu se poate !

Utilizarea rațională a memoriei

Baze de date

DATENBANK

Unix și virușii

Evenimentul PS/1

Anchetă - vreau să cumpăr un AT !

if if if if if

**HARDWARE-UL DUMNEAVOASTRĂ ESTE TOT ATÎT DE BUN
CÎT ESTE ȘI PARTENERUL DE SERVICE
PE CARE L-AȚI ALES !!**

SIRECA - Secția de întreținere și reparații echipamente de calcul și automatizări din cadrul întreprinderii INFO-HARD execută cu personalul său calificat întreținerea și repararea echipamentelor de calcul electronice și automatizări din gama:

- sisteme de calcul FELIX-256-1024 , FELIX-5000;
- minicalculatoare CORAL , INDEPENDENT;
- videoterminale VDT , alfagraf;
- stabilizatoare de tensiune de 10 KVA si 20 KVA;

De asemenea , prin specialiștii din laboratorul central,execută:

- lucrări de repunere în funcțiune sisteme de calcul;
- transformări de echipamente;
- reparații de plachete;

Telefoanele de la sediul central și de la filialele din țară sînt:

-sediul central:

BUCUREȘTI , str.Cluceru Sandu Nr.2 , sector 2 , cod 73156

telefon:90/276330-int.3

sau 90/273185

-puncte de lucru:

ARAD	966/14173
BACĂU	931/42895
BAIA-MARE	994/34983
BRAȘOV	921/10430-int.226
BRĂILA	946/35863
CLUJ	951/30677
CRAIOVA	941/32934
FOCȘANI	939/22087
IAȘI	981/37947
ORADEA	991/42761
PLOIEȘTI	971/44778
PITEȘTI	976/35492
SIBIU	924/16502
TIMIȘOARA	961/16599
GIURGIU	912/15021
RÎMNICU VÎLCEA	947/11571
TÎRGOVIȘTE	926/11951-int.25

Pentru informații suplimentare specialiștii noștri vă stau la dispoziție.

if

Angajament

revistă de informatică
editată de firma Micro ATCI

Director: ing. Dumitru Dunca

La realizarea acestui număr au
colaborat:

ing. Attila Darvas,
ing. Zoltán Egyed,
ing. Iosif Fettich,
ing. Alin Flaidăr,
ing. Ingrid Maier,
ing. Romulus Maier,
ing. Emil Palade

Desene: ing. Ștefan Doboș

Tiparul: tipografia Tîrgu Mureș

Revista apare lunar.

Preț: 39 lei per exemplar, la care
se adaugă eventualele taxe de
expediție.

Adresa și telefonul redacției:

Micro ATCI,
RO-4300 Tîrgu Mureș,
C.P. 64, O.P. 1,
tel. 954/17024, fax 954/35208.

Manuscrise originale sau listing-uri
de programe sînt primite cu
plăcere de redacție, cu condiția să
nu fi fost publicate și în altă parte.
Prin expedierea unui manuscris pe
adresa redacției, autorul consimte
implicit la publicarea materialului
său în cadrul revistei. Onorariul se
negociază cu directorul'.
Materialele nepublicate nu se
înapoiază și nu se rețin.

Revista noastră vă oferă spațiu
pentru reclamă și publicitate la
următoarele tarife:

1 cm ² (alb-negru)	30 lei,
1 cm ² (alb-negru și o culoare suplimentară)	40 lei,
1 cm ² color	60 lei.

Doritorii sînt rugați să ia legătura cu
redacția.

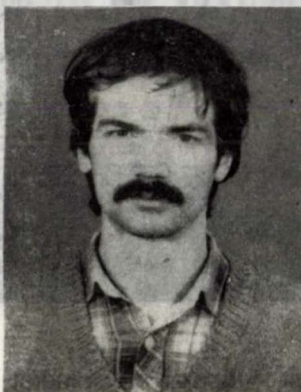
Subsemnații, echipa redacțională a revistei **if**, acum la început
de an, ne angajăm solemn în fața cititorilor noștri să facem totul ca
în ciuda greutăților, piedicilor, vicisitudinilor soții, etc., etc., revista
if să meargă mai departe... Piei satană !

În mod serios nu vă promitem nimic, dar vă împărtășim cîteva
dintre dorințele și speranțele noastre. Dorim ca revista noastră,
care de-acum sperăm că este și a d-voastră, să apară lunar.
Sperăm că așa va fi, dar nu sîntem siguri. În privința acestui prim
număr din '91 Moș Crăciun și Moș Gerilă au stabilit un consens și
i-au dat un brînci pe ghețușul sărbătorilor de iarnă sporind și mai
mult întîrzierea cu care va apare pe piață. În momentul de față cea
mai dificilă problemă cu care ne confruntăm s-a putea formula în
modul următor: "Dacă se predă la tipografie un material la o dată
prestabilită x și se promite tipărirea materialului pînă la data y, care
este data z la care va putea fi ridicat materialul?". Dacă ne-ați putea
comunica adresa vreunei babe care știe să ghicească în bobi, sau
în orice altceva, care ne-ar putea ajuta în această problemă, v-am
fi recunoscători.

Dorim, de asemenea, ca structura și conținutul revistei să se
îmbunătățească și îmbogățească continuu. Sîntem siguri că, dacă
veți continua să ne jutați cu aprecierile și criticile d-voastră, așa se
va întîmpla. Iar dacă vom mai și greși sperăm că ne veți ierta. Este
foarte greu să mulțumești pe toată lumea. Ni s-a întîmplat, și nu o
singură dată, ca un articol apreciat de mai mulți cititori să nu placă
altor cititori. Nivelul articolelor reprezintă o altă piatră de încercare.
Unii cititori ne reproșează că nu "urcăm destul de sus", în timp ce
alții se plîng de faptul că nu pornim "mai de jos". Pluta conceptului
redacțional plutește astfel între Scylla și Charybda, dar ținînd bine
cîrma sperăm că ... "E la nave va !".

Tematica abordată, începînd cu acest număr, s-a diversificat,
rămînem însă credincioși și pe mai departe temelor de fond. În
acest număr o să puteți arunca o privire asupra familiilor de
imprimante și asupra sistemelor de gestiune a bazelor de date
(SGBD). În privința bazelor de date ne simțim datori să vă avertizăm
că doar alegerea unui SGBD, oricît ar fi acesta de bun, nu este
suficientă. Dacă în implementarea și întreținerea bazei de date vor
lipsi organizarea, disciplina, rigurozitatea, nu veți obține decît un
"cimitir de date".

Sperăm ca informațiile din articolele publicate să nu vă fie doar
pe plac ci și utile. Dacă așa se va întîmpla înseamnă că osteneala
noastră nu este în zadar.



ing. Iosif Fettich



ing. Romulus Maier



Everything O.K., Mr. President - tocmai am infectat rețeaua de calculatoare adversă
cu noul virus de la Pentagon ...

Cuprins

Editorial pag. 1

Știri

- Trei noi PC-uri marca Apple pag. 4
- Placă TV pentru PC-uri pag. 4
- Cartelă PC-Fax pentru Est pag. 4
- Afacerea Borland versus Lotus :
o poveste urâtă, tristă, regretabilă ! pag. 5
- Specificații pentru administrarea
LAN-urilor pag. 5
- Standard multimedia pag. 5
- Urmașul lui Intel pentru SX pag. 6
- Discuri flash-memory pag. 6

Magazin

"Destinse, fără frică și vesele" pag. 7

Multimedia

A noua, vă rog ! pag. 8

Simfonia a noua a lui Ludwig van Beethoven, în cuvinte și imagini, cu sunet digital Hi-Fi : noul eveniment multi-media pentru Mac.

Vă prezentăm

Evenimentul PS/1 ? pag. 10

Un microcalculator puternic la îndemîna tuturor. O victorie à la Pyrrhus pentru Apple, care-și vede strategia convivialității preluată de IBM. Și totuși, ce aduce nou PS/1?

Virușologie

Unix și virușii pag. 12

- "Orange Book"- ce înseamnă "sigur"? pag. 15

Hardware

Imprimanta
- sau fără hîrtie nu se poate ! pag. 16

- 7 sfaturi pentru o instalare corectă
a imprimantei pag. 18
- 7 sfaturi pentru achiziționarea
de imprimante. pag. 18
- Cele mai cunoscute imprimante pag. 19

Cum se lucrează cu Extended Memory

Utilizarea rațională a memoriei
calculatoarelor personale pag. 20

Rețele

10 Base-T - Strategii de implementare pag. 26

10 Base-T este noul standard pentru Ethernet pe cablu bifilar neecranat. Implementarea standardului, cu multe funcțiuni noi, este foarte simplă, dacă se respectă anumite reguli.

Software

Baze de date pag. 29

- dBase IV 1.1 - revenire reușită pag. 30
- Paradox 3.5 - o punte între
două lumi pag. 31
- R:Base 3.1 - examen de maturitate pag. 33

Compilatoare de baze de date pag. 35

- Clipper 5.0 pag. 35
- Foxpro 1.0 pag. 36
- Clarion 2.1 pag. 37

Aplicații distribuite pag. 38

Practica

Se poate formata și altfel... pag. 44

Pe un AT cu o unitate floppy de tip MF, utilizatorul nu poate produce pe dischete simple (2S,DD) dischete de 420 kbyte, 720 kbyte sau 800 kbyte. O alternativă o oferă programul "mformat".

Rubrici

- Caseta redacției pag. 1
- Caricatura pag. 2
- Prețuri, prețuri... pag. 52
- Anchetă - Vreau să cumpăr un AT ! pag. 53
- Poșta redacției pag. 54
- Mica publicitate pag. 54

Trei noi PC-uri marca Apple

Cu cele trei noi calculatoare Macintosh prezentate: "Classic", "LC" și "IIsi", Apple încearcă pentru prima dată să câștige cumpărători în domeniul costurilor reduse. Apple oferă un model pentru începători, în versiunea cea mai ieftină, la același preț la care IBM oferă modelul PS/1, adică aprox. 2000 mărci. Macintosh Classic este oferit, ca și predecesorul Mac Plus, într-o carcasă integrată în care se află atât calculatorul cât și monitorul. Procesorul este un Motorola 68000 la o frecvență de tact de 8 MHz, iar unitatea floppy standard este "Superdrive", care poate prelucra dischete de 3,5" în format MS-DOS, OS/2 și Pro DOS (Pro DOS este formatul dischetelor vechiului Mac Plus). Sistemul mai conține: 1 MByte memorie de lucru, șase interfețe, facilități de rețea, sistemul de operare V.6.06, programe utilitare, tastatură și mouse. O limitare o constituie lipsa slot-urilor interne, extensiile fiind posibile numai prin intermediul interfețelor SCSI. Modelele Mac Plus și Mac SE nu vor mai fi produse deoarece Classic urmează să le înlocuiască pe amândouă. Classic va fi oferit și într-o versiune cu 40 MBytes harddisk și cu 2 MBytes RAM și va costa 3000 mărci.

Cu o clasă mai sus se situează Macintosh LC, care a fost primul model Apple prezentat într-o carcasă desktop de 8 cm înălțime. LC are un microprocesor 68020, 16 MHz, dar nu are nici coprocesor și nici soclu pentru acesta. Spre deosebire de Classic, LC are un slot de extensie cu acces direct la bus-ul CPU. Slotul

direct al lui LC nu este totuși compatibil cu slotul direct 68030 al lui Mac SE/30, dar ar putea fi folosit pentru o plachetă pentru un procesor 68030, care să poată utiliza modul de tratare a memoriei al de mult anunțatului sistem 7.0. Standard, cu ajutorul lui "Voice-Mail", LC este capabil să prelucreze semnale audio, preluate prin intermediul unui microfon livrat odată cu sistemul. Spre deosebire de redarea pură a sunetului în acest caz pot fi memorate documente sau știri (de ex. în rețea) cu comentarii și apoi pot fi apelate. În mod standard LC dispune de: unitate floppy Superdrive, șapte interfețe, un slot, 2 MBytes RAM, 40 MBytes harddisk, tastatură și software sistem. Memoria de lucru poate fi extinsă pînă la 8 MBytes. Configurația, cu un monitor de 12", costă aprox. 4900 mărci.

Ca model de trecere spre cunoscutele calculatoare Mac II, este gândit noul model Macintosh IIsi, care dispune de un procesor Motorola 68030, 20 MHz, și este oferit în trei versiuni: IIsi 2/40, IIsi 5/40 și IIsi 5/80. Cele trei variante se deosebesc doar în ceea ce privește capacitatea memoriei de lucru și a harddisk-ului. La toate trei le sînt comune: o posibilitate de extindere a memoriei pînă la 17 MBytes, unitate floppy Superdrive, microfon, software sistem, ca și opt interfețe și un slot pentru o plachetă Nubus sau o plachetă de extensie 030. IIsi 2/40 (2 MBytes RAM și 40 MBytes harddisk) costă 8.000 mărci, IIsi 5/40 costă 9.000 mărci, iar IIsi 5/80 costă 10.000 mărci.

Placă TV pentru PC-uri

O firmă canadiană nouă, AView Technology, și-a planificat introducerea pe piață a unei plăci pentru calculatoarele personale compatibile IBM, care să permită recepționarea imaginilor TV pe monitorul calculatorului. Placheta "Desktop TV-Board" este o plachetă pe 8 biți pentru monitoare multi-frecvență. Placheta va putea fi activată, prin intermediul unui soft propriu, sub sistemul de operare DOS, sau Windows, va putea prelucra semnale după standardul american NTSC, dar și semnale video, și va permite comutarea între modulele

de lucru televizor și monitor PC. Prin intermediul software-ului sonorul, contrastul, luminozitatea și selecția canalului, vor putea fi comandate de la tastatură. Prima versiune a plachetei lucrează numai cu monitoare multi-frecvență în domeniul 15,7 pînă la 31,5 kHz și numai cu plăci grafice EGA și VGA. O versiune ulterioară va lucra cu monitoare VGA pure și cu plăcile grafice corespunzătoare. Ambele plăci vor costa în jur de 395 de dolari și vor fi disponibile începînd din februarie.

Cartelă PC-Fax pentru Est

Cartela pentru PC-uri "JT-Fax", a producătorului Interquad, a fost concepută în mod special pentru liniile telefonice proaste din Germania de Est. În plus cartela de 8 biți pentru PC-urile IBM XT/AT, este livrată cu un adaptor extern pentru operarea fără probleme pe PC-urile Robotron de 16 biți Eser EC1834. Cartela fax respectă normele grupei III după CCITT (pînă la 4800 caractere/secundă, cu autofallback pînă la

1200 de bauds) și costă inclusiv software-ul, manualul și cablul de cuplare la telefon 799 mărci, fără impozitul pe valoarea adăugată. Pentru desfacerea plăcii răspunde firma Beeking Consulting din Kranenburg. Cartela JT-Fax a primit avizul de utilizare la 24.8.90 din partea Institutului pentru poștă și medii de comunicare la distanță, din acea vreme.

Afacerea Borland versus Lotus : o poveste urâtă, tristă, regretabilă !

Informatica nu mai este de mult domeniul programatorilor izolați. Încetul cu încetul geniile în micro au cedat locul oamenilor de afaceri. Telefoanelor de felul " Hey man ! Noaptea asta am găsit o chestie grozavă..." le urmează înregistrările legale, procesele verbale, certurile avocaților

Filmul afacerii: Joi 28 iunie 1990, la Federal District Court din Boston se stabilește că societățile Paperback Software și Stephenson Software Ltd au încălcat drepturile de copyright ale pachetului Lotus 1-2-3 în comercializarea produsului lor VP-Planner. Vineri 29 iunie 1990, Borland depune plângere împotriva lui Lotus la District Court, California de Nord, cu scopul de a obține o hotărâre judecătorească declarativă specificând că pachetele de programe de calcul tabelar Quattro și Quattro-Pro nu contravin drepturilor de copyright ale lui Lotus . Luni 2 iulie, Lotus angajează o acțiune juridică în statul Massachussets împotriva lui Borland pentru violare de copyright . Lotus pretinde că interfețele utilizator ale lui Quattro încălcă copyright-ul lui Lotus 1-2-3 . Lotus își explică poziția prin Thomas Lemberg, vice-președinte și secretar general al firmei: "Lotus 1-2-3 este produsul nostru cel mai important, în care am investit resur-

se considerabile pentru satisfacerea necesităților crescînde ale clienților noștri. Față de clienții noștri, față de angajații și acționarii noștri avem responsabilitatea de a ne proteja produsul împotriva oricărei copii ilicite. Noi considerăm în special că dezvoltînd Quattro, Quattro Pro și SCO Professional, Borland International și Santa Cruz Operation au copiat comenzile și structura meniurilor lui Lotus 1-2-3 practic în totalitate , " De cealaltă parte Borland afirmă că Quattro Pro este un program original și nu contraface în nici un caz Lotus 1-2-3 . În ceea ce privește presa americană, ea a adoptat o poziție mai degrabă severă în ceea ce-i privește pe protagoniștii afacerii : " Quattro Pro nu copiază Lotus în maniera lui Paperback, dar oferă efectiv posibilitatea utilizării de meniuri și de comenzi Lotus 1-2-3 ca o alternativă la propriile sale comenzi . Numeroși critici consideră Quattro un program superior lui 1-2-3 ; face tot ceea ce face și Lotus, dar are nevoie de mai puțină memorie, are grafice mai bune și este efectiv vîndut la un preț de 99\$ spre deosebire de 495\$ pentru Lotus 1-2-3 ." Jim Seymour, PC Week, 16/07/90

" Dacă nu puteți cîștiga competiția prin preț, calitate, inovație sau servicii puteți întotdeauna recurge la o ultimă posibilitate-

te: angajați un avocat și prezentați-vă în fața instanței. Este ceea ce se învață acum în școlile de comerț, este lecția de care și-a amintit Lotus cînd s-a găsit în fața unei competiții dure cu admirabilul nou program de calcul tabelar al lui Borland, Quattro Pro ." T. R. Reid, Washington Post, 16/07/90

"Se atinge uneori un stadiu cînd protejarea soft-ului pretinde mai multe eforturi decît cele necesare pentru a crea ceva nou. Lotus a atins acest stadiu și l-a și depășit, presupunînd lesigur că-i-au rămas destule resurse pentru a face ceva nou. Judecînd însă după cele petrecute, s-ar părea că această ultimă prezumție nu este credibilă ." Jerry Pournelle, Info World, 23/07/90

Este dificil de înțeles un proces pentru o interfață, într-un moment în care piața se deplasează spre o uniformizare a meniurilor în beneficiul utilizatorului. Din fericire cei doi protagoniști nu și-au portat încă produsele sub Windows. În ziua cînd o vor face riscă să devină piinea avocaților. Cel puțin se poate spune că "afacerea" provoacă o adevărată hîduială în presa americană. Oamenii celor două societăți beneficiază de o publicitate dificil de obținut în condiții normale. Recurgerea la justiție prezintă astfel noi virtuți .

Specificații pentru administrarea LAN-urilor

IBM și 3Com au anunțat că specificațiile elaborate împreună pentru administrarea rețelelor locale (LAN) au devenit publice. Aducerea la cunoștința publică a fost precedată de o evaluare a lor din partea echipelor tehnice ale firmelor Banyan, Microsoft, Novell și SCO. Specificațiile pentru administrarea LAN-urilor în medii eterogene oferă structura de bază pentru elaborarea unor produse care să poată lucra cu o multitudine de sisteme de operare. Sînt disponibile și diferite interfețe pentru comunicația între sisteme de operare și adaptoare de rețea. Interoperabilitatea, scopul final al specificațiilor, este un element cheie al prelucrării automate a datelor în viitor.

Standard multimedia

IBM și Microsoft, împreună cu alți șapte producători, doresc să impună un standard comun pentru tehnologia multimedia. Multimedia trebuie să contopească într-un tot unitar domenii diferite, cum ar fi: muzică, video, calculatoare sau televiziune. Observatorii pieței așteaptă de la această tehnologie un nou impuls de creștere pentru industria de calculatoare a viitorului. IBM a trasat ca urmare primele linii directe producătorilor, pentru proiectarea de software multimedia sub OS/2 și Windows 3.0. După cum anunță IBM specificațiile prevăd un format general valabil de conversie fișiere pentru imagine, text și prelucrarea sunetului ca și o standardizare a componentelor hard core-spunzătoare. Michael Braun, vicepreședinte la IBM este optimist: "Noi credem că multimedia este o mare provocare și șansă pentru industria calculatoarelor".

Urmașul lui Intel pentru SX

Intel, SUA, a creat un nou microprocesor, "386SL", destinat în mod special calculatoarelor portabile și altor calculatoare de format mic. După datele furnizate de Intel, urmașul lui 80386SX este mai rapid și este dotat cu mai multe funcții.

Noul microprocesor "386SL", și chip-ul de I/O corespunzător "82360", sînt produse numai în versiunea 20 MHz și concurează chiar și cele mai rapide microprocesoare "386SX" produse de Intel. Desigur, Intel a integrat în acest procesor și funcții suplimentare. Acesta a fost realizat cu un nou sistem de comandă a întreruperilor (System Management Interrupts - SMI), un nivel de întreruperi pur

hard cu un nou spațiu de adresare. Modul de lucru SMI este invizibil pentru sistemul de operare și pentru aplicațiile soft și controlează operațiile interne, periferia, viteza CPU și funcțiile de protecție împotriva căderilor de curent. Intel afirmă că, în ciuda funcțiilor suplimentare, care necesită și ele timp, un 386SL este mai rapid decît un 386SX la același tact de 20 MHz. Procesorul SL dispune de un controller suplimentar de memorie cu un spațiu de adresare de 32 MBytes, un LIM/EMS 4.0, bus AT și controller cache și susține coprocesorul 387SX. Intel livrează procesoarele 386SL și 82360 și separat. Prețurile, pentru o comandă de 100 de bucăți se ridică la 176 dolari

pentru 386SL și 45 dolari pentru 82360.

Setul de procesoare 386SL și 82360 îi pune în dificultate pe ceilalți producători, care oferă funcțiile suplimentare ca și componente proprii. Pentru moment doar Intel oferă un procesor care să aibă aceste funcții integrate. După cum afirmă Intel această soluție este mai bună decît cele oferite de concurență, deoarece în cazul lui SL procesorul nu mai este încărcat cu sarcini suplimentare de supraveghere a memoriei și a întreruperilor și de coordonare cu sistemul de operare și cu aplicațiile, ci aceste funcții sînt integrate și nu mai au nevoie de control extern.

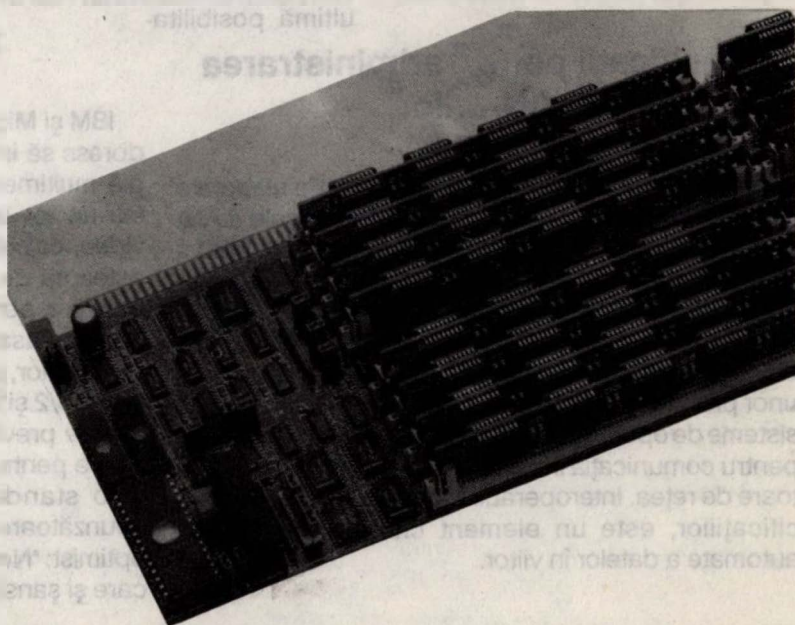
Discuri flash-memory

Dezvoltări noi în tehnologia chip-urilor de memorie au făcut posibilă apariția discurilor flash-memory. Firma SCM (SCM Microsysteme GmbH, Bertha-v.-Suttner-Weg 1, D-8033 Martinsried b. München, Tel. 089/8561363, fax 089/8561412) prezintă primele flash-memory discuri de 16 Mbyte (utilizînd module SIMM de 1 Mbyte) sau 32 Mbyte (utilizînd module SIMM de 2 Mbyte). Un flash-memory disc este de fapt o plachetă (cartelă), PC-XT și AT compatibilă, folosibilă cu un bus de tip ISA sau EISA, putînd opera transferuri pe 8 sau 16 biți, dotată cu 16 socluri SIMM, constituindu-se într-un "hard-disk" de memorie nevolatilă.

Flash-diskul de memorie nevolatilă face posibilă o mărire a vitezei de 100 de ori (!) în comparație cu un harddisk obișnuit. Deoarece nu există uzură mecanică, și durata de viață crește cu ordine de mărime, astfel încît devine posibilă utilizarea în aplicații industriale, unde apar praf, vibrații și temperaturi ridicate. Un driver DOS (soft) face posibilă o integrare simplă în PC. E disponibilă și o versiune autoboot (de pe care se încarcă direct sistemul la pornire). Flash-diskul poate fi îndepărtat din sistem fără ca datele stocate pe (în?) el să sufere; cu toate acestea, nu este necesară nici un fel de baterie. Flash-diskul poate fi citit/scriș de sub DOS și emulează un harddisk. EPROM-diskul - altfel întru-totul similar flash-diskului - poate fi programat (scriș) numai o dată. O nouă programare este posibilă numai prin schimbarea modulelor SIMM (ceea ce oferă, pe de altă parte, o excelentă protecție împotriva virușilor).



Modul SIMM de 2 Mbyte, scara 1:1



"Destinse, fără frică și vesele"

În München s-a înființat prima școală de calculatoare pentru femei- fără bărbați, căci "chinezeasca" lor în calculatoare deranjează climatul de studiu.

"Emancipare! Acesta a fost primul cuvânt pe care fiul meu l-a citit din acest program". Tînărul de 18 ani a răsfoit programa școlii de calculatoare pentru femei, unde mama lui a petrecut sfîrșitul de săptămînă - trei ore vineri și cîte opt ore sîmbătă și duminică. Pe programă era scris: "Bazele prelucrării electronice a datelor + MS-DOS (inclusiv ergonomie practică)".

În martie 1990 a apărut ideea deschiderii unei școli comerciale de calculatoare pentru femei. "Nu am vrut să dezvolt un proiect model, la care an de an trebuie să alergi după mijloace de transport", povestește Angelika Huber faza de început. Deci s-a înscris la un curs de întreprinzătoare și a ajuns la concluzia: "sau e greu sau e foarte greu". A înălțat în cîteva luni școala din pămînt. În august a preluat încăperile, în septembrie s-a sărbătorit inaugurarea.

Întreprinzătoarea în vîrstă de 30 de ani este de profesie comerciant bancar. Ea a predat generației tînere de bancheri, apoi a studiat Germanistica, s-a lansat ca autoare, și a predat în diferite instituții cursuri de prelucrare electronică a datelor. Aici a fost deranjată de faptul că, deși femeile sînt cele care vin în primul rînd la cursurile de prelucrare electronică a datelor, acestea sînt concepute pentru femei. "Cîteodată la un curs aveam cîte 15 femei la un bărbat. Chiar și în acest raport bărbatul a reușit să domine cursul cu cunoștiițele lui și să-mi vîre pe gît o discuție despre un 386 SX". Această experiență regretabilă a fost confirmată și de cercetarea efectuată în cadrul cursurilor de calculatoare ținute de femei

pentru femei: aici printre altele se afirmă, că toate participantele consideră că o învățare efectivă se poate obține doar pornind de la o atmosferă "lejeră, fără frică și veselă". În continuare studiul arată că motivul acestei atmosfere de lucru pentru multe cursante era faptul că "la acest curs au participat doar femei". Urmarea: "Au considerat cursul ca un spațiu liber, unde au putut învăța fără stress și fără să fie mînate de la spate de un bărbat, care din principiu le știe pe toate mai bine".

La fiecare sfîrșit de săptămînă opt femei stau în încăperea de școlarizare. Fiecare are propriul ei loc la calculator. "Femeile grase trebuie să stea la perete", se distrează o participantă de locul îngust al suprafețelor de pe mesele de calculător. "Locul este prea mic pentru un bloc-notes", observă alta. Lipsese și podeaua antistatică. Cum trebuie să arate un loc de muncă atrăgător pentru om, nu pentru tehnică? Ce șanse sînt de a se împotrivi anumitor pericole? Se face un schimb de informații. "Deunăzi am citit un articol interesant pe această temă. Am să vi-l trimit"- femeile nu se tutuiesc.

Șase din cele opt cursante nu au mai lucrat niciodată pe un calculator înainte de acest sfîrșit de săptămînă. Două eleve au mai participat la alte cursuri și le pare rău și în ziua de azi: "am petrecut un week-end întreg acolo și în final n-am știut nici cum se introduce o dischetă în unitate" ne spune una dintre ele. Cealaltă a plătit pentru un week-end aproape 1000 de mărci și a fost înscrisă la un curs, care făcea trecerea pentru specialiști de la Word 4.0 la 5.0, "cu toate că știau că sînt începătoare". N-a priceput nimic, dar a fost o învățatură pentru ea.

Doar în München aproximativ 180.000 de locuri de muncă la calculator sînt ocupate de femei. Dar Irmgard Kerschbaum, care se ocupă la Siemens filiala din München de egalitatea șanselor femeilor în meserie, este de părere că aceste cursuri pentru femei "au puțin sens". Cu toate că această problemă se discută mereu, noi "nu oferim asemenea cursuri". Motivația: cursurile pentru femei nu sînt rentabile. În cadrul filialei germane a celui mai mare producător de calculatoare IBM modul diferit de apropiere de tehnologia calculatoarelor este considerată ca o problemă specifică vîrstei și nu sexului.

Referentul de presă Cornelia Weiss consideră cursurile pentru femei ca o adevărată "discriminare a femeilor". Tocmai această școlarizare separată va întări ideea preconcepțată a bărbaților, cum că femeile ar fi mai proaste.

Elevele școlii de calculatoare pentru femei discută această problemă foarte des. Întrebarea îngrijorătoare apare des: oare acest certificat va fi recunoscut în cadrul solicitărilor de angajare? "Dacă se leagă cineva de faptul că în certificat scrie ceva despre femei, atunci asta este problema sa și nu a mea", obiectează o cursantă. În schimb, atunci cînd șeful de personal pune problema emancipării specialiștilor de calculatoare, femeia va trebui să renunțe la job. Angelika Huber vede soluția acestei întrebări mai degrabă pragmatic: "dacă se află pe piață că noi realizăm aici o specializare bună, atunci important nu va fi dacă femeia a studiat la o școală pentru femei, ci cît de capabilă este".

(continuare în pag. 9)

(COMPUTER LIVE 12/90,
Veronika Mirschel)

A noua, vă rog !

Simfonia a noua a lui Ludwig van Beethoven, în cuvinte și imagini, cu sunet digital Hi-Fi : noul eveniment multi-media pentru Mac.

Sloganul actual în branșa calculatoarelor este multi-media. Prin această denumire seacă se înțelege îmbinarea tuturor mediilor existente, ca: vorbire, muzică, imagine și video, într-un supermediu. Tehnologia multimedia se folosește în momentul de față mai ales în domeniul școlarizării. "Elevii" cunoscători cu succes munți întregi de știință prin metoda do-it-yourself (fără singur) în mod interactiv, adică în dialog cu ecranul. Cu ajutorul tastaturii sau a unei zone sensibile la atingere, reprezentată printr-un buton pe ecran, ei își conduc singuri "excursia de date".

Acest mod de transmitere a cunoștințelor nu numai că face mai multă plăcere, dar este și mult mai eficient. Nu doar pentru că învățând astfel cunoștințele se fixează mai bine, ci și pentru că utilizatorul unui astfel de sistem își poate fixa singur ritmul de învățare. Nu din întâmplare această metodă este introdusă deja de o serie de întreprinderi în cadrul școlarizării personalului cu ajutorul calculatorului (Computer Based Training CBT).

Important pentru înțelegerea sensului și construcției unei astfel de producții multi-media în scopuri de CBT este faptul că un astfel de sistem are avantajul de a putea oricând rechema informații despre fiecare noțiune superioară. Multimedia este pe cale de a se dezvolta tot mai mult și în domeniul calculatoarelor. Până în prezent au fost arătate în public doar superproducții, ticsite cu video, animație, muzică, zgomote și vorbire, probabil pentru a etala toate posibilitățile. O astfel de furtună de informații scurte cere o dotare costisitoare hard. Memoria principală a calculatorului trebuie să fie de mai mulți MBytes, și de cele mai multe ori sînt necesare și hard-discuri de mare

capacitate (mai multe sute de MBytes). Foarte des sînt folosite picupuri de imagini (răspîndite mai ales în SUA ca playere pentru discuri video).

Calculatorul multi-media cel mai bun în acest moment este considerat a fi Apple Macintosh, datorită posibilităților de grafică și sunet excelente. Aici, în lumea Macintosh, a apărut o nouă tendință, cea a "înțoarcerii la simplitate".

Exemplificăm această tendință prin producția multi-media "Ludwig van Beethoven, simfonia a 9-a". Pentru a ne familiariza cu viața și opera lui Beethoven pe această cale, nu avem nevoie decît de un Macintosh și o unitate CD-ROM (pentru memorarea datelor, textelor și a graficii, cu o capacitate de aproximativ 650 MByte). În ceea ce privește calculatorul, ne ajunge chiar și modelul cc! mai "mic", noul Macintosh Classic, sau referitor la tipurile mai vechi, Macintosh SE. Mac Classic, prezentat în octombrie, costă aproximativ 3000 mărci.

Unități CD-ROM se pot obține pentru circa 1500 mărci. Ele au avantajul că de pe ele, cuplate cu un Macintosh, se pot prelucra atît discuri compacte CD-ROM cu date, texte, imagini și grafică, și de pe aceste unități se pot asculta și discurile audio compacte uzuale după dorință: prin căști sau prin instalația stereo.

Referitor la simfonia a 9-a a lui

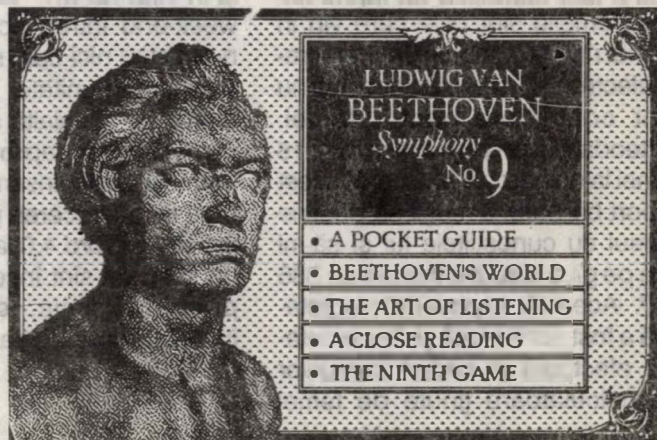
Beethoven, se poate cumpăra un disc compact cu simfonia a 9-a cît și două dischete cu baza de date propriu-zisă, la un preț de circa 400 mărci. CD-ul este o înregistrare comercială a simfoniei a 9-a. Cine se satură de învățat poate să se delecteze ascultînd muzică.

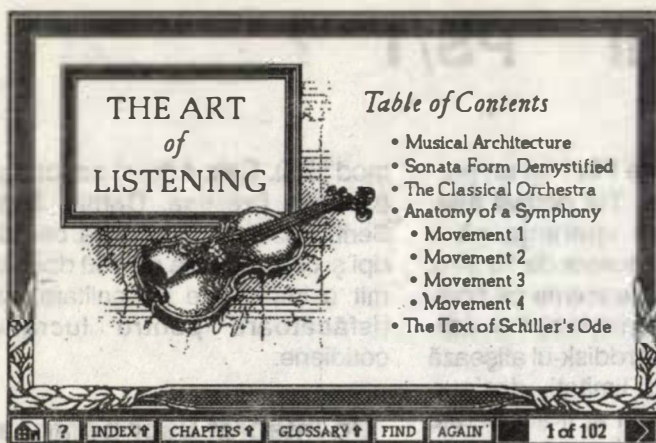
Baza de date se bazează pe programul multi-media "Hypercard", ea conține texte voluminoase, grafice și imagini ilustrate. Prin simbolurile de pe ecran se poate accesa discul compact din unitate. Astfel se pot derula anumite pasaje din simfonia a 9-a, bineînțeles stereo și de calitate Hi-Fi. Baza de date este împărțită în 5 părți, legate între ele. Meniul scurt permite accesarea directă și ascultarea diferitelor pasaje din simfonie.

Pe 124 de pagini ilustrate capitolul "Beethoven's World" ne poartă în lumea marelui maestru. Obținem informații atît din punct de vedere istoric cît și despre problemele și tragediile lui personale. Se subliniază atît influența lui Schiller asupra lucrărilor lui Beethoven cît și importanța celor doi în spiritul acelor timpuri. Prin apăsarea unei taste putem răsfoi însemnări interesante din jurnalul lui zilnic sau să ascultăm secvențe din schițe muzicale ale lui Beethoven. Practica concertistică de atunci și de acum sînt comparate în detaliu și ni se înșiră bibliografia pentru aprofundarea materiei.

Introducerea în "Arta auzului" ex-

Meniul principal al programului





O introducere
informativă în
teoria muzicii
clasice

plică modul în care este construită o simfonie sau ce instrumente compun o orchestră și care sînt caracteristicile lor. Textele sînt relaxate prin scanarea în cadrul textului a unor imagini contemporane, pasaje de muzică din CD și prin melodii produse direct de Macintosh. Plictiseala nu poate să apară.

Esența propriu-zisă o constituie partea "A close reading" (o citire atentă), care reprezintă o considerație aproape științifică muzicală asupra simfoniei a 9-a. Textele explică de ce Beethoven a compus respectivelor pasaje așa și nu altfel. În plus se explică și termeni muzicali de specialitate.

După dorință se pot vedea notele pasajelor alese care în timpul derulării respectivelor pasaje vor ieși în evidență. În această parte a sistemului utilizatorul are loc pentru noțiile proprii. Acest produs se

dovedește a fi o formă cu adevărat nouă a transmiterii cunoștințelor, printr-un fel de "control al scopului învățării", ce se face foarte lejer sub forma unui joc întrebare-răspuns: un fel de "cine știe, câștigă". Este vorba de a răspunde la întrebări de dificultate diferită din 3 categorii de probleme (exemple de ascultare, instrumentație, viața și opera). Chiar maestrul Beethoven în persoană apare ca prezentator, cicălește, laudă și face cu ochiul, în funcție de necesitate.

Modul acesta de îndeletnicire cu simfonia a 9-a a lui Beethoven nu este interesant doar pentru muzicologi sau pentru cei ce doresc să devină muzicologi. Acest produs multi-media este recomandat chiar și persoanelor care nu-și doresc să strălucească neapărat în concursuri "cine știe, câștigă". Deoarece face o imensă plăcere să înveți să înțelegi un artist prin joc, el este

cadoul ideal pentru posesorii de Mac iubitori de muzică.

Această parcurgere Beethoven este prima dintr-o serie întregă, care vor apărea în curînd în comerț, produse de firma americană Warner New Media. Produse asemănătoare despre Flautul fermecat al lui Mozart și Igor Stravinsky sînt deja terminate. Singurul handicap al acestor pachete informaționale muzicale: textele sînt disponibile, din păcate încă, doar în limba engleză.

Cu aceste produse noi este posibil un salt de calitate aproape fascinant al aplicațiilor pe calculator, atît în școli, universități, biblioteci, cît și acasă. Concluzia: jocurile pe calculator și televizorul sînt depășite, multi-media este în vogă.

(COMPUTER LIVE 12/90,
Stefan Horst)

Destinse fără frică și vesele

(continuare din pag. 7)

Cursurile se bazează, conform programei școlii de calculatoare pentru femei, pe deviza: "nu mai mult decît trebuie, nu mai puțin decît este necesar!" Cursantele confirmă acest motto: "cred că știu încă tot ce am învățat în acest week-end, fără ca seara să mai trebuiască să studiez". Pentru a con-

tracara ideea preconcepută despre femeile de la bucătărie, încă din anul '80 guvernul a sprijinit proiecte model, în care femeile vor fi instruite de femei. "Dacă la început lucrul pe calculator a fost impregnat de teamă, lipsă de independență, și nesiguranță, cu timpul mînuirea calculatorului a devenit mai sigură, mai independentă și mai evidentă, cum nici o femeie nu și-ar fi putut închipui la început", se spune în raportul

proiectului "Deutscher Frauenring e.V.". Aceste rezultate ale cercetării se suprapun experienței din practică pe care doctele cursurilor pentru femei de la Universitatea din München le-au putut face. La femei întrebarea "ce face calculatorul cu mine, cu gîndurile mele?" iese mai tare în evidență decît la bărbați.

De aceea cursurile pentru femei oferă și timp pentru discuții, de exemplu,

despre problema "prelucrarea electronică feminină a datelor, există așa ceva?". La etajul clădirii vechi mai există încă două încăperi. Atunci cînd ele vor fi reamenajate și aici se vor ține cursuri. O practicantă de medicină face cursuri de masaj - o apropiere feminină de tehnologia calculatoarelor este mai mult decît o simplă discuție profesională despre hardware.

Evenimentul PS/1 ?

Un microcalculator puternic la îndemâna tuturor. O victorie a la Pyrrhus pentru Apple, care-și vede strategia convivialității preluată de IBM. Și totuși, ce aduce nou PS/1? "Lucrări" pe un sistem personal.

Era utilizatorilor programatori s-a încheiat odată cu apariția primului Macintosh. Conceptul de calculator pentru un utilizator neavizat a penetrat greu în lumea PC-urilor, dar s-a impus chiar și în fața lui "Big Blue" care se vede nevoit să-și abandoneze politica de subordonare a utilizatorului față de specialist. Interfețele utilizator iconografice cu suport mouse se dezvoltă în ciuda coexistenței cu un sistem de operare în mod linie de comandă - MS-DOS - și a varietății de standarde grafice hardware și software. Și iată că, mai bine mai târziu decât niciodată, IBM propune un concept care nu mai e original, dar rămâne foarte actual: un sistem personal accesibil oricui.

PS/1 - o nouă generație ? Este noul venit atât de diferit pentru a impune un nou standard? Rămâne de văzut. Din punct de vedere al performanțelor nu este altceva decât un AT ca toate celelalte. Procesorul 80286/10MHz îl prezintă ca pe o mașină modestă, fără nimic extravagant. Dincolo de aparențele fișei tehnice vom (re)descoperi calitățile unui calculator compact: dimensiuni foarte reduse, grad mare de integrare, opțiuni ușor instalabile și nu în ultimul rând un preț accesibil la o calitate IBM. Iar pentru ca similitudinea cu Macintosh să fie completă, interfața seacă a MS-DOS este rapid înlocuită după boot-are cu imaginea mai prietenoasă a integratului MS Works. Utilizatorul neavizat poate demara rapid cu ajutorul mouse-ului un ciclu de autoinstruire sau chiar o aplicație.

Calitatea înainte de toate - constructorul nu a cedat tentației care animă majoritatea concurenților. IBM afirmă că nimic nu l-ar fi împie-

dicat să echipeze PS/1 cu un procesor mai rapid. Tot potrivit IBM-ului, este vană speranța că o mașină cu un procesor de 16 MHz ar fi mai rapidă de vreme ce componentele de memorie sînt de calitate medie, iar harddisk-ul afișează timpi de acces limitați; desigur, anumite procesări ar fi accelerate dar în utilizare prelungită și în condiții normale de lucru dezechilibrul sistemului determină o uzură precoce iar riscurile de pană cresc. Oricum, decizia unui ansamblu echilibrat de componente este semnificativă pentru calitatea materialului. Și altceva? Salutară este prezența adaptorului VGA integrat pe motherboard. Lizibilitatea informației pe ecran este mult ameliorată astfel, fapt perfect normal pentru un calculator care se doarește convivial. Și pentru că IBM nu face compromisuri, clientul are de ales doar între un monitor color și unul monocrom, standardul grafic rămînînd același. Harddisk-ul de 30 Mbytes și timp de acces mediu de 21 ms se pretează la numeroase aplicații exigente. Și aici, clientul cu posibilități modeste sau ale cărui necesități nu depășesc cadrul lui Works, poate opta pentru o a doua unitate de 3,5"/1,44 Mbytes în locul harddisk-ului. Mouse-ul, compatibil Microsoft, este sobru dar agreabil - de fapt este același care echipează familia PS/2; cablul foarte lung permite și de la distanță o utilizare lejeră a sistemului. O mențiune specială pentru livrarea în configurația standard a modelului V23 "Light Modem" al societății Com 1. De dimensiunile unui pachet de țigări, se cuplează la ieșirea serială și la rețeaua telefonică. Modem-ul este livrat în corespondență cu programul de comunicație Overcom al societății Goto Informatique.

Concomitent cu PS/1, IBM prezintă imprimanta matricială 2205 model 1. De tip 24 ace, bidirecțional, format A4, viteza ei de imprimare este de 192 cps în mod draft și 53 în

mod NLQ. Cele 4 tipuri de font-uri (Courier, Prestige, Gothic, Sans Serif), rezoluția orizontală de 360 dpi și cea verticală de 180 dpi permit o imprimare de calitate, satisfăcătoare pentru lucrările cotidiene.

Un integrat ușor pentru a porni: Works 2

Un integrat este un remarcabil instrument de inițiere în microinformatică, iar Works își demonstrează din plin această calitate. Odată cu procesorul de texte se învață mai bine de 50% din tehnica ... lucrărilor. Structura meniurilor și opțiunile odată asimilate, nu mai rămîne de înțeles decât finalitatea diferitelor module (calcul tabelar, gestiunea bazelor de date, etc.) Remarcabilă este și facilitatea de schimb de date între diferite module, chiar dacă nu de o manieră DDE (Dynamic Data Exchange); spre exemplu un raport conceput cu procesorul de texte care include și o parte dintr-un tabel realizat sub un program de calcul tabelar va fi actualizat ori de cîte ori tabelul este modificat dacă în prealabil se definesc legături DDE.

Tehnica Windows: opt ferestre conținînd fiecare un fișier pot fi afișate simultan pe ecran. Works integrează de asemenea și o reprezentare iconografică, iar utilizatorul are de ales între două moduri de lucru: text (25 sau 50 de linii) și grafic (60 de linii în mod VGA). Este posibilă astfel opțiunea între viteză și convivialitate, prima lăsînd deseori de dorit sub Windows.

Trusa de scule: nici un accesoriu de birou nu a fost neglijat: calculator, ceas și calendar sînt disponibile în orice modul. Funcțiile DOS sînt binevenite pentru orice neavizat care ignoră sintaxa comenzilor corespunzătoare. Remarcabilă rămîne posibilitatea "point and shoot" de a lansa orice alt program de sub Works.

Un procesor de texte dichisit: fără a aspira la puterea lui Word, procesorul de texte este bine conceput și remarcabil prin simplitate. Dacă este de neimaginat compunerea unei lucrări de 500 de pagini sub Works, el se pretează totuși foarte bine la redactarea unei corespondențe cotidiene. În afara funcțiilor uzuale de editare, un foarte complet dicționar precum și dicționarul de sinonime sînt plăcute surprize (evident, pentru utilizatori anglofoni, francofoni,...). Din păcate editarea nu se face în mod WYSIWYG (what you see is what you get) (care "garantează" că ceea ce va ieși la imprimantă este identic cu ceea ce se vede pe ecran), hap greu de înghițit în mod grafic și mal ales VGA. Pentru a compensa lipsa, Works propune o previzualizare înaintea imprimării, cam strîmtă și greu lizibilă în absența unei funcții de zoom.

Tabele și grafice: capacitatea modului de calcul tabelar este de 4096 linii pe 256 coloane. Deși selectarea celulelor cu mouse-ul este foarte facilă, autorii, precauți, au prevăzut și un sistem de numerotare stil Multiplan. Departe de posibilitățile unor Excel sau Quattro Pro, Works propune numeroase funcții matematice, statistice sau financiare, acceptabile pentru o utilizare curentă: urmăriri de bugete, calcule previzionale. Fără a pretinde puneri sofisticate în pagină, s-a dovedit imposibilă selectarea unui anumit tip de caracter, cu toate că și această opțiune figurează în meniu. Pentru a reuși aceasta, trebuie exportat tabelul în procesorul de texte, operațiune relativ simplă de altfel. Menținîndu-se în limitele 2D, modulul grafic are variate opțiuni de reprezentare: histogramme, curbe, grafice sectoriale, densități de puncte, etc. Modificarea legendelor, titlurilor permit personalizarea graficelor, încercarea de a manipula însă graficele se dovedește o întreprindere temerară, în contradicție cu pretențiile de soft didactic. În sfîrșit Works nu permite imprimarea unui tabel și a unui grafic pe aceeași pagină.

Gestiune de baze de date: în intenția autorilor acest modul își propune familiarizare începătorului cu noțiunile de înregistrare, cîmp, raport, bază de date. Departele însă de a răspunde rigorii conceptuale în proiectarea bazelor de date, Works 2 rămîne cu o singură pretenție: didacticismul. Fără a intra în detalii, o bază de date este în mod obligatoriu multifîșier în timp ce Works este monofîșier. Ar fi fost de dorit ca Microsoft să fie mai reținut în publicitatea făcută acestui modul simplisim și net depășit de managerul de baze de date al unor integrate de același calibru precum First Choice și Open Access Junior.

Comunicații: expedierea sau recepția de fișiere la distanță - acesta este rolul acestui modul; la aceasta se adaugă și posibilitatea de a emula un terminal. Posesorii de PS/1 au la îndemînă un obiect dedicat: Overcom.

Works rămîne un soft omogen și face un bun mariaj cu PS/1. Fără funcționalități excepționale dar fără defecte insuportabile - iată caracterizarea acestui integrat de generație nouă. Fără a privilegia un modul în maniera integratelor grele - Open Access sau Symphony - Works prezintă un excelent echilibru între diversele componente - un atu important pentru vocația sa inițiativă.

Overcom sau cum să scapi de singurătate

Livrarea lui PS/1 în Europa a început prin cea mai puternică reprezentanță "Big Blue": IBM France. Pentru a se integra specificului local - rețeaua de telecomunicații Minitel - IBM a optat pentru produsul societății Goto Informatique: Overcom. Prezentarea unui emulator complet al modului de lucru Minitel nu se face în defavoarea caracteristicilor de bază ale unui program de comunicații care se respectă. Overcom nu se adresează exclusiv francezilor. Modul memorare permite 'învățarea' tuturor

acțiunilor utilizatorului. În timpul accesului la un server o anumită secvență de răspunsuri poate fi "învățată" de Overcom, facilitate deosebită în consultarea unui cont bancar, căutarea unui orar de curse aeriene sau accesarea unor baze de date profesionale. "Internaționalismul" transpare și prin recunoașterea celor mai uzuale protocoale de comunicație, între care X/Y Modem, Kermit și evident Goto. Descoperirea unui editor de texte și a unui limbaj de programare indică existența unui nivel superior de utilizare a modului meniu - care răspunde problemelor cotidiene dar nu corespunde exigențelor unui transfer complex și important de date, cu conexiuni la anumite ore și însoțit de prelucrări. Interpretorul de comenzi permite atît lucrul în mod linie cît și execuția de fișiere de comenzi. Remarcabilă este posibilitatea de a pune la punct o aplicație sub controlul unui depanator (debugger) intern care permite urmărirea execuției fiecărei comenzi și localizarea sub editorul de text a liniei incorecte. **Limbajul de programare** conține peste 100 de comenzi, grupate în opt clase: comenzi de gestiune a comunicațiilor, a funcțiilor video, de tastatură, de acces la disc, comenzi de supraveghere a liniei și tratare a șirurilor de caractere în paralel cu procesarea comenzilor directe, comenzi de configurare Overcom, funcții videotex (specifice terminalului Minitel) precum și instrucțiuni condiționale. O mențiune aparte pentru existența unor tipuri variate de variabile și a posibilității de grupare a comenzilor în proceduri.

Overcom se conturează ca un program profesional în contextul exigențelor de convivialitate al "filozofiei PS/1". Cu toate acestea el rămîne foarte abordabil în modul de lucru cu meniuri, iar realizatorii lui fac eforturi spre a-l impune și neprofesioniștilor anunțînd de curînd versiunea operabilă cu ajutorul mouse-ului.

(ing. Alin Flaidăr)

Unix și virușii

Sistemele multiuser, respectiv sistemele de calcul înglobate în rețele, sînt foarte expuse înspre afară. În mod deosebit calculatoarele Unix, racordate în multiple feluri la rețeaua mondială, sînt în mod deosebit periclitare.

Recent, prin condamnarea studentului american Robert T. Morris, s-a încheiat un caz care a generat enorm de multe discuții relativ la securitatea datelor și în mod deosebit a sistemelor Unix.

Robert Morris este cel care a creat acel virus care în noiembrie 1988 a infectat rețeaua americană Internet. Această rețea acoperă întregul continent american și leagă practic toate sistemele Unix mari din S.U.A. Există și o legătură înspre rețelele militare. Deși virusul în mod evident nu a fost gândit să producă pagube în sistemele atacate, datorită blocajelor de sistem și utilizatorilor speriați care s-au deconectat de la rețea s-au produs pagube cifrate la cîteva milioane de dolari. Părea demonstrată în felul acesta sensibilitatea deosebită a sistemelor Unix la astfel de atacuri, prezisă de experți. Dar oare sînt sistemele Unix într-adevăr foarte sensibile?

Pentru a răspunde la această întrebare, merită făcută o inventariere a diferitelor surse de pericol care ar putea amenința securitatea calculatoarelor. Principial, tema are două aspecte: pe de o parte securitatea vizavi de pierderi de date prin erori de sistem, pe de altă parte protecția față de accesul neautorizat și distrugere. Ne vom ocupa în continuare articolului numai de cel de-al doilea aspect al problemei.

Un caz limită este distrugerea datelor de către utilizatori autorizați. Fie prin utilizarea defectuoasă a unui program, din neatenție sau pur și simplu din neștiință: probabil

că oricui i s-a întîmplat deja odată să distrugă date, fără să fi vrut. De așa ceva probabil că nu e sigur nici un sistem. Dar desigur că există caracteristici de sistem care fac mai mult sau mai puțin probabile astfel de incidente. Că cele mai multe comenzi Unix au destul de multă încredere în utilizator și execută și comenzi distructive fără să se mai asigure prin întrebări suplimentare poate fi cotate din acest punct de vedere ca destul de periculos. Și cuprinzătoarele posibilități disponibile prin Unix-Shell pot avea efecte neplăcute în caz de programare sau utilizare defectuoasă. Deși aici se pot aduce critici serioase Unix-ului, per total gradul său de securitate față de erorile utilizatorului este mai mare decît cel de sub MS-DOS. Distrugerea accidentală a datelor sistemului sau ale altor utilizatori este practic exclusă sub Unix, pe cînd un utilizator MS-DOS nepriceput, în ziua lui bună, poate face sistemul inutilizabil fără să se obosească prea mult.

În discuția generală asupra securității, pericolul reprezentat de utilizatorii autorizați este lăsat de-o parte. Punctul de interes al discuției este securitatea față de accesul neautorizat în sistem. Astfel de intruși pot fi atît persoane cît și programe. Urmările pot varia de la neplăceri mai mult sau mai puțin grave, ca în cazul virusului Internet, pînă la furtul sau distrugerea unor date cu daune în valoare de milioane. Pericolul unor intruși din afară, a așa-numiților "hackeri", este estimat în general a fi cel mai mare. Există o contramăsură aproape sigură și anume controlul absolut al accesului fizic la sistem. Dacă toate terminalele și calculatorul se află în încăperi la care accesul este complet controlat, nu poate lucra nimeni neautorizat la sistem. În practică însă, exceptînd poate domeniul militar, astfel de soluții nu pot fi luate în considerare. Chiar și

în întreprinderi care își pot permite controlul necesar, protecția obținută nu este fără fisuri.

La considerarea accesului la echipamentele de calcul, nu trebuie uitat subiectul rețea. Interconectările între diverse sisteme cresc mereu pe plan mondial, iar la sistemele Unix racordarea la rețeaua mondială este o chestiune aproape subînțeleasă. Posibilitățile pentru o imixtiune din afară cresc desigur mult prin aceasta. Că își crează acces persoane, prin dialog la terminal, sau programe corespunzătoare, este lipsit de însemnătate în acest context.

Mai ales atunci cînd legătura cu rețeaua este intens folosită, există interes pentru un acces confortabil la rețea și la alte calculatoare. Astfel încît puțini vor fi utilizatorii care vor fi dispuși să introducă mai multe parole, numai pentru a putea comunica cu un alt calculator. Dacă însă acest lucru nu se întîmplă, există pericolul ca intrusul să se poată răspîndi în toată rețeaua, după ce a reușit să se infiltreze într-unul din sistemele conectate. Printre altele, răspîndirea rapidă a virusului Internet a fost posibilă și pentru că între multe sisteme din rețea există o 'înțelegere' relativă la marcate "de încredere", la care nu se mai face și a doua verificare a parolei. Din nefericire și utilizarea rețelei pentru schimbul de știri via poștă electronică este un punct slab al securității datelor din acest punct de vedere. Prin exploatarea unor greșeli în programele respective, virusul Internet s-a putut multiplica folosind chiar acest mecanism.

Ce este pentru Unix rețeaua, este pentru multe alte sisteme rețeaua telefonică. Cel tîrziu de la filmul "War Games" (jocuri de război) fiecărui utilizator ar trebui să-i fie clare pericolele ce pîndesc un sistem în care se poate intra via

telefon. Cel puțin în ce privește acest aspect, filmul este foarte aproape de realitate. Pentru astfel de instalații, protecția simplă a marcajelor prin parole nu ajunge în toate cazurile. În astfel de situații se obișnuiesc măsuri de protecție mult mai cuprinzătoare. Astfel de exemplu este uzual ca după mai multe încercări nereușite de Login, să se întrerupă legătura. Dar hackerii perseverenți nu se vor lăsa descurajați de această măsură. O înregistrare a unor astfel de încercări ar trebui să fie regula, ca de altfel și controlul activităților desfășurate prin legătură externă, mai ales noaptea. Cea mai simplă și mai bună metodă este deseori deconectarea telefonului.

Între "atacurile" efectuate de persoane și cele efectuate de programe există deosebiri principiale. Programele sînt numite de regulă "virusi". Există o varietate de sub-specii, printre ele "ciii troieni", care spre exterior se prezintă ca utilitare folositoare. Apoi "bombele cu ceas" care "explodează" abia la un moment bine determinat.

Spre deosebire de modelul din antichitate, ciii troieni pe care îi discutăm aici necesită în cele mai multe cazuri existența unei posibilități de acces la sistem. Scopul artificului este de a obține drepturi mai mari în sistem. Sub Unix, ciii troieni pot profita de faptul că în anumite condiții programele pot avea alte drepturi decît utilizatorii care le-au lansat. Pentru programe sistem "adevărate", această proprietate nu numai că are sens, dar nu poate fi evitată. Un exemplu bun este programul pentru modificarea parolei. Sigur că utilizatorul obișnuit nu are drept de scriere în fișierul de parole. Dar pentru a-și putea modifica propria parolă, programul corespunzător trebuie să aibă acest drept. Aceasta necesită desigur o programare care să garanteze că în felul acesta utilizatorul își modifică numai propria parolă. Un factor de risc devine această proprietate atunci cînd așazisul program de sistem este în

realitate un virus, care în afara funcției propriu-zise mai urmărește scopuri ascunse, rareori curate. Dacă unui astfel de program i-a reușit odată să-și creeze drepturi sporite în sistem, soarta acestuia este cel mai adesea pecetluită. Un exemplu deosebit de "mîrșav" pentru specia cailor troieni sînt falsele Login-Shell-uri. Ele se substituie Login-ului din sistem și se prezintă utilizatorului ca de obicei, determinîndu-l să-și dea parola și marcajul. Dacă utilizatorul astfel indus în eroare își dă datele, nu i se face încă intrarea în sistem ci e păcălit în continuare printr-un mesaj de eroare. Login-ul "fals" ține minte parola și iese din scenă, lăsînd Login-ul adevărat să-și facă treaba, astfel că cel mai ades următoarea încercare de Login reușește fără probleme. Cum oricui i se mai întîmplă să tasteze greșit vreun caracter din parolă, procedura nu trezește nici un fel de suspiciuni. Cu puțin noroc, în felul acesta intrusul "pescuiește" chiar o parolă a unui utilizator privilegiat, astfel încît pagubele ce se pot produce sistemului devin incontrollabile.

Foarte asemănător funcționează "bombelă cu ceas". Și acestea se vor ascunde de obicei sub o mască inofensivă, fără a-și arăta prin ceva adevărata natură. La un moment dat aceasta se dezvăluie, cel mai ades provocînd distrugerile maximal posibile. Nu rareori aceste mecanisme sînt moștenite de la foști colaboratori, care în felul acesta vor să se răzbune pentru o prezumtivă sau reală nedreptate care li s-a făcut. Din punct de vedere al sistemului astfel de atacuri sînt aproape imposibil de prevenit, mai ales pentru că e vorba de persoane care pe baza funcției lor trebuie să aibă acces liber la sistem.

În lumea MS-DOS virusii de obicei sînt programe executabile sau secvențe de program care se răspîndesc îndeosebi prin dischete. Deoarece PC-urile sînt compatibile aproape fără excepție chiar la nivel de cod executabil, această metodă de răspîndire este optimă.

În lumea Unix acest gen de virusi sînt mult mai puțin răspîndiți. Hard-ul foarte diversificat ar face imposibilă răspîndirea largă a unui astfel de intrus. De aceea aici s-a creat un mecanism de reproducere diferit. Virusii din Unix își duc de obicei sursa cu ei. Dacă virusul a reușit să pătrundă într-un sistem, se reproduce acolo utilizînd uneltele disponibile pe acel sistem. În acest fel, el exploatează la modul optim interfața standardizată a Unix-ului și este la fel de independent de hardware ca oricare alt program aplicativ. Singura limită are se datorează diferitelor derivate ale Unix-ului. Astfel virusul Internet s-a putut replica numai pe calculatoarele cu Berkeley Unix, pentru că o parte din programele folosite nu există sub System V. După cum s-a văzut, aceasta nu l-a putut opri să blocheze întreaga rețea în numai cîteva ore.

Camuflajul virusilor

În ciuda unor diferențe semnificative, există o serie de caracteristici care se găsesc la toate programele-virus. Cea mai importantă este camuflarea. Un virus care se descoperă de la bun început nu prea are șanse de reușită. O metodă verificată de camuflaj este "ascunderea" banală a unui program. În MS-DOS aceasta se poate face de exemplu setînd atributul de 'hidden' (ascuns) la înscrierea în fișierul director. În Unix 'ascunderea' unui fișier se poate face în cele mai simple cazuri dîndu-i un nume care să înceapă cu punct. Ambele metode sînt primitive și expuse la descoperiri accidentale. Camuflarea stil 'cal troian' are mult mai multe șanse de reușită. Fie sub un nume fals de utilitar sistem, fie sub aspectul unui utilitar folositor și eficient, un astfel de virus își crează 'la vedere' accesul la sistem. În lumea PC-urilor, îndeosebi jocurile sînt posibili inamici mascați.

Pasageri clandestini prin sisteme

O altă caracteristică a mai tuturor virusurilor este tendința de răspîndire și reproducere. Rareori un astfel de intrus se mulțumește cu a fi infestat un singur sistem. Răspîndirea molimei se poate face prin cele mai diferite trucuri. O regulă de bază este de a nu te face repede remarcat. Un sistem recunoscut ca infestat micșorează drastic șansele virusului de a se mai răspîndi de pe el.

Atît pentru camuflare cît și pentru răspîndire este deseori folosită tehnica 'agățării' de alte programe, nevinovate, pentru a începe astfel călătoria spre alte sisteme ca un pasager clandestin. Varianta este destul de neplăcută, pentru că nici cunoașterea purtătorului original nu-i mai ajută utilizatorului.

Din fericire, nu în toate cazurile virusii au tendința să distrugă datele. În cazuri limită însă, sisteme complete devin nefolosibile, ceea ce denotă însă și insuficiente măsuri de protecție. S-au aflat ocazional cîteva erori din Unix care au făcut posibile astfel de incidente, însă aceste 'găuri' s-au peticit totdeauna repede după ce au ieșit la iveală. În antiteză, la un sistem lipsit de protecție cum este MS-DOS, chiar și distrugerea unor date de sistem e greu de împiedicat după ce un virus a pătruns.

Se pot găsi o multitudine de contramăsuri împotriva virusurilor în diversele lor variante. Au devenit populare așa-numitele 'Virus-checker'. Deseori acestea se bazează pe principiul că dimensiunea și eventualele caracteristici ale programelor ce trebuie protejate sînt cunoscute, astfel că orice modificare determină o alarmare. Asta funcționează cu succes la clasa de virusi 'agățatori' și desigur poate fi eficient numai dacă activarea programului s-a putut face cît timp sistemul era "sănătos". Alte programe de protecție cunosc "tiparul" mai multor virusi și-i caută pe

aceștia. Această metodă are sens la sistemele infestate. E nevoie însă de actualizări rapide ale programului de protecție în momentul apariției unui virus nou. Autorii virusurilor vor avea totdeauna un ușor avantaj din acest punct de vedere.

Așa se recunosc virusii

Și un control 'manual' poate fi de folos. Cîte un virus se deconspiră prin activități care pe un observator atent îl pot preveni din timp. Astfel de 'semne' sînt de exemplu timpi de încărcare neobișnuit de lungi sau, ceea ce mai ales la PC-uri se observă ușor, accese la disc la momente de timp neașteptate. Virusul Internet de exemplu s-a făcut remarcat prin fișiere inexplicabile ce apăreau într-un director prevăzut pentru fișiere temporare.

O anumită protecție o pot oferi și sisteme corespunzătoare orientate spre securizarea datelor. Atît penetrarea cît și răspîndirea sînt vizibil îngreunate. La o organizare modulară corespunzătoare a sistemului cresc șansele de a păstra eventualele pierderi în limite rezonabile.

Cel mai sigur mijloc de prevenire, ca la toate molipsirile, este 'igiena'. În cazul protecției datelor aceasta înseamnă mai ales: atenție la orice schimb de date. Cine nu are neapărată nevoie de instalații potențial riscante ca rețelele de calculatoare sau legăturile telefonice, face mai bine renunțînd la ele.

Dincolo de acestea există desigur și măsuri tehnice de protecție ca cele prevăzute în Orange Book. Desigur însă nici o încadrare superioară după 'cartea portocalie' nu este neapărat o garanție pentru un standard global de securitate ridicat. Și aici intră în joc factorul "utilizator". Căci și cele mai strălucite mecanisme de securitate nu sînt decît atît de bune cît o permit oamenii care lucrează cu ele. Pe de o parte, prin încadrarea într-una din clasele definite în Orange Book se fac niște aserțiuni principiale despre securitatea unui sistem. Pe de

altă parte, exact opusul sistemului absolut sigur este sistemul "prietenos cu utilizatorul", și aceste două obiective se exclud reciproc. Cele mai multe mecanisme ale clasificărilor superioare îngăduiesc de mult libertatea utilizatorilor, încît periclitează acceptarea acestor măsuri. De aceea încadrarea într-unul din nivelele superioare după aceste criterii nu este de dorit fără obiecții la toate sistemele.

Mecanismul de parolă din Unix se pretează, în principiu, pentru a împiedica accesul în sistem al unor intruși. Dacă însă o mare parte a utilizatorilor folosesc acest mecanism numai superficial sau de loc, atunci sporul de securitate este foarte mic. Și acest lucru a fost demonstrat de virusul Internet. Virusul conținea o variantă rapidă a algoritmului folosit pentru cifrarea parolelor utilizator depuse în fișierul "passwd". Prin utilizarea de dicționare pe respectivele calculatoare a pornit apoi la decriptarea cuvintelor conținute în acestea, cu speranța de a "nimeri" ceva din fișierul passwd. Această tentativă simplă a ajuns la rate de reușită de pînă la 50%. Înspre sistem, trebuie criticat faptul că fișierul dicționar putea fi accesat fără opreliști în citire. Faptul că toate parolele din el erau cifrate nu este o scuză, după cum s-a văzut, în primul rînd pentru că algoritmul de cifrare este cunoscut și e relativ simplu și ușor de programat. Dar un factor esențial sînt și utilizatorii. Practica de a folosi parole simple și uzuale reprezintă o slăbire suplimentară a mecanismului de protecție prin parole. Din comoditate, foarte des se folosesc coduri foarte scurte. Ceea ce face relativ simplă "spargerea" codului prin încercări. Pe lângă apelele către utilizatori, care în acest caz e greu de crezut că va fi un succes total, o seamă de măsuri pot fi luate la nivelul sistemului. Astfel de exemplu mecanismul de protecție prin parole poate fi determinat să nu accepte decît parole netriviale. O lungime minimă și folosirea de caractere speciale poate fi "forțată" în felul acesta. E drept că acest gen de măsuri nu au darul să stîrnea-

"Orange Book" - ce înseamnă "sigur"?

Ca multe alte "cuvinte mari", cuvîntul "siguranță" nu spune mare lucru prin el însuși. Fundamentele pentru posibilitatea măsurării și comparării securității sistemelor de calcul sînt create abia de o mulțime de criterii verificabile. Primele încercări în această direcție au venit, după cum era de așteptat, din domeniul militar. La solicitarea ministerului american al apărării s-a născut încă din 1983 un studiu devenit cunoscut sub numele de "cartea portocalie" datorită copertilor. Clasele și criteriile stabilite acolo sînt unanim recunoscute de atunci și nu servesc numai instituțiilor militare ca referință pentru achiziționare de sisteme noi.

Cartea portocalie împarte sistemele în 4 clase principale, notate cu "D" pentru cele foarte labile pînă la "A" pentru cele extrem de sigure. În cadrul acestor grupe există o altă subdivizare în trepte. Pentru fiecare grupă există un catalog de cerințe care trebuie îndeplinite pentru a permite încadrarea unui sistem în acea grupă. Fiecare treaptă conține desigur și toate cerințele claselor inferioare.

În clasa D sînt cuprinse toate sistemele care nu pot fi încadrate în nici una din treptele celelalte. Astfel de sisteme nu dispun practic de nici un fel de mecanisme de securizare. În această categorie intră multe sisteme de operare de la calculatoare mari, dar și toate sistemele de operare larg răspîndite pentru PC-uri, cum ar fi MS-DOS.

Clasa C se divide în grupele C1 și C2. Pentru încadrarea în C1 o premisă este existența unui meca-

nism de protecție la acces. Orice utilizator are anumite drepturi de acces la date, care pot fi și transmise. E prevăzută și acordarea de drepturi unor grupe de utilizatori. Prin aceasta sînt create premisele de bază ca tuturor utilizatorilor să le fie accesibile datele de care au nevoie și nimic în plus. C2 cere o structurare mai fină. Trebuie să se poată stabili și la nivel de fișier care utilizatori și în ce mod anume au acces. Sînt necesare și mecanisme de protocol pentru supravegherea acțiunilor întreprinse de utilizatori.

Mecanismele de acces ale Unix-ului pledează pentru încadrarea în clasa C1. Unele probleme mici și care principial pot fi rezolvate ușor ar face însă discutabilă această încadrare dacă s-ar face o examinare formală. Astfel fișierul "passwd" în care sînt depuse parolele cifrate poate fi citit de toți utilizatorii. Acest lucru contravine cerințelor C1 și poate însemna un factor de risc real în practică.

Aceasta și încă cîteva probleme au fost eradicate de o serie întreagă de ofertanți de Unix. O încadrare în C2 nu poate fi realizată pentru că pentru diferențierea drepturilor de acces nu există decît trei grupe de utilizatori: proprietarul, grupa proprietarului și toți ceilalți. O acordare realmente individualizată de drepturi de acces este imposibilă. După cum o demonstrează însă diverse sisteme bazate pe Unix, acest obstacol poate fi trecut. În zona PC-urilor trebuie menționat aici noul SCO-Unix System V/386 Release 3.2, care satisface aceste cerințe, păstrînd compatibilitatea datelor către versiunile mai vechi.

Clasa B se divide în 3 subgrupe. B1 cere mai întîi un control obligatoriu al accesului. B2 necesită o arhitectură modulară a sistemului. Asupra cerințelor și mai speciale din B3 și A nu se va mai insista aici, deoarece cerințelor formulate de acestea nu li se poate da curs decît cu eforturi deosebite, ceea ce face ca ele să nu prea aibă importanță în practică. În prezent este cunoscut un singur sistem care a putut fi încadrat în grupa A.

Deși deja treapta B1 poate fi atinsă numai cu mari eforturi de Unix, există cîteva încercări în această direcție. Unii producători oferă și "seturi de completare" cu ajutorul cărora Unix-ul poate fi ridicat pînă la nivelul B1. În ultimul timp unii candidați Unix se prezintă la examenul B2; printre ele și noul System V.4 de la AT&T. O restructurare semnificativă a kernel-ului Unix este în orice caz o premisă.

Se pune și întrebarea dacă orice calculator trebuie să îndeplinească cerințe de securitate complexe. Pentru utilizatori care nu lucrează cu date deosebit de pretențioase, necesitatea unui standard de securitate ridicat este discutabilă. Apoi, toată afacerea mai este și o problemă de preț, căci securitatea ridicată se plătește cu un preț corespunzător.

Punctul de greutate pentru utilizatorul "obișnuit" îl vor reprezenta în viitor probabil treptele C2 și B1. Aici se realizează un compromis între siguranță și utilizabilitate în viața de zi cu zi. Pentru utilizatorul mai sensibil intră în discuție și B2, pe cînd B3 și A nu par potrivite decît în cazuri excepționale.

scă entuziasmul utilizatorilor, însă securitatea sistemului are de cîștigat în mod sigur. Împreună cu un mecanism de control care urmărește încercările nereușite de

log-are și întrerupe legătura, anunțînd totodată și administratorul sistemului, pot fi evitate măcar încercările mai simple de a pătrunde în sistem din afară.

(continuare în pag.28)

(PC+Technik 12/90,
Stefan Wende)

Imprimanta - sau fără hîrtie nu se poate

Vă prezentăm succint principalele tipuri de imprimante folosite pe lîngă calculatoarele personale, creionînd în fiecare caz avantajele și dezavantajele principale. Articolul se încheie cu un tabel de sinteză, menit a prezenta cele mai cunoscute imprimante din fiecare clasă.

Imprimanta cu ace

Majoritatea utilizatorilor lucrează cu imprimanta cu ace, deoarece aceasta reprezintă alegerea cea mai ieftină (între 400-2500 DM).

Cele mai uzuale sînt imprimantele cu 9 sau 24 ace, dar există și modele "exotice" cu 18 sau 48 de ace.

Principiul de funcționare: scrisul și grafica sînt realizate pe hîrtie, prin impactul produs de ace din metal prețios, acționate de electromagneți, peste o bandă tușată, asupra hîrtiei. Acele, în funcție de tipul imprimantei, sînt plasate pe una, două sau trei coloane.

La imprimantele cu ace se deosebesc două moduri de imprimare:

1) modul de listare rapidă (draft-modus). Caracterele și literele sînt compuse din puncte singulare. Acest mod de lucru este recomandat în cazul listărilor mai voluminoase și unde calitatea nu este importantă.

2) modul corespondență (Near Letter Quality-NLQ). Se realizează o supraimprimare de puncte alăturate, rezultînd o calitate a listării mai bună. Firește în acest caz viteza de tipărire este mai scăzută.

Singurul avantaj al acestor imprimante îl reprezintă costul redus.

Dezavantaje:

- sînt zgomotoase
- nu realizează o calitate de imprimare profesională

Imprimantele cu 9 ace se recomandă pentru:

- începători
- utilizatori care doresc numai imprimări rapide sau liste de date, tabele cu calitate redusă
- corespondență de medie calitate

Imprimantele cu 24 ace se recomandă pentru:

- imprimări rapide
- utilizatori care pentru bani puțini caută imprimante cu calitate ridicată de imprimare, dar nu neapărat profesionale.

Imprimanta cu jet de cerneală

Deși sînt cumpărate de către tot mai mulți utilizatori (avînd prețuri între 1200-4000 DM) au avut pe piață o nemeritată poziție de cenușareasă.

Imprimantele cu jet de cerneală produc caractere și imagini prin pulverizarea fină a unor picături de cerneală pe hîrtia de imprimat. Există modele cu 12, 24, 48 de duze (cu cît mai multe cu atît calitatea este mai bună). În varianta color, culorile sînt aduse pe hîrtie din 4 patroane (fiecare conținînd 12, 24 sau 48 de duze). Cîte un patron negru, albastru, galben și roșu se mișcă deasupra hîrtiei pe un cîrucior. Din cele 3 culori de bază imprimanta își mixează culorile necesare. În modul de lucru text putem lista cu patronul negru pînă la 900-1000 pagini, iar în modul de lucru grafic numărul de pagini variază între 100 și 700 de pagini în funcție de densitate.

Rezoluția unei imprimante cu jet de cerneală este de 200x200 d.p.i. (dots per inch - puncte/inch).

Avantajele imprimantei cu jet de cerneală:

- costă mai puțin decît imprimantele profesionale din domeniu (cele cu laser) și imprimă la fel de frumos și silențios ca și acestea.
- imprimarea uniformă de la primul și pînă la ultimul caracter spre deosebire de imprimantele cu ace la care uzura benzii tușate scade calitatea imprimării.

Dezavantaje:

- sînt mai scumpe decît cele cu ace
- unele tipuri necesită o hîrtie specială pentru a obține o calitate ridicată a imprimării
- nu se poate utiliza hîrtie în mai multe exemplare

Se recomandă pentru imprimări rapide de calitate medie.

Imprimanta termică

Ca și imprimantele cu ace sau cu jet de cerneală, imprimanta termică face parte din familia imprimantelor matriciale.

Principiul de funcționare se bazează pe topirea substanței de pe banda tușată pe hîrtie. Asemănător imprimantelor cu ace, caracterele sînt alcătuite din puncte singulare, care sînt depuse pe hîrtia de imprimat de pe banda de film prin influența temperaturii unei perii de imprimare asupra acesteia. Capul de imprimare trebuie schimbat periodic. (Din cauza frecvențelor variații de temperatură). Imaginea grafică a paginii este foarte bună, aceasta obținîndu-se la o viteză de imprimare medie.

Cheltuielile de imprimare însă sînt ridicate. Ultimele modele utilizînd un rînd de elemente termice

mai late, au o viteză de imprimare comparabilă cu cea a imprimantelor cu laser, avînd și o rezoluție bună: 300 dpi (puncte/inch).

Avantaje:

- foarte silențioasă
- calitate de imprimare foarte bună

Dezavantaje:

- preț de cost ridicat
- nu se poate utiliza hîrtie în mai multe exemplare

Imprimantele termice se recomandă utilizatorilor care doresc să obțină imagini grafice frumoase și cu contrast ridicat.

Imprimanta laser

Experții prevăd un viitor deosebit acestor tipuri de imprimante. Pe măsură ce scade pe piață numărul de imprimante cu ace vîndute, crește numărul celor cu laser. Dacă prețurile acestora vor scădea și în continuare (în prezent sînt între 3000-12000 DM) s-ar putea ca ele să scoată de pe piață chiar și imprimantele cu 24 de ace.

Performanțele imprimantelor laser sînt mult mai ridicate decît ale celor matriciale. Pe cînd la imprimantele matriciale imprimarea se face caracter după caracter, la cele cu laser înainte de imprimare toată pagina este 'construită' în memorie, ca o imagine grafică.

Imprimantele cu laser produc litere și grafică cu ajutorul unei raze de lumină. O rază laser sau mici diode luminescente încarcă electrostatic un tambur de imprimare, în funcție de caracterul de imprimat. Acest tambur se rotește într-o baie de pulbere de toner, care aderă în locurile unde tamburul este încărcat static. Trecînd hîrtia prin dreptul tamburului, tonerul este "ars" pe hîrtie la temperatură ridicată.

Avantaje:

- relativ silențioase
- calitate ridicată de imprimare
- sînt cele mai rapide imprimante

(pînă la 12 pagini/minut față de 1,5 pagini/minut la imprimantele cu 24 ace)

Dezavantaje:

- cost ridicat, la care se adaugă costurile pentru întreținere (tambur, toner)
- nu se poate utiliza hîrtia în mai multe exemplare

Se recomandă pentru:

- utilizatorii care doresc o imprimare rapidă și de calitate ridicată, profesională, pentru texte și grafică
- utilizatori care doresc să se orienteze spre Desktop Publishing

Consultînd cataloagele de imprimante se găsesc unele date ce trebuiesc tratate cu circumspecție; de exemplu: în viteza de imprimare nu este inclus timpul necesar pentru pregătirea paginii de listat (care poate fi 15-20 minute!).

La achiziționarea unei imprimante laser numai anumite teste speciale de imprimare ne pot lămurii asupra performanțelor reale ale acesteia. (De exemplu programul de culegere Ventura Publisher conține teste de imprimantă: forme diferite, caractere albe pe fond negru.) Deci cumpărarea unei imprimante cu laser trebuie făcută cu mare atenție și după o documentare și testare amănunțită.

Imprimanta laser cu Postscript

Sînt cele mai bune imprimante pe piața mondială în momentul de față (prețuri între 6000-12000 DM).

Procedeul de tipărire a caracterelor și a imaginilor este asemănător cu cel utilizat la imprimantele cu laser obișnuite și de aceea au aceleași avantaje:

- calitate de tipărire ridicată
- zgomot redus

Diferența constă în faptul că imprimanta este comandată de cal-

culator printr-un limbaj special, de imprimantă, Postscript.

La aceste tipuri de imprimante caracterele se formează numai în perioada de pregătire pentru listare, ca urmare se realizează o economie de memorie atît în calculator cît și în imprimantă. Mărimea caracterelor se poate alege opțional în timpul culegerii.

La imprimantele Postscript performanțele de imprimare sînt determinate de memoria operativă (RAM) pe care o au la dispoziție și de soft-ul rezident care execută comenzile Postscript. Pentru realizarea unei imagini grafice este nevoie de o memorie minimă de 1,5 MByte RAM, necesară atît la transferul caracterelor de la calculator cît și la construcția paginii.

Avantaje:

- utilizatorul poate alege din 35 de tipuri (standard) de caractere de diferite mărimi
- caracterele, desenele pot fi rotite
- se pot crea foarte ușor cercuri, pătrate și alte forme geometrice

Dezavantaj:

- imprimanta trebuie să convertească comanda Postscript în sute de mii de mici puncte negre pe hîrtie, ceea ce duce la micșorarea vitezei de transfer.

Deoarece inventatorii, (firma "Adobe" din S.U.A.), cer o taxă de licență pentru fiecare imprimantă Postscript, prețul acestora rămîne ridicat în comparație cu imprimantele laser obișnuite. Iar varianta color a unei imprimante laser cu Postscript poate atinge 20.000 DM.

Imprimanta laser cu Postscript este un instrument de lucru pentru profesioniști:

- culegători pentru tipografie
- graficieni
- topografi
- specialiști în marketing și în reclame..

7 sfaturi pentru o instalare corectă a imprimantei

1. Puneți imprimanta pe un suport stabil. Mișcarea rapidă a capului de scriere, mai ales la imprimantele cu ace, poate răsturna mesele ușoare.

2. Imprimanta să fie instalată într-o încăpere apărată de soare. Folosirea imprimantei aflată sub razele soarelui, un timp îndeiungat, poate duce la scăderea randamentului, cauzată de încălzirea excesivă a pieselor mecanice.

3. Imprimantele cu ace produc un zgomot mare la imprimare. Punând sub imprimantă un suport

din material sintetic spongios, de grosime de 2 cm., reduceți simțitor zgomotul. Dacă se listează des și în mod profesional se recomandă achiziționarea unei carcase pentru protecția împotriva zgomotului (preț: 300 DM).

4. Înainte de a conecta imprimanta la calculator deconectați-o de la rețea. Tensiunea de alimentare cît și electricitatea statică proprie pot distruge circuitele de interfață.

5. Dacă nu reușiți să listați trebuie verificat dacă conectoarele

cablului de date atît la calculator cît și la imprimantă sînt fixate bine cu șuruburi.

6. Hîrtia care urmează a fi utilizată la listare trebuie pusă pe masă și nu pe jos. Dacă imprimanta trebuie să tragă hîrtia în sus un metru atunci există posibilitatea ruperii sau tractării necorespunzătoare a hîrtiei.

7. Deoarece imprimantele cu laser emană ozon (nociv) în timpul funcționării, încăperile mici trebuie bine aerisite.

7 sfaturi pentru achiziționarea de imprimante.

1. Să știți exact la ce tipuri de listări veți utiliza imprimanta. În primul rînd puteți consulta vânzătorul, iar în al doilea rînd alți utilizatori cu experiență.

2. Trebuie să reflectați dacă doriți să cumpărați de la un furnizor autorizat sau de la un "discounter" (angrosist), deoarece există diferențe mari de prețuri. De aceea o discuție înainte de cumpărare și despre asistența tehnică este bine venită.

3. Consultînd prețurile de la mai mulți furnizori autorizați sau discounteri puteți realiza economii importante.

4. Nu întotdeauna modelele noi sînt mai bune, dar sigur că sînt mai scumpe. Deci, să nu vă fie teamă de modelele mai vechi. Astfel

se poate cumpăra ieftin o imprimantă foarte bună.

5. Consultați mai multe oferte de preț; astfel o să aveți o privire de ansamblu mai bună asupra pieții. Lăsați pe vânzător să vă demonstreze modul de funcționare al imprimantei. Important: Nu vă grăbiți ! Cumpărați în liniște !

6. Dacă ați cumpărat o imprimantă, verificați încă în magazin că nu-i lipsesc accesoriile.

7. Nu uitați să cumpărați cablu de conectare la calculator. Ar.esta de obicei nu este inclus în prețul imprimantei.

ing. Zoltan Egyed

Cele mai cunoscute imprimante

Producător Produs	Model	Tipuri caractere	Interfață		Format hirtie			Color	Compatibil			Preț DM
			p.	s.	A4	A3	cont		EPS	HP-L	IBM	
Brother												
HL8PS	L	11	da	da	da	nu	nu	nu	da	da	da	9700,-
HL8D	L	7, B	da	da	da	nu	nu	nu	da	da	da	1000,-
HL8e	L	7, 3xB	da	da	da	nu	nu	nu	da	da	da	6400,-
HL4	L	B(2x),L,G,BRO1/2	da	da	da	nu	nu	nu	da	da	da	3900,-
M3524L	24A	B,A,G,Q	da	da	da	da	da	nu	da	nu	da	4300,-
M2524L	24A	P	da	da	da	da	da	da	da	nu	da	3400,-
M1824L	24A	P,B,G,Q	da	da	da	nu	da	da	da	nu	da	1800,-
M1224L	24A	P	da	nu	da	nu	da	nu	da	nu	nu	1000,-
M2518	18A	P	da	nu	da	da	da	da	da	nu	da	2850,-
M1918	18A	P,Q	da	da	da	da	da	da	da	nu	da	1700,-
M1818	18A	P,Q	da	da	da	nu	da	da	da	nu	da	1500,-
M1209	9A	P,G,Q	da	da	da	nu	da	nu	da	nu	da	700,-
Citizen												
ProDot 9	9A	T,R,SS,	da	opț.	da	nu	da	opț.	da	nu	da	1500,-
ProDot 9x	9A	T,R,SS	da	opț.	da	da	da	opț.	da	nu	da	1900,-
120D+	9A	C,SS,R	da	opț.	da	nu	da	nu	da	nu	da	600,-
124D	24A	C,T,R	da	opț.	da	nu	da	nu	da	nu	da	800,-
Prolaser 12	L	C,L,R,T	da	da	da	nu	nu	nu	da	da	da	7000,-
C.Hoh												
C-240	9A	SS,R,S	da	da	da	nu	da	nu	da	nu	da	1700,-
CI-5000	18A	C,E	da	da	da	nu	da	opț.	da	nu	da	6850,-
C-610	24A	C10,12,14	da	da	da	nu	da	nu	da	nu	da	2250,-
C-645	24A	D,C,SS,R,G	da	da	da	da	da	nu	da	nu	da	3650,-
Epson												
LX-400	9A	R,D,SS	da	nu	da	da	da	nu	da	nu	nu	650,-
LX-850	9A	R,D,SS	da	nu	da	da	da	nu	da	nu	nu	800,-
FX-1050	9A	R,D,SS	da	nu	nu	da	da	nu	da	nu	nu	1850,-
LQ-400	24A	R,D	da	nu	da	nu	da	nu	da	nu	nu	900,-
LQ-860	24A	R,D,SS	da	da	da	nu	da	nu	da	nu	nu	2500,-
LQ-2550	24A	R,SS,P,C,OCR-A,B	da	da	nu	da	da	da	da	nu	nu	4500,-
SQ-850	24C	R,SS,D,P,COCR-A,B	da	da	da	nu	da	nu	da	nu	nu	2600,-
GQ-500	L	13 tipuri	da	da	da	nu	nu	nu	da	da	nu	3950,-
Kyocera												
F800	L	79 tipuri	da	da	da	nu	nu	nu	da	da	da	4700,-
F1000	L	*	da	da	da	nu	nu	nu	da	da	da	5950,-
F1800	L	*	da	da	da	nu	nu	nu	da	da	da	1050,-
F2000	L	35 tipuri	da	da	da	nu	nu	nu	nu	da	da	14800,-
Star												
LC-10	9A	SS,C,OR	da	opț.	da	nu	da	nu	da	nu	da	580,-
LC-10	9A	SS,S,OR	da	opț.	da	nu	da	da	da	nu	da	750,-
LC 24-10	24A	D,4LQ	da	opț.	da	nu	da	nu	da	nu	da	900,-
XB 24-10	24A	D,11LQ,	da	opț.	da	da	da	opț.	da	nu	da	2060,-
Laser Print 8DB	L	C,P,TR,LF	da	da	da	nu	nu	nu	da	da	da	7000,-
Laser Print 8DX	L	C,P,TR,LP	da	da	da	nu	nu	nu	da	da	da	8000,-
Siemens												
HighPrint												
PT 18	18A	D LQ	da	da	da	nu	da	nu	da	nu	da	2000,-
4400e	24A	E,Pi,LQ	da	da	da	da	da	da	da	nu	da	4200,-
PT 88S	C	6 tipuri	da	da	da	nu	da	nu	opț.	nu	opț.	2570,-
PT 10	L	6 tipuri	da	da	da	nu	nu	nu	nu	da	nu	6370,-
Oki												
ML 320 Elite	9A	C,SS	da	opț.	da	nu	da	nu	da	nu	da	1400,-
ML 32 Elite	9A	C,SS	da	opț.	da	da	da	nu	da	nu	da	1800,-
ML 380	24A	C,D,OR,H	da	opț.	da	nu	da	nu	da	nu	da	1109,-
OkiLaser	L	C,LP,H,TR	da	opț.	da	nu	nu	nu	nu	da	nu	3000,-

Legendă:

B = Brougham, G = Gothic, A = Anglia Q = Quatro, P = Prestige,
SS = Sans Serif, C = Courier, LP = Line Printer, D = Draft
TR = Times Roman, O = Orator, H = Helvette, E = Elite, P = Pica

Utilizarea rațională a memoriei calculatoarelor personale

În numărul 1/90 al revistei "if" am prezentat metode de extindere a capacității fizice a memoriei calculatoarelor personale, cât și posibilitățile de adresare ale microprocesoarelor 80286 și 80386 în "Real-mode" și "Protected-mode". Revenim cu un articol în care vom prezenta mai în detaliu felul în care putem utiliza diferitele categorii de memorie, insistând asupra memoriei extinse - EXTENDED MEMORY.

În general există trei categorii de memorie utilizate de calculatoarele personale compatibile IBM-PC:

- 1. Convențională
- 2. Expandată (EXPANDED MEMORY)
- 3. Extinsă (EXTENDED MEMORY)

1. Noțiunea de memorie CONVENȚIONALĂ este legată de sistemul de operare MS-DOS și se referă la memoria de lucru cuprinsă între 0 și 640 K, fiind utilizată pentru rularea programelor DOS și de aplicații. Restul de 384 KB memorie, pînă la 1024 KB (1MB) cît pot adresa microprocesoarele 8088, 8086 și 80286/80386 în "Real-mode", sînt rezervați pentru memoria de ecran a plăcilor grafice, sau memoria ROM (BIOS sau I/O). Pentru a putea trece peste limita celor 640 KB utilizabili sub MS-DOS există cele 2 metode prezentate în continuare.

2. Noțiunea de memorie EXPANDATĂ a rezultat din standardul introdus de firmele Lotus/Intel/Microsoft (LIM) și denumit EMS (EXPANDED MEMORY SPECIFICATION). Conform acestui standard se poate utiliza zona de memorie fizică cuprinsă între 640

KB - 1 MB, și peste 1 MB pînă la 32 MB, (la 80286/80386 este aplicabil numai în "Real-mode!") prin tehnica denumită "page switching"-comutare de pagină. Această tehnică divide segmente de 64 kbyte din memoria calculatorului în pagini de 16 kbyte, fiecărei pagini fiindu-i atribuită o adresă unică. O asemenea pagină poate fi "suprapusă" peste o zonă neutilizată din domeniul de 384 KB, apărînd astfel în domeniul memoriei adresabile de pînă la 1 MB. Existînd posibilitatea de comutare a acestor pagini, se poate utiliza practic întreg spațiul de memorie existent (pînă la max. 32 MB!). (pag 78 PC Magazin Nr.47/90) Această metodă necesită atît un hardware adecvat (plăci speciale de memorie expandată care conțin o logică de comandă specială, pentru a putea suprapune paginile de memorie în spațiul de adresare dorit și care să permită schimbarea efectivă a adreselor de memorie) cît și software realizat în acest scop (un driver de memorie, denumit Expanded Memory Manager - EMM). Versiunea actuală a acestor "administratori" de memorie expandată se numește LIM EMS 4.0. Acesta pune și la dispoziția programatorilor mai multe

funcțiuni, prin intermediul întreruperii 67 hex.

Memoria expandată poate fi și simulată, pe calculatoare care nu au memorie multă și pe care se dorește totuși rularea unor programe mai mari care nu pot opera fără memorie EMS. Pentru aceasta există emulatoare EMS speciale care utilizează ca memorie expandată memoria extinsă, sau hard disk-ul.

3. Noțiunea de memorie EXTINSĂ a apărut odată cu utilizarea microprocesoarelor 80286/80386, care au extins posibilitatea de adresare fizică a memoriei operative la 16 Mbyte și chiar 4 Gbyte. Acest lucru permite extinderea memoriei fizice prin instalarea de plăci de memorie suplimentare. Această extindere a memoriei a fost prevăzută pentru utilizarea unor sisteme de operare care să nu fie limitate de cei 640 Kbyte ai sistemului MS-DOS, cum sînt OS/2, XENIX și UNIX, care pot administra 16 Mbyte de memorie.

Deoarece MS-DOS folosește modul de lucru al 80286, 80386, 80486 care se comportă ca și

Fig. 1 - Rutinele întreruperii 15hex, Protected Mode

Funcția 87 hex: Transfer de cuvinte între memoria de lucru și memoria extinsă.

Intrare: AH = 87 hex
CX = Numărul cuvintelor de transferat
ES = Adresa de segment a GDT
SI = Adresa de offset a GDT
leșire: Bitul CF carry (0-fără eroare; 1-cu eroare)
AX = Cod de eroare (0 - nici o eroare)

Funcția 88 hex: Obținerea capacității memoriei extinse

Intrare: AH = 88 hex
leșire: AX = Capacitatea memoriei extinse (în kbyte)

Funcția 89 hex: Comutare în "mod protejat"

Intrare: AH = 89
leșire: nici una

8086, nu s-ar putea utiliza decât maxim 1 Mbyte al acestei memorii extinse.

Există totuși posibilitatea de a utiliza această memorie extinsă și sub MS-DOS. Spre deosebire de memoria expandată, accesul la memoria extinsă nu este normat. Totuși s-au cristalizat două metode de utilizare a acestei memorii: metoda "întreruperii 15 hex" și metoda Vdisk.

Cum se poate accesa memoria extinsă ?

În principiu, utilizând microprocesoarele 80286 și 80386 în "mod real" - cum o face MS-DOS - acestea nu sînt altceva decît niște 8086/8088 mai rapide, și nu pot adresa decît 1 Mbyte de memorie. Noutatea adusă de 80286/80386 în această privință, o reprezintă posibilitatea funcționării în "mod protejat". Acesta permite izolarea logică a domeniului de adresare pentru mai multe programe. Prin schimbarea rapidă între aceste programe este posibilă rularea în același timp (aparent) a mai multor programe. În "mod protejat" se pot utiliza 24 linii de adresă la 80286 și 32 linii la 80386, putînd adresa astfel 16 Mbyte respectiv 4 Gbyte de memorie. Pentru a putea beneficia de aceste facilități oferite de microprocesoare, MS-DOS trebuie să comute funcționarea în "mod protejat", pentru a accesa memoria extinsă dorită și apoi să revină în "mod real". Acest lucru este posibil prin controlul unui bit de stare (flag) al CPU, dar utilizarea acestui procedeu este mai complicată decît pare la prima vedere. Cea mai simplă sarcină este de a comuta

procesorul în "mod protejat". Pentru aceasta există instrucțiunea în cod mașina >lmsw<, dar mai trebuie avute în vedere și anumite registre și structurile de date, care în "mod real" nu au nici o utilizare, dar devin foarte importante în "mod protejat". Urmează apoi efectuarea accesului propriu-zis la memoria extinsă, iar ultimul pas este și cel mai greu: revenirea în "mod real", pentru a continua MS-DOS. Nu există nici o instrucțiune de revenire din "mod protejat" în "mod real" la 80286.

Abia la proiectarea lui 80386 s-a prevăzut acest lucru. Există totuși o metodă de revenire în "mod real" și la 80286: procesorul să primească un Reset, deoarece după un Reset procesorul se startează în "mod real". Dar după Reset calculatorul începe cu "self-test", ceea ce noi nu dorim în acest moment! Din fericire nu trebuie să vă îngrijiiți de chiar tot ceea ce ține de aceste lucruri, deoarece BIOS-ul unui AT conține cîteva funcții pentru accesul la memoria extinsă. Aceste funcții sînt disponibile prin intermediul întreruperii 15 hex, și permit determinarea mărimii memoriei extinse, trecerea în "mod protejat", accesarea efectivă a memoriei extinse. (fig. 1).

În principiu, prin intermediul întreruperii 15 hex se poate face tot ceea ce ține de "protected mode". De un interes deosebit sînt primele două funcții (87 hex și 88 hex) ale acestei întreruperi. Funcția 89 hex nu este interesantă în acest moment, deoarece utilizarea ei, fără a pregăti anterior cîteva registre importante, conduce la blocarea calculatorului.

Funcția 88 hex. Prin apelarea acestei funcții se determină capacitatea memoriei extinse (în Kbyte) și se returnează în registrul AX. În exemplele 1 și 2 (xramchk.pas și xramchk.c) se demonstrează cum puteți folosi această funcție în Turbo Pascal și C. (listing 1 și listing 2)

Funcția 87 hex. Cu ajutorul acestei funcții pot fi transferate blocuri de memorie din memoria convențională în memoria extinsă și invers. După cum se vede din fig. 1, este necesară adresa de segment și de offset a unei tabeli denumită GDT (Global Descriptor Table). Microprocesoarele 80286 și 80386 au fost proiectate să fie utilizate ca suport hardware pentru un sistem multitasking efectiv. Pe lîngă mecanismul protecției domeniului de adrese al unui program față de alții, ele prevăd și procedeu de a memora, la nevoie, programe întregi sau segmente de program sau date pe harddisk, atunci cînd memoria operativă trebuie utilizată în alt scop. Cînd sînt necesare din nou aceste părți de program sau date, ele sînt readuse în memoria operativă. Acest procedeu presupune existența unui mod de adresare a memoriei total diferit față de modul "real". Pentru a citi, de exemplu, în acest mod un byte din memorie se utilizează o adresă de segment și una de offset, care împreună formează adresa fizică a celei de memorie adresate.

În mod "protejat" un asemenea acces nu este posibil. Dacă un program ar avea aici posibilitatea să utilizeze o adresă fizică, ar putea să acceseze și adrese din domeniul altui program. De aceea se utilizează numai adrese logice, iar transformarea adreselor logice în adrese fizice este realizată de către o unitate de administrare a memoriei (MMU = Memory Management Unit) integrată hard în procesor. Astfel conținutul unui registru segment nu trebuie înțeles ca o adresă ci ca un "selector". Acesta este un indicator al unei tabeli în care se găsesc la rîndul lor "descriptori".

Deplasament	Conținut	Lungime
00 hex	rezervat	8 bytes
08 hex	Tabela de descriptori (GDT)	8 bytes
10 hex	Adresa sursă	8 bytes
18 hex	Adresa destinație	8 bytes
20 hex	Segment de cod BIOS	8 bytes
28 hex	Segment de stivă BIOS	8 bytes

Fig.2 -
Structura
GDT
(Global
Descriptor
Table)

Un descriptor descrie în final parametri fizici ai segmentului de memorie utilizat, adică: -adresa fizică, lungimea segmentului și drepturile de acces. Această tabelă, care conține descriptori, se numește GDT.

În concluzie: -Un program care se rulează în mod protejat utilizează "selectorii" pentru a accesa zona de memorie. -MMU determină, pe baza acestor selectori, descriptorii, numai după aceea procesorul putând accesa adresa fizică. În cazul în care un domeniu de memorie a fost transferat pe un suport extern, descriptorul conține un marcaj special, astfel încât MMU va încărca mai întâi acel domeniu de pe suportul extern în memoria internă. Cu ajutorul acestor selectori și descriptori procesorul poate supraveghea toate accesele la diverse segmente de memorie.

Pentru a lucra în mod protejat trebuie mai întâi să pregătim deci o tabelă globală de descriptori (GDT), care să permită accesul atât la memoria convențională cât și la cea extinsă. Adresa acestei tabele trebuie definită în combinația de registre ES : SI. Această tabelă GDT este valabilă pentru întreg sistemul de operare și nu numai pentru un anumit program. (în acest caz ar fi vorba despre o tabelă locală de descriptori). (fig.2)

Importanța câmpurilor "adresă sursă" și "adresă destinație." Aceste două câmpuri conțin descriptori, care trebuie să indice adresele fizice ale segmentelor de memorie utilizate ca sursă respectiv destinație a transferului.

Fiecare descriptor se compune din 8 bytes. (fig.3)

Câmpul "adresă fizică de segment" trebuie să conțină adresa fizică a segmentului de memorie de transferat (24 de biți = 3 bytes)

Câmpul "lungime segment" conține lungimea zonei de memorie de transferat. Dar pentru BIOS

Fig. 3 - Structura unui descriptor în cadrul GDT

Deplasament	Conținut	Lungime
00 hex	Lungimea segmentului	2 bytes
02 hex	Adresa fizică a segmentului	3 bytes
05 hex	Drepturi de acces	1 byte
06 hex	Rezervat	2 bytes

trebuie să avem numărul de cuvinte de transferat (este bine să punei valoarea FFFF hex). Câmpul "drepturi de acces" conține informații despre: felul accesului permis (R, W, R/W), nivelul de privilegiu al programului, și marcajul de "prezență în memoria internă". Este bine să prevedeați valoarea 97 hex pentru acest câmp.

Pentru a transfera cu ajutorul funcției 87 hex un bloc de memorie din memoria operativă în memoria extinsă, trebuie să pregătiți doi descriptori în GDT. Apoi BIOS-ul copiază octeții adresei de memorie indicate de descriptorul "adresă sursă" la adresa de memorie indicată de descriptorul "adresă destinație". Deci problema importantă la utilizarea funcției 87 hex constă în pregătirea unei structuri de date corecte în GDT. Am pregătit un exemplu (listing 3) "mutăram.asm" care arată cum pot fi transferați 1000 octeți (500 cuvinte) de la adresa de memorie operativă 3000:0000 hex la adresa 0000:0000 hex din memoria extinsă.

La executarea acestei rutine, microprocesorul comută în "mod protejat" și transferă numărul dat de cuvinte. Pentru a ajunge din nou în "real mode" se emite un semnal de Reset la procesorul 80286 de către funcția 87 hex. Intrând în self/test de inițializare procesorul apelează un byte din memoria operativă, care a fost pregătit mai înainte acolo în acest scop. Acesta semnalizează procesorului că nu este vorba despre un "Cold-Start" ci de un Reset din "mod protejat". Drept urmare se efectuează o ramificare în partea de program a funcției 87 hex care urmează transferului de octeți, astfel încât funcția poate fi încheiată corect. Această metodă a fost de-

numită de către Gordon Letwin, conducătorul proiectanților de sistem ai firmei Microsoft, în cartea sa "Inside OS/2", drept <gararea automobilului la viteza de 100 Km/h>.

Un mic dezavantaj al acestei metode pentru utilizatorul de MS-DOS: La fiecare apel al funcțiilor întreruperii 15 hex, care comută în "mod protejat" se pierd un număr de semnale de întrerupere. În general acest lucru nu este prea rău, dar poate conduce la anumite probleme în programele care utilizează întreruperea de timer 1C hex, pentru a fi apelate periodic. Ceasul intern al MS-DOS este un asemenea program. La fiecare comutare în "mod protejat" și înapoi se pierd câteva ns. Dacă se utilizează des funcțiile din "mod protejat", timpul pierdut se cumulează la câteva secunde. După cum arată listing-ul 3, dacă se cunosc formalitățile la apelarea funcției 87 hex a întreruperii 15 hex, nu mai reprezintă o problemă schimbul de date între memoria operativă convențională și memoria extinsă. Înainte însă de a începe să transferați Kbytes de date să vă mai puneți o problemă mare care apare odată cu accesul la memoria extinsă: BIOS-ul AT-ului și al 386-PC-ului nu poate face o distincție între diferitele programe care apelează memoria externă. Ce se întâmplă dacă mai multe programe utilizează în același timp memoria extinsă?

De exemplu un program ajutător rezident și o aplicație de calcul tabelar, depun date în același domeniu de memorie extinsă deoarece "nu știu unul de celălalt". Începând cu DOS versiunea 3.3 este și mai rău: memoria extinsă poate fi utilizată ca RAM-disk. Cu DOS versiunea 4.0 se pot depune în memoria extinsă buffere de date

Deplasament	Conținut în hexa
00 hex	59 22 29 00 00 00 00 00
08 hex	00 00 00 00 00 00 00 00
10 hex	00 00 00 00 00 00 00 18
18 hex	1c 21 3d 10 df 10 4f 01

Fig.4 Primii 32 de octeți ai întreruperii 19 hex, fără RAM-disk instalat.

sau fișiere. Dar cine garantează că acestea nu utilizează exact aceeași zonă pe care ați utilizat-o în programul d-voastră ? Prima cerință este deci de a împiedica utilizarea de către un program a unei zone din memoria extinsă utilizată deja într-o aplicație. Pentru a ajunge la o soluție, trebuie să vă puneți întrebarea de unde știe un program care zonă poate fi utilizată și care nu. Există soluții și pentru aceasta! Mai întâi programul determină mărimea memoriei extinse prin intermediul funcției 88 hex a întreruperii 15 hex. Memoria semnalată de această funcție este utilizată apoi total sau parțial.

Un program nu va utiliza deci o zonă de memorie care nu a fost confirmată de funcția 88 hex, deoarece presupune că nu există altă zonă de memorie extinsă în afara celei semnalate. Deci pentru a realiza protejarea unei zone de memorie față de accesul altui program trebuie să vă asigurați ca BIOS-ul să nu returneze întreaga mărime a memoriei extinse, ci numai a domeniului utilizabil liber. Pentru aceasta trebuie să deviați vectorul disponibil pentru întreruperea 15 hex (adică adresa rutinei de deservire a întreruperii) într-o rutină proprie.

Să presupunem că ați scris un program foarte avid de memorie, care necesită 640 Kb memorie extinsă. Dacă dispuneți de 1 Mb de asemenea memorie, funcția 88 hex vă returnează valoarea 1024 (Kb). Programul d-voastră trebuie să devieze întreruperea 15 hex într-o rutină proprie, care cuprinde funcția 88 hex și care semnalizează capacitatea memoriei extinse rămasă

disponibilă, adică 1024-640=384 Kb.

Un program următor care apelează funcția 88 hex va stabili o capacitate de 384 Kb a memoriei extinse și

nu va încerca să acceseze cei 640 Kb rezervați de d-voastră. La rândul lui acest program poate deruta vectorul întreruperii 15 hex într-o rutină proprie și să-și rezerve în același mod o zonă de memorie necesară, lăsând pentru programul următor și mai puțină zonă de memorie extinsă.

Totuși acest procedeu nu reprezintă o soluție universală. O problemă apare la încheierea programului, când ar trebui eliberat spațiul de memorie rezervat de acesta. Teoretic acest lucru se poate realiza prin redarea controlului vectorului întreruperii 15 hex către rutina inițială din BIOS. Ce se întâmplă însă dacă între timp alte programe au utilizat acest vector ? Rutinele de deservire a întreruperii acestor programe ar fi de asemenea evitate și este relansată problema inițială: un program nou încărcat va porni de la situația că dispune de întreaga memorie extinsă.

Mai există și a doua problemă. Ce se întâmplă cu rutinele ajutoare ale MS-DOS-ului, care utilizează memoria extinsă?. Un exemplu îl reprezintă driverul de RAM-disk <vdisk.sys> sau <ramdrive.sys>, care utilizează memoria extinsă. Pentru a utiliza deci memoria extinsă trebuie mai

întâi să verificați că nu a fost instalat un RAM-disk. Dacă există, atunci trebuie să determinați capacitatea acestuia, iar spațiul de memorie rămas puteți să-l utilizați d-voastră.

Pentru a determina acest lucru trebuie să știți

Cum funcționează un RAM-DISK

Acesta se bazează pe derutarea vectorului întreruperii 19 hex într-o rutină proprie. În mod normal această întrerupere este utilizată la procedura de bootstrapare a sistemului. Noua rutină a driverului de RAM-disk nu face în principiu altceva decât să redea controlul la rutina de deservire a "vechii" întreruperii 19 hex.

Ca orice vector de întrerupere și noul vector al întreruperii 19 hex se compune din: adresă de segment și de offset.

Adresa de segment indică începutul segmentului de memorie pentru memoria extinsă de deservire a întreruperii iar adresa de offset este cea a primei instrucțiuni din această rutină. De cele mai multe ori adresa de offset este 0, deoarece începutul segmentului este în același timp și începutul rutinei. În cazul prezentat, startul rutinei de deservire a întreruperii nu se găsește totuși la începutul segmentului, unde se află câțiva octeți de recunoaștere care arată dacă este instalat un RAM-disk. În continuare urmează prima și în același timp ultima instrucțiune a noii rutine de deservire, care apelează din nou vechea rutină a întreruperii 19 hex. Efectul global al noii

Deplasament	Conținut în hexa	Conținut în ASCII
00 hex	00 00 e7 19 00 08 a9 00	-----
08 hex	d4 00 01 00 00 00 00 00	-----
10 hex	00 00 56 44 49 53 4b 20	-- VDISK
18 hex	20 56 33 2e 33 28 00 00	V3.3 ---

Fig.5 Primii 32 de octeți ai întreruperii 19 hex, cu RAM-disk instalat.

rutine, pentru întreruperea 19 hex, este acela că la începutul ei sînt depozitați cîțiva octeți de recunoaștere care indică instalarea unui RAM-disk. Fig.4 și fig.5 arată ca exemplu primii 32 bytes și întreruperea 19 hex.

Puteți realiza această situație introducînd în fișierul <config.sys> următoarele:
device = vdisk.sys 128/e.

După cum rezultă din fig.4 și 5 prezența unui RAM-disk este indicată de conținutul octeților 12 - 1C hex : VDISK V3.3.

Chiar dacă poziția unde începe acest text diferă de la o versiune la alta a MS-DOS, dacă el se găsește undeva în primii 32 bytes înseamnă că există un RAM-disk instalat. Nu este greu de imaginat o rutină care să determine prezența acestui text semnalizînd apoi prezența unui RAM-disk instalat.

Începutul acestui RAM-disk în memoria extinsă se poate găsi tot în primii 32 octeți ai rutinei. Astfel la adresele 2C, 2D, 2E hex găsiți valorile: 00, 00, 01, care reprezintă o adresă pe 24 de biți, în format Intel: 010000 hex. Aceasta este adresa de start a RAM-disk-ului, deci imediat după limita de 1 Mb. În primii octeți ai acestui RAM-disk găsiți blocul de bootstrapare a RAM-diskului (în fig.6, primii 32 bytes). În octeții 03 - 04 hex regăsiți textul <VDISK V3.3> care indică existența unui RAM-disk instalat.

În octeții 0B - 1D hex urmează blocul de parametrii BIOS, care conține informații despre mărirea sectoarelor, a cluster-elor, membrul de FAT etc. În fig.6 avem instalat un RAM-disk care lucrează cu 128 bytes/sector, utilizează un sector rezervat și 1 FAT, permite maximum 64 de intrări în directorul principal și are o capacitate globală de 1024 de sectoare = 128 Kb.

Octeții 1E și 1F conțin adresa primului octet, neocupat de RAM-

disk, din memoria extinsă (reprezentată în Kbytes). În fig.6 recunoașteți valorile de 04 hex și 80 hex, care dau adresa 0480 hex, ceea ce înseamnă că primul octet liber este la adresa 1024+128 Kbyte.

Cunoașterea acestei adrese vă oferă și posibilitatea să desemnați memoria extinsă ca ocupată, prin mărirea acestei valori.

Un program care urmează să folosească și el memoria extinsă "vede" această zonă de memorie "rezervată" de programul d-voastră tot ca o zonă ocupată de RAM-disk. În felul acesta puteți asigura protecția datelor d-voastră din memoria extinsă față de alte programe.

Ce aveți de făcut practic:

- căutați în primii 32 octeți ai rutinei de serviciu a întreruperii 19 hex șirul de caractere "VDISK", pentru a stabili dacă este instalat sau nu un RAM-disk.
- dacă da, verificați dacă acest RAM-disk începe în memoria extinsă.
- dacă da, citiți valorile de la adresele 1E hex și 1F hex, adunați la ele zona de memorie pe care doriți să o rezervați programului d-voastră și scrieți noua valoare în blocul de boot a RAM-disk-ului.
- în final, puteți folosi zona de memorie rezervată astfel pentru rutinele sau datele programului d-voastră.

În concluzie, putem spune că există două procedee de a accesa memoria extinsă:

Deplasament	Conținut în hexa	Conținut în ASCII
00 hex	00 00 00 56 44 49 55 4b	--- 'VDISK
08 hex	33 2e 33 80 00 01 01 00	3,3 -----
10 hex	01 40 00 00 02 fe 06 00	-----
18 hex	08 00 01 00 00 00 80 04	-----

Fig.6 Primii 32 de octeți ai RAM-disk-ului.

Primul, prin comutarea întreruperii 19 hex într-o rutină proprie, astfel încît programele care urmează să "vadă" că este mai puțină memorie liberă în memoria extinsă.

Al doilea, prin care se rezervă o zonă în memoria extinsă, modificînd blocul boot al RAM-disk-ului.

Observație: -deosebirea între cele două procedee este semnificativă: în primul caz se utilizează memoria extinsă de sus în jos, în al doilea caz de jos în sus.

Există și o cale de mijloc (cea mai bună, de altfel!): mai întii aflați prin intermediul întreruperii 15 hex mărirea memoriei extinse. Apoi verificați dacă este instalat un RAM-disk în memoria extinsă. În caz că da, aflați mărirea zonei de RAM-disk. Apoi trebuie să porniți de la faptul că cea mai mică din valorile astfel aflate reprezintă valoarea adevărată spațiului de memorie disponibil în memoria extinsă, și rămîne disponibil după RAM-disk.

Acest spațiu de memorie puteți să îl folosiți pentru programele d-voastră. Mai trebuie doar să vă asigurați că alte programe nu vor avea acces la acest spațiu de memorie. Pentru aceasta folosiți o nouă rutină pentru întreruperea 19 hex care să modifice valoarea mării spațiului de memorie extinsă cu valoarea dorită de d-voastră, atunci cînd este apelată funcția 88 hex.

De asemenea trebuie să modificați și valoarea mării "oficiale" a memoriei pentru RAM-disk, astfel încît zona ocupată de programele

d-voastră să pară ocupată de RAM-disk.

extinsă un număr de octeți care să simuleze blocul boot al unui RAM-disk instalat. În acest fel programele d-voastră sînt păstrate în

memoria extinsă cu toată siguranța posibilă.

Dacă nu este instalat un RAM-disk, atunci depuneți în memoria

(ing. Emil Palade, ing. Iosif Fettich)

```

program xramchk;

  (* -----
  determina marimea memorie extinse
  (eXtended memory) la un AT
  cu ajutorul functiei 88 hexa a INT 15 hexa.
  ----- *)

uses dos; (* biblioteca pentru accesul la UC *)

var
  cpu: registers;
      (* variabile pentru acces la BIOS *)
begin
  writeln;writeln;
  cpu.ah := $88; (* functia 88 hexa *)
  cpu.al := 0; (* AL sa fie nul *)
  intr($15, cpu); (* declanseaza intreruperea *)
  if (cpu.ax = 0)
  then begin
    writeln ('Calculatorul D-voastra fie este un XT,');
    writeln (' fie nu are EXTENDED MEMORY instalata. ');
    writeln;
  end
  else begin
    write ('Sistemul D-voastra poseda ',cpu.ax,
           'kbyte Extended Memory. ');
  end;
end.
  
```

Listing 1 - rutina extram în Pascal

```

/* -----
XRAMCHK.C

determina marimea memoriei extinse (eXtended memory)
la un AT prin intermediul functiei 88 hex a INT 15 hex
----- */

#include "dos.h"
integer extram()

/* -----
functia intoarce marimea memoriei extinse
----- */

{
  union REGS cpu;          /* variabile registre */
  cpu.h.ah = 0x88;         /* functia 88h */
  cpu.h.al = 0;            /* AL sa fie nul */
  int86 (0x15, &cpu, &cpu); /* apel de BIOS */
  return cpu.x.ax;        /* si retur */
}

main()
{
  if (extram() == 0)
  {
    printf ("\nCalculatorul D-voastra fie este un XT,\n");
    printf ("fie nu are EXTENDED MEMORY instalata.\n\n");
  }
  else
  {
    printf ("\n Calculatorul D-voastra poseda %d kbyte
            Extended Memory.\n\n");
  }
}
  
```

Listing 2 - rutina extram în C

```

; mutaram.asm
;
; muta un bloc de 100 de octeti de la adresa 3000:0000 din mem
; oria conventionala la adresa 0000:0000 din Extended Memory
;
; stiva segment para stack
dw 100h dup (?)
stiva ends

; date segment para 'data'
gdt db 16 dup(0) ; rezervare pentru GDT
;
; adresa de origine:
dw 0ffffh ; lungimea segmentului
db 0,0,3 ; adresa 30.000h
db 97 ; drepturi de acces
db 0,0 ; rezervat

; adresa destinatie
dw 0ffffh ; lungimea segmentului
db 0,0,10h ; adresa 100.000h
db 97 ; drepturi de acces
db 0,0 ; rezervat

; resetarea tabelii GDT
db 16 dup(0)
date ends

program segment para 'code'
assume cs:program, ds:date, ss:stiva
start proc far;
mov ax,date ; incarca adresa segmentului de date in DS
mov ds,ax ; 1000 octeti = 500 cuvinte
mov cx,500 ; functia 87 hexa
mov ah,87h ; DS pe stiva
push ds ; ES = DS
pop es

mov si,offset gdt ; adresa GDT
int 15h ; apel functie
mov ax,4c00h ; termina programul prin MS-DOS
int 21h

start endp
program ends
end
  
```

Listing 3 - program de mutat o zonă de memorie

10 Base-T - Strategii de implementare

10 Base-T este noul standard pentru Ethernet pe cablu bifilar necranat. Implementarea standardului, cu multe funcțiuni noi, este foarte simplă, dacă se respectă anumite reguli.

În ultimele șase luni s-a ajuns la un aflux de prezentări de produse compatibile 10Base-T. Acestea au creat impresia că standardul 10Base-T este de mult omologat. De fapt aceasta s-a încheiat definitiv doar în septembrie. Utilizatorii care și-au instalat deja o rețea Ethernet pe cablu necranat torsadat (Unshielded Twisted Pair/UTP), s-ar putea lovi acum de probleme de transportabilitate și de interope-rativitate.

10Base-T este o specificație pentru Ethernet pe UTP cu o viteză de transfer de 10 MBit/sec. 10Base-T este o extensie a standardului Ethernet IEEE 802.3. Printre altele, specificația se referă la o topologie stea, o cablare structurată, izolarea erorilor și administrare centrală. Instalarea ar trebui să fie în mod normal foarte simplă: cablurile UTP leagă nodurile rețelei cu distribuitoare centrale (Hubs) sau concentratoare aflate în dulapuri de cablare. Fiecare nod al unui LAN 10Base-T dispune de un MAU (Media Access Unit).

Unii producători, ca de exemplu SynOptics, au oferit produse Ethernet UTP chiar înainte de elaborarea standardului. Ca rezultat <Lattis Net> nu este compatibil cu standardul. SynOptics nici n-a pretins acest lucru niciodată. Lattis Net folosește o altă schemă de integrare pentru legături ca și un semnal de 2 V vîrf la vîrf (10Base-T prevede 5 V). În plus Lattis Net folosește așa numita procedură de preegalizare în loc de postegalizare pentru a egaliza distorsiunile de semnal pe cablu. Lattis Net este totuși un produs ce s-a dovedit a fi bun, cu o bază de instalări largă, și

SynOptics a dezvoltat o tehnică pentru integrarea 10Base-T în rețele Lattis Net. Alți producători s-au angajat doar mai târziu pe piața Ethernet-UTP. Ei au avut avantajul că standardul exista deja într-o versiune preliminară.

<Express Net> al firmei David Systems, de exemplu, a fost realizat compatibil cu 10Base-T.

10Base-T este compatibil cu celelalte standarde 802.3, ceea ce înseamnă că integrarea unei rețele LAN 10Base-T într-o rețea Ethernet existentă nu trebuie să fie neapărat complicată și costisitoare. Există două metode de a lega o rețea 802.3 sau chiar nestandard UTP cu 10Base-T. O posibilitate este conectarea segmentului LAN printr-un așa numit AUI (Attachment Unit Interface). Toate rețelele 802.3 sprijină această interfață. Astfel se pot conecta noduri cu diferite specificații Ethernet, ca de exemplu 10Base-5 și 10Base-T. Interfața se va introduce între MAU și Hub. Și concentratoarele pot fi integrate în acest fel.

Produsele 10Base-T pot fi cuplate printr-un port AUI la un Hub sau MAU și cu alte medii de transmisie (de ex. fibră optică). În plus se pot conecta Hub-uri prestandard prin AUI la Hub-urile standard. Cu această metodă se obține o rețea 10Base-T ierarhizată, care leagă mai multe rețele prestandard UTP-Ethernet în LAN. Distribuitorii (Hub) 10Base-T pot fi utilizați la extinderea unui bus Ethernet existent, sau pentru a înlocui un repetitor multiport. Astfel este posibilă, de exemplu, conectarea unui Hub 10Base-T la un segment coaxial sau din fibră de sticlă. În acest mod se pot utiliza cablurile telefonice existente pentru LAN-uri individuale.

În afară de metoda cu AUI mai există o a doua posibilitate pentru integrarea 10Base-T: Un modul de

interfață se conectează direct la Hub sau concentrator. Aceste module posedă mai multe mufe pentru diferite metode de cablare, de exemplu coaxial sau fibră de sticlă. Astfel se pot cupla într-o topologie stea diferite segmente de cablu, ceea ce implică atât avantaje cât și dezavantaje.

Această arhitectură permite integrarea tuturor mediilor de transmisie a segmentelor de rețea individuale pînă la Hub. Nu sînt deci necesare adaptoare pentru UTP. Comunicația dintre modulele gazdă va fi preluată de placa de bază (fundul de sertar) din Hub. Standardul 10Base-T nu furnizează specificații pentru Hub și concentratoare, care dispun de un fund de sertar detașabil. În principiu modulele prestandard Ethernet de interfață și MAU pentru UTP sînt descrise în recomandări. Din această cauză produsele prestandard și 10Base-T pot lucra împreună în același Hub. Avantajul acestei metode constă în faptul că protejează investițiile deja făcute și păstrează funcționalitatea fiecărei părți a rețelei. Dezavantajul este că volumul de muncă al administratorului de rețea, mai ales în LAN-urile mari, se mărește, deoarece trebuie să supravegheze fiecare MAU și să-i atașeze modulul de interfață corespunzător.

În mod obișnuit legăturile prestandard și cele 10Base-T nu funcționează în cadrul aceleiași nodul Host (gazdă). Totuși nu deranjează dacă se racordează un MAU prestandard la un modul 10Base-T. Din cauză că ambele componente utilizează tehnici de conectare diferite, nu recunosc faptul că există conexiunea și nu transmit date. Trebuie să avem întotdeauna grijă ce tip de MAU se află conectat la ambele capete ale segmentului, înainte de a adăuga o legătură.

În prezent se lucrează la dezvoltarea unor proceduri care testează interoperabilitatea produselor 10Base-T. Aceste metode de test sînt dezvoltate în principal de asociațiile de producători, cu toate că și anumite instituții independente lucrează la rezolvarea acestor probleme, de exemplu Universitatea din New Hampshire, Interoperability Laboratory (UNIOL).

Cu metoda UNIOL producătorii pot să-și integreze produsele într-un mediu de la furnizori multipli, să determine și să elimine problemele de interoperabilitate. Rezultatele testului sînt accesibile în orice caz doar membrilor unui consorțiu și respectivilor producători. În testele UNIOL se testează în plus în principal interoperabilitatea și nu respectarea standardului. Astfel este posibil ca produsele care trec aceste teste să fie integrabile într-un mediu 10Base-T și cu toate acestea să nu fie conform standardului.

Pentru a elabora o strategie de integrare a 10Base-T într-o rețea Ethernet existentă trebuie să ținem cont de cîțiva factori importanți: printre altele mărimea și felul bazei hard instalate, mărimea extensiilor planificate 10Base-T cît și locul de amplasare.

Cea mai importantă este în mod sigur mărimea rețelei existente precum și cea a extensiei dorite. Cînd partea instalată este relativ mică și se are în vedere o extindere de amploare, cea mai avantajoasă soluție poate fi înlocuirea echipamentelor existente și reluarea de la început a implementării rețelei. Astfel se poate garanta o rețea omogenă conformă standardului. Dacă baza instalată este deja prea mare, schimbarea din prestandard în MAU-uri 10Base-T nu mai este necesară și în plus este prea costisitoare.

Tipul Ethernet-ului prestandard UTP și locul de amplasare al extensiei planificate sînt factorii următori ce trebuie avuți în vedere. Cînd

rețeaua urmează a fi extinsă într-o nouă clădire sau cu un dulap de conexiuni se recomandă metoda AUI sau legarea cu un modul gazdă (Host) la un Hub. În acest fel fiecare segment al rețelei este clar separat. Pe de altă parte dacă trebuie extins un segment deja existent se recomandă folosirea metodelor de integrare oferite de producătorul instalației.

Cînd mediile 802.3 trebuie implementate mixt, mai există și alte reguli care trebuie respectate, pentru a ne conforma specificațiilor standardului. O parte a acestor reguli sînt conținute deja în recomandările de pînă acum pentru 802.3, altele sînt stabilite în noile specificații 10Base-T. Secțiunea 13 a standardului "System Considerations for Multi-Segment Networks" (considerații de sistem asupra rețelelor multisegment), detaliază regulile care trebuie respectate cînd se implementează medii mixte. Aici aparțin: factori de întîrziere, numărul de repetoare între două terminale, lungimea maximă a segmentului de cablu.

Această parte a standardului 10Base-T furnizează informații despre realizarea unei rețele multisegment în cadrul unui așa numit "Single Collision Domain" (domeniu de coliziune singulară). Ethernet se bazează pe metoda CSMA/CD (Carrier Sense Multiple Access/Collision Detection). Domeniile de coliziune singulară se definesc ca rețele CSMA/CD independente, învecinate. Un repetor nu împarte rețeaua în domenii de coliziune singulară, această funcție o realizează puntea (bridge). De vreme ce un Hub 10Base-T este un repetor, el nu împarte deci rețeaua în domenii independente ci doar îl extinde în principal.

Recunoașterea coliziunilor este o activitate importantă a metodei CSMA/CD și întîrzierea este un factor hotărîtor, care trebuie avut în vedere la realizarea rețelei. Coliziunea totală, care se consideră într-

un domeniu de coliziune, se stabilește ca sumă a întîrzierilor ce rezultă de la MAU, repetoare, echipamente de introducere date și mediul de transmisie. Fiecare din aceste componente produce un anumit factor de întîrziere, care totuși este cunoscut în majoritatea cazurilor. Acești timpi trebuie calculați atunci cînd o rețea este redusă sau extinsă.

Pentru recunoașterea coliziunilor stația emițătoare trebuie să știe dacă un bloc (frame) s-a ciocnit cu altul înainte de terminarea completă a transmisiei. Stația emițătoare va fi informată în acest caz pentru că ea trebuie să repete transmisia blocului. În cazul cel mai defavorabil (Worst Case) standardul 802.3 stabilește: chiar și atunci cînd transmisia a parcurs întreaga rețea și coliziunea a apărut în capătul cel mai îndepărtat, stația emițătoare trebuie să ia cunoștință de acest fapt, înainte de terminarea transmisiei.

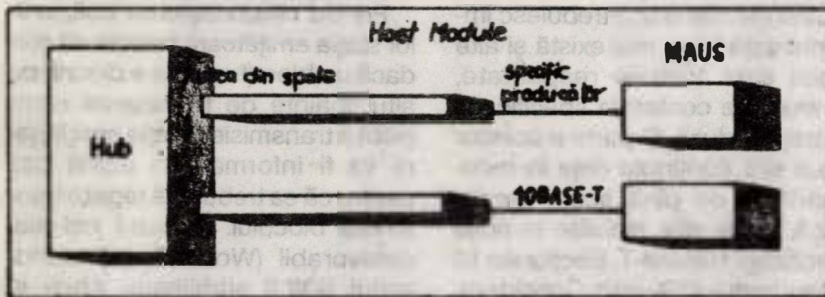
În plus față de întîrzieri, numărul repetoarelor implementabile este influențat și de așa numitele pauze între blocuri (interpacket gap). Aceste pauze între blocuri sînt perioade de timp în care nu are loc nici un fel de transmisie. Cu ajutorul acestor pauze interbloc, aparatele pot identifica începutul și sfîrșitul transmisiei. Lungimea maximă a segmentelor de cablu se află întotdeauna în legătură directă cu felul cablului utilizat. Fiecare clasă 802.3 are o lungime maximă recomandată de IEEE. De exemplu 10Base-2 este recomandată de Ethernet pentru banda de bază cu o viteză de transmisie de 10 Mbit/sec și o lungime de segment de 200 m.

Limitarea lungimii cablului este în legătură cu factorii stabiliți, specifici pentru un anumit mediu de transmisie, de exemplu distorsiunea datelor la transmisie. Toate segmentele dintr-un mediu mixt trebuie să respecte recomandările prevăzute pentru respectivele medii. Așa de exemplu un LAN

10Base 2 nu trebuie să aibă segmente mai lungi de 200 m. Mulți producători dau pentru produsele lor o distanță posibilă mai mare, totuși trebuie avute în vedere lungimile care sînt prevăzute de specificațiile IEEE. Doar astfel se poate realiza independența în perspectivă față de un anumit producător.

existentă sau într-un LAN mixt este important să realizăm în prealabil o diagramă a rețelei. Diagrama trebuie să cuprindă locul de amplasare al tuturor aparatelor precum și lungimile și felul cablurilor. La conectarea unor echipamente noi trebuie văzut, dacă au fost respectate recomandările Ethernet. Re-

asigurat că această prescripție este respectată, trebuie să calculeze factorii de întârziere pentru fiecare componentă și distribuira rețetelor. Calculele vor arăta de asemenea dacă are sens o punte sau un alt mediu de transmisie.



O instalare nouă Ethernet sau o extensie a unei baze instalate nu necesită o planificare minuțioasă. Cînd trebuie integrate componente 10Base-T într-o rețea Ethernet

comandările 10Base-T prevăd ca nici un echipament să nu se afle la o distanță mai mare de o sută de metri de la Hub-ul corespunzător. După ce planificatorul rețelei s-a

Definitivarea standardului 10Base-T a declanșat un aflux de anunțuri de noi produse. În plus a crescut în continuare acceptarea de Ethernet pe cabluri bifilare necranate (UTP). Integrarea acestei tehnologii pare la prima vedere destul de complicată, dar este relativ simplă, dacă regulile de implementare sînt respectate.

(Greg Cambell)

Greg Cambell lucrează pentru Grupul Interconnect Consulting, o întreprindere care s-a specializat în planificarea, instalarea și asistență pentru LAN și WAN. Cambell conduce sucursala din San Francisco, California USA.

Unix și virușii

(continuare din pag. 15)

Împotriva exploatării slăbiciunilor sistemului, așa cum a practicat-o cu succes virusul Internet, nu ajută decît grija permanentă asupra sistemului și corectarea tuturor punctelor slabe devenite cunoscute. Prea multe puncte slabe le sînt cunoscute celor care administrează sistemul, dar nu sînt 'tratate' din comoditate sau din lipsă de simț al responsabilității. Dacă o astfel de problemă nu poate fi corectată fără probleme, mai bine se renunță la utilizarea softului respectiv.

Chiar la atacuri prin persoane care lucrează direct la sistem, utilizatorii autorizați sînt punctul cel mai nevraigic. În afară de deja pomenitele păcate, există o serie întregă de obiceiuri care îi ușurează sarcina unui agresor. Astfel de exemplu deseori parolele sînt notate în locuri ușor accesibile, uneori chiar direct la terminal. Terminale nesupravegheate, unde sînt logați utilizatori privilegiați sau chiar administratorul sistemului, nu sînt nici ele neapărat o excepție. Un intrus 'pe fază' poate în cîteva clipe să își asigure și pe viitor accesul. Abstracție făcînd de paguba pe care un răuvoitor poate să o provoace uneori chiar și ou o singură comandă. Aici și administratorul

Unix-ului poate să facă cîte ceva. Mediul multiproces al Unix-ului permite activarea unor așa-ziși "demoni". Aceste programe pîndesc în background, devenind active numai cînd se întîmplă anumite evenimente. Astfel de exemplu un "idle"-demon, ușor de instalat, poate servi supravegherii utilizatorilor logați în sistem. Dacă mai mult timp nu se întîmplă nimic, sesiunea se termină în mod automat. Și controlul activităților nocturne poate fi organizat în felul acesta.

Rezumînd, se constată că printr-o serie de caracteristici, cum ar fi uzuala conectare la rețea, Unix este mai expus atacurilor decît multe alte sisteme. Mecanismele de protecție în forma de pînă acum sînt insuficiente și au multe breșe. Arhitectura sistemului conține însă posibilitățile necesare pentru a le aduce pe acestea la un nivel rațional pentru practică, fără un efort deosebit. Astfel treptele de securitate C1 și C2 pot fi atinse în acest fel și sînt deja realizate într-o serie întregă de sisteme. Trepte de securitate mai înalte pot fi atinse în principiu, însă cu un efort considerabil. Față de MS-DOS sau chiar și OS/2, Unix este sistemul mai sigur, chiar și fără măsuri suplimentare. Cea mai slabă verigă în lanțul măsurilor de protecție este însă utilizatorul - și acest lucru e valabil nu numai pentru sistemele Unix.

Baze de date

Mult timp dBase III Plus s-a aflat în fruntea sistemelor de gestiune a bazelor de date (SGBD). dBase IV trebuia să întărească acest rol conducător. Prima încercare însă a dat greș, noua versiune lansată acum pe piață își propune să recuce-rească tronul, dar concurența nu doarme.

Atunci când PC-urile au început să aibă succes, ele au fost folosite cu precădere în trei domenii de activitate: prelucrare de texte, calcul tabelar și administrarea datelor. PC-urile însă nu erau destul de performante pentru cerințele unui SGBD. Inițial "bazele de date" nu erau decât simple cartoteci. Aplicațiile care se bazează pe prelucrarea datelor - cum ar fi de exemplu facturarea sau contabilitatea - erau scrise cu ajutorul unor limbaje obișnuite de programare.

Aceasta s-a schimbat abia atunci când pe piață au apărut SGBD-uri puternice, care permiteau accesul programatorilor și la colecțiile de date externe. dBase II era doar un simplu program de prelucrare a unei cartoteci, dBase II a fost totuși un sistem puternic. Avantajul acestui program a fost limbajul său simplu. Sistemele anterioare erau scrise prea des de programatori, pentru programatori, utilizatorii fiind lăsați pe dinafară. dBase în schimb folosește un limbaj cu ajutorul căruia pînă și neinițiatii se pot descurca intuitiv. Urmarea acestui avantaj a fost acapararea a mai mult de 60% din piață.

Odată cu creșterea performanțelor hardware-ului, s-au modificat și cerințele utilizatorilor. PC-urile își pierd rolul de "calculatoare personale" și devin, datorită cuplării în rețele, tot mai mult, părți ale sistemului informatic al unei întreprinderi. Pentru SGBD-uri aceasta înseamnă satisfacerea unor cerințe sporite, mai ales în

ceea ce privește posibilitatea de cuplare în rețea. Ashton-Tate s-a moșmondit prea mult cu această dezvoltare și concurența a putut cîștiga teren.

Astăzi PC-urile au devenit o concurență serioasă pentru multe minicalculatoare. (Mainframe-urile nu sînt amenințate, PC-urile neavînd încă puterea necesară. Pe de altă parte piața de mainframe-uri este saturată, așa încît creșterea se va face în domeniul PC-urilor.) Slogaturile anilor viitorilor fi downsizing, cuplare în rețea și concepte client-server.

Atunci când cei de la Ashton - Tate au observat că sînt pe cale "să rămînă de căruță", și-au propus să facă un salt cantitativ. dBase IV trebuia să fie nu doar urmașul celui mai renumit SGBD, ci să marcheze și "the-state-of-the-art" pentru mulți ani. SQL-ul și managementul tranzațiilor sînt doar două din facilitățile preluate din lumea calculatoarelor mari. Compatibilitatea în jos, spre dBase III, trebuia însă să rămînă, pentru ca dBase să poată deveni veriga de legătură între aplicațiile monoutilizator și sistemul complex de administrare a datelor.

Prima încercare a dat greș, deoarece dezvoltarea lui dBase IV s-a făcut în pripă și produsul a fost plin de erori. A urmat o dezvoltare de aproape doi ani, cu una din cele mai costisitoare faze de beta-testare din istoria prelucrării electronice a datelor. Acum a fost prezentat dBase IV 1.1. Dacă se ține cont numai de caracteristicile de putere anunțate, acest program are toate atuurile să rămînă o referință pentru anii următori. Dacă însă nu-și respectă promisiunile, atunci Ashton-Tate își poate scoate firma la vînzare. Mediul în care se mișcă dBase IV este multiplu. Alte SGBD-uri, cu concepte în parte contradictorii, încearcă să satisfacă diferitele cerințe ale grupurilor de utilizatori

individuali. Utilizatorii care doresc să-și administreze datele într-un mod mai simplu pot folosi sisteme conduse prin meniuri, ca de exemplu Dataease, cu care chiar și neinițiatii, fără cunoștințe de programare, pot ajunge la rezultate satisfăcătoare. Pentru administrarea unor volume mari de date există sisteme performante cum ar fi Paradox. Cerințele programatorilor sînt satisfăcute de SGBD-uri cu compilatoare, cum ar fi Clipper și Quicksilver, compilatorul SGBD-ului dBase XL.

Versiunile PC ale marilor SGBD-uri, cum ar fi Ingres, Informix sau Progress, sînt concepute de producători doar ca versiuni de inițiere. Ei își vînd sistemele cu suport mult și nu găsesc în afacerea PC marjele necesare. În plus deservirea acestor sisteme, de multe ori, nu corespunde standardelor cu care ne-am obișnuit la PC-uri.

Drumul victorios al suprafețelor utilizator grafice, început de Windows 3.0, încă nu a produs efecte notabile asupra bazelor de date. Pentru producătorii tradiționali, ca Ashton Tate sau Borland, investiția necesară pentru transpunerea sistemelor lor sub Windows este enormă. Dacă însă pentru SGBD-ul cu care se lucrează nu există o versiune Windows, atunci cheltuielile de transpunere sînt enorme pentru utilizatori, pentru că de regulă pot fi preluate doar fișierele de date, nu și programele. De aceea SGBD-urile Superbase4 sau Omnis sînt folosite mai ales pentru proiecte noi.

Decizia de a utiliza un SGBD sau altul nu trebuie să depindă numai de performanțele acestuia, ci și de problema care trebuie rezolvată. Nu programul cel mai bun este și cea mai bună soluție, ci programul cu ajutorul căruia problema pusă poate fi cel mai bine rezolvată.

(Computer Persönlich 20/90,
Robert Bäurle)

dBase IV 1.1 - revenire reușită

Chiar și în cazul lui dBase, SGBD-ul care a impus la PC-uri standardul pentru această categorie de software, programul se adresează în principal la două categorii de utilizatori: utilizatorilor finali, care își pot administra foarte confortabil datele folosind "centrul de control", și proiectanților și programatorilor care scriu aplicații cu dBase. După versiunea 1.0, care a avut mai multe defecte decât calitate, Ashton Tate a lansat acum pe piață un produs lucrat cu migală și testat profund. Ca orice alt software, nici dBase IV nu este fără cusur, dar este mai sigur și mai performant.

Față de versiunea inițială s-au făcut transformări în patru domenii: necesarul mai redus de memorie și deservirea mai simplă se adresează în primul rând utilizatorilor finali. Funcțiile și comenzile create pentru obținerea unei viteze de lucru mai mari, dezvoltarea limbajului de programare dBase, cât și mai buna conlucrare dintre limbajul dBase și SQL (Structured Query Language), se adresează în primul rând programatorilor și proiectanților. De aceste îmbunătățiri ale performanțelor se bucură, bineînțeles, și utilizatorii finali.

Prin folosirea mai economică a memoriei, dBase IV necesită acum doar 450 KBytes, față de 516 KBytes cât necesita înainte. Se ajungea destul de des la probleme atunci când memoria trebuia împărțită între programe dBase foarte mari, drivere de rețea și alte programe rezidente în memorie. Reducerea spațiului de memorie utilizat are două aspecte pozitive: aplicațiile individuale au la dispoziție mai mult loc în memoria principală și simultan pot fi încărcate în memorie drivere de rețea mari.

Deservirea lui dBase a devenit mai confortabilă. Din opțiunile "Browse" și "Edit", poate fi accesat

direct meniul de administrare. Pe această cale se poate face o prelucrare rapidă de index. Pentru utilizator aceasta reprezintă o simplificare esențială a lucrului, deoarece din cadrul acestui meniu el poate adăuga, modifica sau șterge foarte ușor indexuri. În versiunea anterioară acest lucru nu era posibil.

La conceperea unei interogări, în versiunea 1.0, lățimea coloanei era limitată la 21 de caractere. Rezultatul unui QbE - interogare prin exemplu (Query by Example) - era prezentat pe ecran în maxim patru coloane. În versiunea 1.1 aceste limite au fost ridicate. Îmbunătățirile din domeniile optimizării programelor și tuning-ului se adresează în primul rând proiectanților. Viteza de lucru a programelor dBase poate fi sporită prin diferite metode.

Acum dBase conține o opțiune de Disk-Cache, cu ajutorul căreia memoria extinsă sau expandată poate fi folosită ca buffer de memorie pentru accesul la harddisk. Dacă datele se află deja în memoria cache, atunci ele stau mai repede la dispoziția prelucrărilor. Încă de la instalare se hotărăște dacă se folosește memoria extinsă sau expandată ca memorie cache. Parametrii independenți pot fi modificați și ulterior. Dintre aceștia amintim parametrul "/w", care se folosește atunci când "se scrie direct pe harddisk", în loc să se folosească memoria cache. Dacă se dorește modificarea parametrilor, atunci se poate lansa direct din dBase programul "dBCache". Programul Cache nu funcționează decât împreună cu dBase IV. Trebuie să avem grijă, însă, ca în același timp să nu fie active și alte programe Cache, deoarece aceasta ar duce la un dezastru al datelor.

Ashton Tate a conceput un sistem propriu de optimizare a admi-

nistrării memoriei, denumit "sistem dinamic de management al sistemului" - dMMS (dynamic Memory Management System) - în a căru sarcină cade administrarea dinamică a overlay-urilor (a zonelor reacoperibile). Pe scurt: dBase-ul propriu-zis necesită 350 KBytes de memorie centrală, restul de 100 KBytes sînt rezervați administrării dinamice a zonelor reacoperibile. Zonele reacoperibile sînt părți de dBase care nu trebuie să se afle în permanență în memoria centrală. În afară de dBase în memoria centrală mai este necesar și de loc pentru aplicații. În acest punct trebuie să se determine spațiul necesar aplicației, iar memoria centrală rămasă stă la dispoziția administrării zonelor reacoperibile.

Pentru împărțirea spațiului de memorie, dBase IV folosește variabila de mediu dBHeap. Cu această variabilă se poate stabili modul de împărțire procentuală a spațiului de memorie între aplicații și zonele reacoperibile. Variabila poate lua valori între 1 și 100. Dacă se lucrează preponderent prin intermediul "centrului de control", atunci valoarea recomandată de Ashton Tate este dBHeap=10. În acest mod se rezervă mai mult spațiu pentru zonele reacoperibile, ca de exemplu pentru măștile dBase și pentru generatorul de rapoarte. Dacă dBHeap=90, atunci spațiul este rezervat aproape integral aplicațiilor. Deoarece dBHeap nu poate fi modificată din dBase, ea fiind o variabilă DOS, fixarea valorii ei trebuie făcută sub sistemul de operare, de exemplu în AUTOEXEC.BAT.

Ashton Tate a îmbunătățit atât limbajul dBase cât și cooperarea dintre acesta și SQL. O comandă nouă este "Keyboard". Cu ajutorul ei pot fi scrise o serie de caractere în bufferul tastaturii. dBase poate prelucra apoi aceste caractere ca și cum ar fi fost tastate de utilizator.

Ea poate fi folosită, de exemplu, la programe demonstrative care se parcurg singure, sau pentru selecția automată a unui meniu. Până acum această comandă exista numai în dialectele dBase.

În mod asemănător ca și la SGBD-ul "Foxpro", compatibil dBase, și în dBase IV există acum posibilitatea "indexării condiționate". Folosind o clauză FOR în comanda de indexare se poate formula o condiție care să trebuiască a fi îndeplinită pentru ca un articol să fie preluat în index. Un exemplu de aplicare ar fi construirea unui index pentru abonații din Sălaj, și nu pentru abonații din toată țara. Avantaje: se face economie de spațiu și se mărește viteza de căutare în index. Un index condiționat nu poate fi folosit pentru fișierele cunoscute până acum în dBase, cu extensia .NDX, ci doar la cele cu extensia .MDX.

S-a extins și domeniul de utilizare pentru funcțiile definite de utilizator (UDF) și pentru comenzile ON. Până acum nu era posibilă executarea unor funcții recursive cu funcții definite de utilizator. Restricțiile existente pot fi ocolite in-

cluzindu-se în fișierul CONFIG.DB comanda "DBTrap=Off". Ce-i drept această operație trebuie făcută cu atenție. Implicit DBTrap=On, și valoarea variabilei ar trebui schimbată doar de programatorii cu experiență care pot prevedea efectele acestei modificări.

S-au făcut modificări esențiale și în modul SQL al lui dBase IV. Cu noua versiune, în modul SQL pot fi lansate și funcții ale centrului de control. În modul SQL pot fi folosite acum și comenzi dBase, nu doar comenzi SQL, iar în fișierele SQL pot fi inserate chiar și funcții scrise de utilizator.

În cazul instalării versiunii de rețea, dBase IV 1.1 poate fi folosit în următoarele tipuri de rețele: IBM PC Net începând cu versiunea 1.2, Novell Netware 286, Novell Netware 386 și 3Com3+ Share. În afara versiunii standard și a celei de rețea mai există și o versiune specială pentru programatorii și casele de software care scriu aplicații cu baze de date.

Un programator își poate răspîndi aplicația sa împreună cu un modul Runtime, fără ca pentru

aceasta să trebuiască să plătească licențe la Ashton Tate. Producătorul afirmă că programele scrise cu această versiune sînt valabile și pentru rețea. Pachetul de proiectare conține în mod standard trei licențe, aceasta permițînd testarea confortabilă a aplicațiilor multiuser. Această versiune costă 3819 mărci.

Demne de amintit sînt și noile drivere de imprimantă, ca și cele actualizate. Au fost adăugate drivere suplimentare pentru modelele noi de imprimante, printre ele numărîndu-se și un driver Postscript pentru imprimanta Apple-Laser-Writer și pentru imprimante compatibile Postscript.

Cu toate că dBase IV 1.1 stă acum la dispoziția utilizatorilor, ei așteaptă în continuare o ediție server și compilatorul dBase promis de Ashton Tate. După informațiile obținute în aceste domenii se lucrează zi și noapte, dar cu noile cerințe de calitate, autoimpuse, va trebui să mai avem răbdare.

(PC Magazin 43/90,
Jürgen Wasem-Gutensohn)

Paradox 3.5 - o punte între două lumi

Borland oferă cu Paradox 3.5 în combinație cu SQL Link unul din primele instrumente Front-End pentru servere de baze de date.

Sistemul de gestiune al bazelor de date (SGBD) Paradox, ocupă locul doi în lume, în domeniul bazelor de date pe PC-uri, după dBase. Acesta din urmă a pierdut în ultimii ani părți importante din piață, atît în SUA cît și în Europa. După părerea specialiștilor, în ultimul trimestru al anului 1990, Paradox a depășit pentru prima dată dBase-ul ca număr de instalări noi.

Paradox este, în principal, un soft pentru trațarea confortabilă a

datelor. Cu ajutorul lui pot fi făcute manipulări complexe și cu-prinzătoare de date chiar și fără programare. Prima versiune Paradox a fost realizată în 1985. Sistemul are așadar o tradiție asemănător de lungă cu dBase-ul.

Un important semn de recunoaștere al Paradox-ului este tehnica Query-by-Example (QbE), un sistem interactiv de interogare. Într-un formular prezentat pe ecran sub forma unei tabele se marchează cîmpurile dorite, selecția puțîndu-se face făcînd uz de operatorii aritmetici și logici obișnuiți. La Paradox, ca și la dBase, o tabelă

este identică cu un fișier al bazei de date.

De facilități QbE, pentru analiza și manipularea datelor, dispune și dBase-ul, dar forma de vizualizare tabelară a Paradoxului, este sub acest aspect mai convingătoare.

Paradox este singurul dintre bazele de date de marcă care conține opțiunile standard de grafică integrate. Același lucru este valabil și pentru prezentarea datelor sub formă de tabele încrucișate. Cu acestea informațiile din tabele pot fi analizate asemănător ca la spreadsheet-uri.

Datorită tehnologiei VROOM (Virtual Runtime Object Oriented Memory Manager) și componentului Turbo-Drive, după datele producătorului, Paradox 3.5 a devenit mai rapid și mai eficient. VROOM, cunoscut deja de la programul de calcul tabelar Quattro Pro al firmei Borland, Paradox este descompus în componente mici, numite module. Aceste bucăți de cod sînt încărcate în memorie numai atunci cînd este nevoie efectivă de ele. În acest mod în memorie rămîne mai mult loc pentru date, de exemplu pentru operațiile de sortare mari consumatoare de memorie. VROOM preia în mod automat utilizarea memorie expandate sau extinse.

Turbo-Drive se îngrijește de adaptarea produsului Paradox la tipul procesorului disponibil, fie el un Intel 80286, un Intel 80386 sau un Intel 80486. Cu ajutorul acestei rutine, Paradox poate controla maxim 16 MBytes de memorie extinsă. Pînă acum acest lucru era posibil numai cu produsul Paradox 386, o versiune specială de Paradox optimizată pentru procesorul 386, care de la apariția versiunii Paradox 3.5 a fost retrasă de pe piață. Datorită tehnologiei îmbunătățite de management a memoriei, pusă la punct de Borland, Paradox poate lucra cu volume de date și programe mai mari decît pînă acum. Dacă sistemul dispune de memorie extinsă, Turbo-Drive comută automat Paradox-ul în Protected Mode, cu urmarea că de acum înainte toate acțiunile Paradox se pot desfășura direct în memoria principală. Performanțele se îmbunătățesc în acest mod simțitor.

Setul de instrucțiuni și funcții oferit de PAL (Paradox Application Language, limbaj propriu pentru dezvoltarea de aplicații) a fost extins în noua versiune. Pentru dezvoltarea de aplicații mai complexe este oferit mediul Paradox-Engine. Dar cel mai important atu al Paradox-ului îl constituie și pe mai departe tratarea confortabilă a datelor de către utilizatorul final. Pro-

duse asemănătoare în acest segment de piață sînt dBase, R:Base sau dBasease.

Prin intermediul limbajului de programare PAL și Paradox dispune de unelte pentru dezvoltarea de aplicații. Comparativ cu limbajul dBase și toate variantele lui, sau cu sistemul de dezvoltare al bazelor de date Clipper, PAL este utilizat într-o măsură mai redusă pentru elaborarea de programe. Pentru cine însă programarea este foarte importantă, este pus la dispoziție mediul Paradox-Engine, un add-on, cu ajutorul căruia datele dintr-o bază de date Paradox pot fi accesate de aplicații scrise în limbaj C, cu Turbo-C sau Microsoft-C.

Paradox-Engine este gândit pentru programatori și conține o interfață de dezvoltare a aplicațiilor (API, Application Development Interface), cu ajutorul căreia programatorii în C pot scrie aplicații care să poată utiliza tabelele Paradox în modul de operare stand-alone sau în rețea conlucrind cu un file-server.

Făcînd o comparație între Paradox și alte SGBD-uri cum sînt Clipper, dBase sau chiar R:Base, în ceea ce privește facilitățile pentru dezvoltarea de programe, acestea din urmă ies în avantaj. Pentru limbajele de programare ale acestora există compilatoare care protejează munca programatorului. Programele sursă odată compilate nu mai sînt vizibile pentru nimeni. Și Paradox oferă o astfel de facilitate. Dacă un programator vrea să dea și altora programele scrise de el în PAL, are nevoie de un modul Runtime. Licența pentru acest modul trebuie plătită însă separat și costă 98 DM.

Enumerăm alte noutăți aduse de Paradox 3.5: într-o machetă multitalară pot fi cuplate nouă tabele, față de doar cinci tabele în versiunea anterioară, Paradox poate lucra acum și în rețele 3+Open. În modul de operare stand-alone este suficientă o memorie de 512

KBytes, pentru lucrul cu Paradox. Împreună cu SQL Link, pentru accesul la servere IBM sau Microsoft este necesară o memorie de 1 MByte iar pentru accesul la un server Oracle este necesară o memorie de 1,5 MBytes.

Pentru cuplarea cu lumea exterioară este disponibilă o paletă bogată de formate de date, cum ar fi ASCII, dBase sau Lotus, în care datele pot fi exportate sau importate. Același lucru este valabil și pentru programul de calcul tabelar Quattro Pro. Paradox poate citi fișierele scrise în formatul acestuia, sau poate scrie fișiere în formatul lui.

Cea mai importantă caracteristică a noii versiunii o constituie, pentru utilizatorul Paradox, programul auxiliar SQL Link. El lucrează numai împreună cu Paradox 3.5. Dacă într-o rețea sînt instalate ambele produse, comenzile SQL de interogare date, definire date și manipulare date, sînt arătate automat în sistemul de meniuri al lui Paradox 3.5. Dacă SQL Link nu este instalat, punctele corespunzătoare din meniuri lipsesc.

Cu ajutorul acestui auxiliar, utilizatorii Paradox pot accesa într-o rețea datele SQL ale unei baze de date aflate pe server, de parcă ar fi vorba de tabele Paradox.

Produsul add-on oferă un acces transparent la date de la un IBM Database-Manager, Server Oracle, sau Server SQL Microsoft. După datele producătorului, vor urma extensii pentru a permite accesarea și a altor tipuri de servere de baze de date.

Paradox SQL Link traduce comenzile și interogările Paradox în instrucțiuni SQL, și se îngrijește de comunicația fără probleme cu server-ul disponibil. Acordarea la serverul existent oferit de IBM, Microsoft sau Oracle, bineînțeles că este preluată automat de SQL Link.

Pentru a putea accesa, de la un post de lucru Paradox, datele aflate pe un server trebuie pornit mai întâi programul Set-up. El cercează tabelele SQL ale serverului cu care este cuplat, și face copii după toate aceste tabele pe stația de lucru locală. Aceste tabele sînt goale dar au aceeași structură ca și tabelele de pe server-ul bazei de date.

Manipulările de date care implică fișierele originale SQL, sînt efectuate mai întâi în copiile locale și abia apoi sînt transmise prin SQL Link server-ului bazei. În acest mod sînt respectate și toate cerințele sistemelor tranzacționale. Conform regulilor stricte ale acestora o succesiune logică de acțiuni nu se poate executa decît fie complet fie deloc. Respectarea principiilor sistemelor tranzacționale în accesul în comun la colecțiile de date ale unui server de baze de date este necesară pentru asigurarea integrității datelor.

SQL Link traduce interogările formulate prin sistemul de meniuri Paradox, sau prin QbE, în instrucțiuni SQL. O funcție plină de bun simț este oferită de instrumentul ShowSQL, care afișează într-o fereastră forma SQL a cererii formulate prin QbE. Rezultatele înter-

rogărilor sînt prezentate pe ecran sub formă de tabele.

Din păcate, în versiunea actuală sînt neglijate interogările ce se adresează tabelelor SQL ale server-ului. Cereri care să permită combinarea de date aflate în tabelele Paradox locale și în tabelele SQL Remote, sînt tot la fel de puțin permise ca și formularea de cereri care să se refere la două servere de baze de date. Aceste restricții vor fi ridicate abia în versiunile ulterioare. Cine dorește totuși să prelucreză împreună date aflate în tabele locale și în tabele SQL, trebuie să formuleze întrebări distincte iar rezultatele să le stocheze local. Pe această cale ocolitoare, datele pot fi prelucrate local ca și tabele Paradox și din ele pot fi obținute, de exemplu, și grafice.

Paradox SQL Link lucrează cu comenzi interactive SQL și cu instrucțiuni SQL imbricate în programe PAL. În programele PAL, ele sînt demarcate de cuvintele cheie SQL... ENDSQL. În acest mod pot fi scrise aplicații cu baze de date care se bazează pe arhitecturi client-server.

Paradox SQL Link este un instrument pentru analiza descentralizată a datelor și evaluări de date

care sînt memorate central pe un server de baze de date. Nici una din celelalte SGBD-uri de pe PC-uri nu dispune în prezent de un astfel de instrument Front-End. Există astfel de produse anunțate de Ashton-Tate, Dataease și Nantucket, dar pînă în prezent Borland este singurul producător care le și oferă.

Datorită structurii sale Paradox nu are probleme cu SQL. Toate operațiunile managementului SGBD-urilor se bazează pe o prezentare tabelară și lucrează, ca și SQL-ul, cu conceptul matematic al teoriei mulțimilor. Rezultatele definiției, manipulării și interogării datelor sînt tratate ca o mulțime, în sens matematic. Spre deosebire de acest mod, dBase lucrează orientat pe articole.

Paradox Multipack înlocuiește produsul Paradox Netpack, oferit pînă acum, și este un pachet de programe pentru dezvoltarea unei baze de date în rețea cu un file-server. Prețul de 2793 DM a rămas neschimbat, Paradox 3.5 costă 2451 DM și Paradox SQL Link costă 1600 DM.

(PC Magazin 47/90)

R:Base 3.1 - examen de maturitate

R:Base a apărut pe piață în 1984, primele versiuni numindu-se R:Base 4000 și apoi R:Base 5000, pentru ca să se ajungă astăzi la R:Base 3.1. Inițial produsul a fost dezvoltat de firma Wayne Ericson ca un sistem relațional de tratare a informațiilor, ca urmare a unei comenzi făcute de NASA în cadrul proiectului Shuttle. Pînă nu se sfîrși încă anul, produsul era deja difuzat de firma Microrim. O dată cu noile versiuni Microrim a început din nou să desfacă produsul, de data aceasta însă sub regie proprie.

Versiunea standard a R:Base-ului pentru DOS, permite lucrul într-o rețea cu maxim trei utilizatori. Pentru mai mulți utilizatori trebuie folosite programe auxiliare speciale. Produsul dispune de interfețe care permit prelucrarea datelor R:Base din C, Pascal sau Fortran. Ca o extensie a generatorului de rapoarte integrat, se livrează programul Expanded Reportwriter, care permite o descriere foarte ușoară a raportului, evaluări matematice și chiar și funcții pe șiruri de caractere. Pentru reprezentări grafice este livrat modulul DB-Grafix,

asemănător programelor speciale de reprezentări grafice.

Operarea în R:Base se poate face în două moduri: prin intermediul meniurilor, sau în mod comandă. Pentru utilizatorii experimentați operarea în cel de-al doilea mod este mai comodă deoarece nu mai implică "plimbarea" prin meniuri pentru a se ajunge la funcția dorită. Operarea orientată pe meniuri aduce așa-numita tehnică de dare a comenzilor "Prompt by Example" (PBE) - dare a comenzilor după model - care împreună cu funcțiile

de Help și cu manualul, permit utilizarea produsului și de către utilizatorii neexperimentați. Este de asemenea permisă și interogarea prin exemplu - QBE (Query by Example).

SGBD-ul dispune de generatoare care permit rapid și simplu prin intermediul meniurilor crearea unei baze de date, construirea de aplicații, generarea de măști și elaborarea de rapoarte. Produsul are implementat și limbajul SQL, compatibil 100% ANSI nivelul 2 și cu extensii DB2.

Sînt furnizate, de asemenea interfețe care permit importul și exportul de date dintr-o sursă de date de formate. Amintim doar formatele ASCII, dBase, Lotus, PFS și PIF. Pentru asigurarea securității datelor pot fi folosite două comenzi: fie comanda SQL "Grant-Revoke", fie comanda "Read-Modify-File", cu ajutorul căroră tabelele și formulele pot fi protejate în cazul accesului concurent la date.

R:Base dispune de două compilatoare. Primul poate traduce o aplicație în pseudocod și este de fapt o soluție Runtime, care permite economisirea de timp și spațiu, ca și excluderea intervențiilor nedorite în programe. Al doilea compilator permite convertirea

programelor în fișiere executabile - .EXE -care pot fi rulate apoi independent sub sistemul de operare DOS. Aceste programe pot fi date mai departe fără să mai fie necesară plățirea unor licențe. În cadrul aplicațiilor compilate rapoartele se pot obține de două, trei ori mai repede, măștile apar pe ecran cu viteza fulgerului, iar aplicațiile pot fi rulate cu viteze de trei, cinci ori mai mari decît cu un interpretor. Programele sînt protejate, deoarece codul este transformat în limbaj mașină, și întrucît utilizatorii finali nu mai trebuie să cumpere și ei SGBD-ul putînd folosi direct aplicațiile compilate elaborate de proiectanți, ei vor economisi atît timp cît și bani. Cu ajutorul compilatorului pot fi adăugate și rutine scrise cu Microsoft C, sau cu Microsoft Assembler. Pentru depanare este livrat și un DEBUGGER profesional.

Ce se mai "coace" la Microrim?

Microrim și-a anunțat intrarea pe piață cu un nou SGBD Client / Server, care se va numi Vanguard și a fost dezvoltat sub numele de cod Atlas.

Vanguard va fi prima "soluție totală" și va putea lucra sub OS/2 Presentation Manager, Windows, Apple Macintosh, DEC VAX/VMS,

UNIX, MVS și VM IBM. Versiunea pentru OS/2 trebuia să fie livrabilă în vara lui 1990, celelalte versiuni urmînd a fi terminate în următoarele 18 luni.

Acest SGBD va satisface cerințele utilizatorilor în trei domenii: utilizare prietenoasă, interoperabilitate și integritatea datelor. Utilizare prietenoasă: Vanguard va putea fi învățat și utilizat ușor și rapid atît de începători cît și de specialiști, atît de utilizatori cît și de programatori. Interoperabilitate: Vanguard va putea fi rulat pe toate sistemele existente într-o întreprindere, și va putea lucra și împreună cu alte SGBD-uri existente deja. Integritatea datelor: Vanguard va proteja cea mai prețioasă comoară pe care o deține o întreprindere - informațiile sale - chiar și în condițiile în care la date au acces și nespecialiștii.

Microrim a început dezvoltarea lui Vanguard, sub numele de cod Atlas, încă din mai 1987. David Hull, președintele firmei Microrim, este de părere că: "Vanguard este un nume ideal, pentru un software care vrea să arate direcția în care vor evolua SGBD-urile generațiilor viitoare. Cuvîntul Vanguard desemnează poziția conducătoare într-un domeniu, într-o tendință. Vanguard va deveni prima soluție pentru clienții care caută o soluție totală pentru schimbul de date într-o întreprindere." Vanguard s-ar putea traduce prin avangardă, pionier, înaintaș.

Esențial la soluția Vanguard este faptul că acest SGBD va putea lucra și cu baze de date create cu SGBD-uri străine, cum ar fi de exemplu dBase, SGBD-ul IBM DB/2 Extended Edition, Oracle, DB2 și altele. Aceasta înseamnă că Vanguard va putea citi, modifica și scrie astfel de date. În acest fel bazele de date și aplicațiile existente într-o întreprindere vor putea fi utilizate în continuare, nu vor trebui modificate, și vor putea fi legate între ele pentru efectuarea de schimburi de date.

(ing. Romulus Maier)

	dBase IV 1.1	Paradox 3.5	R:Base 3.1
Producător	Ashton Tate	Borland	Microrim
Preț monoutilizator	2679	2451	2622
versiune rețea	3363	2793	2907
Necesar de memorie	640 KBytes	512 KBytes	640 KBytes
Meniuri pull-down	*	*	*
Număr articole	1 miliard	2 miliarde	nelimitat
Dimensiune articol	4000 Bytes	4000 Bytes	4096 Bytes
Număr cîmpuri	255	255	800
Precizia de calcul (zecimală)	20	15	15
Fișiere deschise simultan	99	nelimitat	80 tabele
SQL	*	cu SQL Link	*
Limbaj de programare propriu	*	*	*
Protecție prin parolă	*	*	*

Compilatoare de baze de date

După cum se poate vedea din articolul precedent, SGBD-urile au devenit unelte confortabile de prelucrare a datelor. Cu ajutorul generatoarelor pot fi obținute automat machete de introducere a datelor, etichete, rapoarte sau chiar aplicații întregi.

Aplicațiile cu baze de date, care se bazează pe un SGBD, au avantajul că permit foarte ușor modificări și că permit utilizatorului un acces direct la date, nu numai prin intermediul funcțiilor pe care i le pune la dispoziție aplicația. Dezavantajul îl constituie prelucrarea relativ lentă a datelor. În afară de aceasta pentru fiecare aplicație în PC trebuie să existe SGBD-ul complet, sau cel puțin un modul Run-Time, ceea ce ridică costurile. Dar cel mai mare dezavantaj îl constituie faptul că atât datele cât și programele sînt "deschise", oricine putînd avea acces la ele.

Compilatoarele de baze de date traduc limbaje de programare specializate, specifice unor SGBD-uri, în cod mașină. Datorită lor, la rulare, SGBD-ul propriu-zis nu mai este necesar, programul are o viteză de execuție mai ridicată, iar datele și sursa programului sînt protejate.

În continuare vă prezentăm trei compilatoare de baze de date: Clipper în noua versiune 5.0, Clarion Developer în versiunea actuală 2.1 și Foxpro 1.0. Fiecare din cele trei programe a fost implementat de mii de ori și și-a dovedit calitățile atât în proiecte mici cât și în proiecte mai mari. Dar cu toate că toate trei au fost dezvoltate pornindu-se de la aceleași cerințe și pot fi rulate pe același hardware, fiecare se bazează pe o filozofie proprie.

Foxpro și Clipper au un trecut comun: ele sînt compilatoare dBase. Cu mai bine de trei ani în urmă încă alternative la marele înaintaș, în noile versiuni drumurile

lor se despart. Foxpro urmează și pe mai departe modelul dBase și s-a dezvoltat într-un sistem atractiv care poate servi ca exemplu și altor producători. Compilatorul integrat mai este încă o opțiune suplimentară a produsului program, dar nu mai este neapărat argumentul principal care să determine cumpărarea lui.

Cu totul altfel decît producătorii lui Foxpro, programatorii de la Clipper încearcă să se îndepărteze de modelul dBase și să-și dezvolte programul ca pe un limbaj de programare pentru aplicații cu baze de date de sine stătător, chiar dacă și pe mai departe se bazează pe sintaxa dBase.

Clarion nu este înrudit cu dBase. Aceasta înseamnă că atât în ceea ce privește limbajul cât și structura datelor urmează reguli proprii. Clarion a fost conceput de la început ca un sistem de dezvoltare de sine stătător și urmărește cu consecvență să facă viața programatorului cât mai simplă cu putință.

Clipper 5.0

La obîrșie Clipper a fost o unealtă utilizabilă pentru compilarea programelor dBase pentru a se obține din ele fișiere executabile (.EXE), care să poată fi rulate apoi și în absența dBase-ului. Din acest motiv Clipper trebuie să preia cîteva "necurățenii" ale dBase-ului. dBase este un interpretor care se concentrează asupra liniei curente din codul programului care se prelucrează. Dacă este apelat un subprogram, atunci dBase se "uită" mai întîi pe harddisk să vadă dacă acolo există un program cu acest nume. dBase stochează variabilele într-un spațiu de memorie rezervat. Toate acestea contrazic, natural, modul de lucru al unui compilator: rezervarea pașală de memorie pentru variabile este risipă curată

de spațiu. Din acest motiv în cazul compilatoarelor variabilele trebuie declarate exact, pentru a se putea rezerva numai spațiul necesar. Un compilator nu se poate uita pe harddisk abia în momentul rulării pentru a vedea dacă o subrutină există sau nu. El necesită prezența tuturor părților programului în momentul compilării și construiește cu ajutorul lor programul final. În sensul acestei tratări diferite a subprogramelor, Clipper a permis încă de la început construirea mai multor proceduri în interiorul unui fișier .PRG. În afară de aceasta Clipper a cunoscut încă de la început așa-numitele funcții definite de utilizator (User Defined Functions - UDF). Ambele facilități le-a preluat însă și dBase în versiunea IV.

În versiunea 5.0, însă, Clipper merge pe drumul său propriu. Clipper 5.0 se aseamănă foarte mult cu limbajele de programare clasice. Tratarea variabilelor este un capitol important în acest sens. Risipa tocmai amintită de spațiu de memorie, datorită principiului de repartizare a memoriei rezervate, nu a putut fi înlăturată în întregime pentru că s-ar fi pierdut compatibilitatea cu versiunile mai vechi de Clipper. Totuși în Clipper 5.0, cu ajutorul unor directive, pot fi declarate variabile care să nu-și schimbe valoarea pe lungimea întregului program. Compilatorul plantează aceste variabile direct în cod și pentru ele nu trebuie rezervat un loc special în memorie. Același lucru se întîmplă și pentru tipul declarativ "Static": și în acest caz este vorba despre o variabilă care nu-și schimbă valoarea în timpul rulării programului.

Și fără aceasta însă în noua versiune a lui Clipper există diferențe demne de consemnat în modul de tratare a memoriei. Astfel este utilizată o altă formă de transmitere a parametrilor unei proceduri. În sintaxa veche a instrucțiunilor cu

DO și WITH, variabilele erau transmise prin referință, aceasta însemnând că i se transmiteau procedurile adresele din memorie la care se aflau aceste variabile, procedura accesând apoi direct respectivele zone de memorie. În cazul unei erori în program procedurile și funcțiile puteau schimba aceste valori involuntar, deoarece erau accesate variabilele originale. În noua sintaxă, și în cazul funcțiilor, variabilele sînt transmise prin valoare, aceasta însemnând că procedurile și funcțiile primesc o copie a variabilelor cu valoarea lor actuală, și că nu mai au acces la original. Dacă este necesară, totuși, transmiterea prin referință, atunci este suficient să se pună în fața variabilei caracterul "T".

Un nou tip declarativ permite o mai bună supraveghere a variabilelor. Dacă o variabilă este declarată "Private", atunci din acest moment ea este valabilă pe nivelul curent și pe toate subnivelele apelate din acest punct. O declarație Private în programul principal este sinonimă cu o declarație Public, deoarece toate celelalte părți ale programului sînt apelate din acest punct. Prin noul tip "Local", din contră, o variabilă este declarată cunoscută numai în partea de program actuală, ea rămînînd necunoscută în subprogramele apelate. Pe lîngă controlul îmbunătățit, aceasta mai are și avantajul că permite utilizarea aceluiași nume de variabile în fiecare parte a programului, atît timp cît acestea sînt declarate Local.

Noutatea esențială o constituie însă driverele de baze de date interschimbabile. Clipper recunoaște formatele fișierelor create cu dBase III, Paradox sau un Server SQL și lucrează intern cu sintaxa dBase/Clipper.

Foxpro 1.0

Foxpro și-a propus un cu totul alt scop. Proiectanții și-au pus problema: cum ar trebui să arate urmașul lui dBase III Plus, și au dotat programul cu tot ceea ce are nevoie un SGBD modern cu mediu de dezvoltare integrat. Rezultatul obținut este un SGBD care are una dintre cele mai atractive suprafețe utilizator, orientată caracter, din momentul de față, cuplat cu un compilator foarte puternic.

Foxpro a preluat structura de date a Foxbase-ului, care era compatibilă cu dBase și care utiliza exclusiv fișiere indexate proprii. Noile tipuri de cîmpuri numerice "Floating Point" și "Simplă precizie", apărute în dBase IV, au fost preluate. Noile fișiere de index .MDX, apărute în dBase IV, care permit pînă la 48 de indexuri pentru fiecare bază de date, rămîn caracteristice doar dBase-ului. Foxpro permite pînă la 21 de indexuri pentru un fișier, dar în total nu pot fi deschise mai mult de 25 de indexuri simultan. Și Clipper renunță la acest tip practic de fișier index, dar în noua versiune permite cîte 15 fișiere index pentru un domeniu de lucru, la o medie de 250 de domenii de lucru. Foxpro a păstrat chiar și limitarea dBase de maxim 100 de caractere pentru o expresie de index.

* Ca și Clipper, nici Foxpro nu permite comenzi SQL adresate direct, așa cum este permis în dBase IV. Sintaxa se bazează pe Foxbase și cu aceasta pe dBase III, dar a fost extinsă cu multe comenzi folositoare. Sînt acceptate și noile comenzi apărute în dBase IV, dar există și comenzi noi.

	Clipper 5.0	Foxpro 1.0	Clarion 2.1
Producător	Nantucket	Fox Software	STF Software
Preț (DM)	2730	2500	2680
Runtime	1930		
rețea	3415		
Necesar memorie	512 KB	512 KB	512 KB
Fișiere EXE	da	nu	da
Debugger	da	da	da
Ferestre	da	da	da
Operare cu mouse	nu	da	nu
Prelucrare bază de date	nu	da	da
Număr articole/fișier	1 miliard	1 miliard	nelimitat
Număr cîmpuri/articol	1024	255	nelimitat
Tipuri de cîmpuri:			
caracter	da	da	da
numeric	da	real	zecimal
			short, long, real
dată	da	da	nu
timp	nu	nu	nu
valoare	nu	nu	nu
logic	da	da	da
selecție	nu	nu	da
memo	da	da	da
Număr fișiere deschise			
simultan	250	10	252
Indexuri / fișier	15	10	252
Compatibilitate			
comenzi dBase	dBase III	da	nu
fișiere dBase	da	da	anunțată
versiune Unix	nu	da	anunțată
versiune Macintosh	nu	da	anunțată

Suprafața atractivă este un mediu de programare confortabil. Editorul este integrat, se poate opera cu ajutorul mouse-ului și pot fi prelucrate mai multe fișiere simultan. Cu ajutorul funcțiilor Cut și Paste pot fi copiate informații din toate ferestrele simultan deschise.

Programul dispune și de un depanator (Debugger) și de un program de trasare (Trace Programm), integrate în mediul de dezvoltare, și de programe ajutoare cum ar fi generatorul de machete și de programe, și chiar de un program ajutător pentru documentarea automată a programelor.

Și în cazul compilatorului Foxpro își urmează drumul propriu. În loc să elaboreze cod executabil complet (.EXE), se obține un format intermediar care la rulare necesită existența lui Foxpro sau a unui modul Run-Time. Acesta nu este neapărat un dezavantaj, deoarece Foxpro este mai rapid cu 20% decât vechiul Clipper. Proiectantul însă trebuie să-și bage mâna adânc în buzunar pentru a plăti și modulul Run-Time. Această investiție se face însă o singură dată, modulul putând fi dat mai departe fără restricții.

Spre deosebire de Clipper, Foxpro nu se limitează doar la lumea DOS, existând versiuni și pentru Macintosh și Unix.

Clarion 2.1

Clarion își primește utilizatorii cu brațele deschise. În meniul introductiv sînt afișate toate instrumentele: instrumente de programare, mediu de programare și partea de

ază de date. Instrumentelor de programare le aparțin editorul, generatorul de programe, precum și un program pentru generarea unor machete cu mesaje de ajutor.

În mediul de programare, programele elaborate pot fi rulate atît timp cît n-au fost transformate încă în fișiere .EXE cu un linkeditor extern. Clarion cunoaște două medii: cu și fără debugger (depanator). Depanatorul este folosit numai în timpul dezvoltării, programele definite sînt rulate apoi fără acesta.

În partea de bază de date sînt create și modificate structurile fișierelor, sînt puse la dispoziție comenzi pentru prelucrarea directă a datelor, dar mai ales aici se găsesc funcțiile de import și export. Pe lîngă formatul propriu, Clarion mai poate citi și scrie și fișiere dBase, DIF și Basic. În acest fel este posibil un schimb de date cu alte programe fără probleme.

Componenta Designer este un amestec între un program de asistență în proiectare și un generator de programe. Nici Clipper și nici Foxpro nu posedă ceva asemănător.

În principal există trei tipuri de subprograme pentru aplicațiile cu baze de date: meniuri, machete și rapoarte. Clarion a standardizat aceste tipuri și le-a înlocuit cu alte două proceduri standard: tabele și altele. Tabelele sînt tratate ca o unitate și nu ca înregistrări individuale. Sub titulatura de "altele" se ascunde o interfață spre toate subprogramele care nu au fost elaborate cu Designer. Acestea pot fi rutine în limbajul Clarion scrise de utilizator, dar și module scrise în alte

limbaje de programare, sau module create cu Designer dar în care s-a intervenit cu editorul și din acest motiv nu mai pot fi prelucrate în continuare cu Designer.

O caracteristică tipică Clarion este predefinirea tastaturii. La cîmpurile numerice, la dorință, este activă automat tasta Num-Lock, la cîmpurile de text poate fi prestabilit modul inserare sau suprascriere. Această caracteristică foarte folositoare conduce însă la o folosire diferită a tastaturii în cadrul fiecărui cîmp. Cine parcurge cu cursorul cîmpurile și ajunge într-un cîmp numeric, se vede dintr-o dată confruntat cu un șir de 2 -uri.

Sistemul Clarion a fost dezvoltat ca limbaj și nu ca program de ajutor în prelucrarea datelor. Rezultatul este un limbaj, care conține și elemente de C (de ex. fișiere Include), foarte curat construit și ușor de învățat.

Facilitatea de lucru în rețea este elegant rezolvată. La generarea codului cu ajutorul instrucțiunilor lui Designer este dat un model de prezentare. Dacă în locul modelului standard se dă modelul rețelei, atunci codul obținut poate fi rulat în rețea.

Cele mai noi concepte, cum ar fi suprafață utilizator grafică, arhitectură client-server, sau programarea orientată pe obiecte, nu sînt implementate. Sînt anunțate versiuni pentru OS/2, Unix și Macintosh, dar încă nu sînt disponibile.

*(Computer Persönlich 26/90,
Robert Bäuerle)*

Aplicații distribuite

(continuare din pag. 43)

În practică trebuie să se mai țină cont și de multe alte aspecte suplimentare:

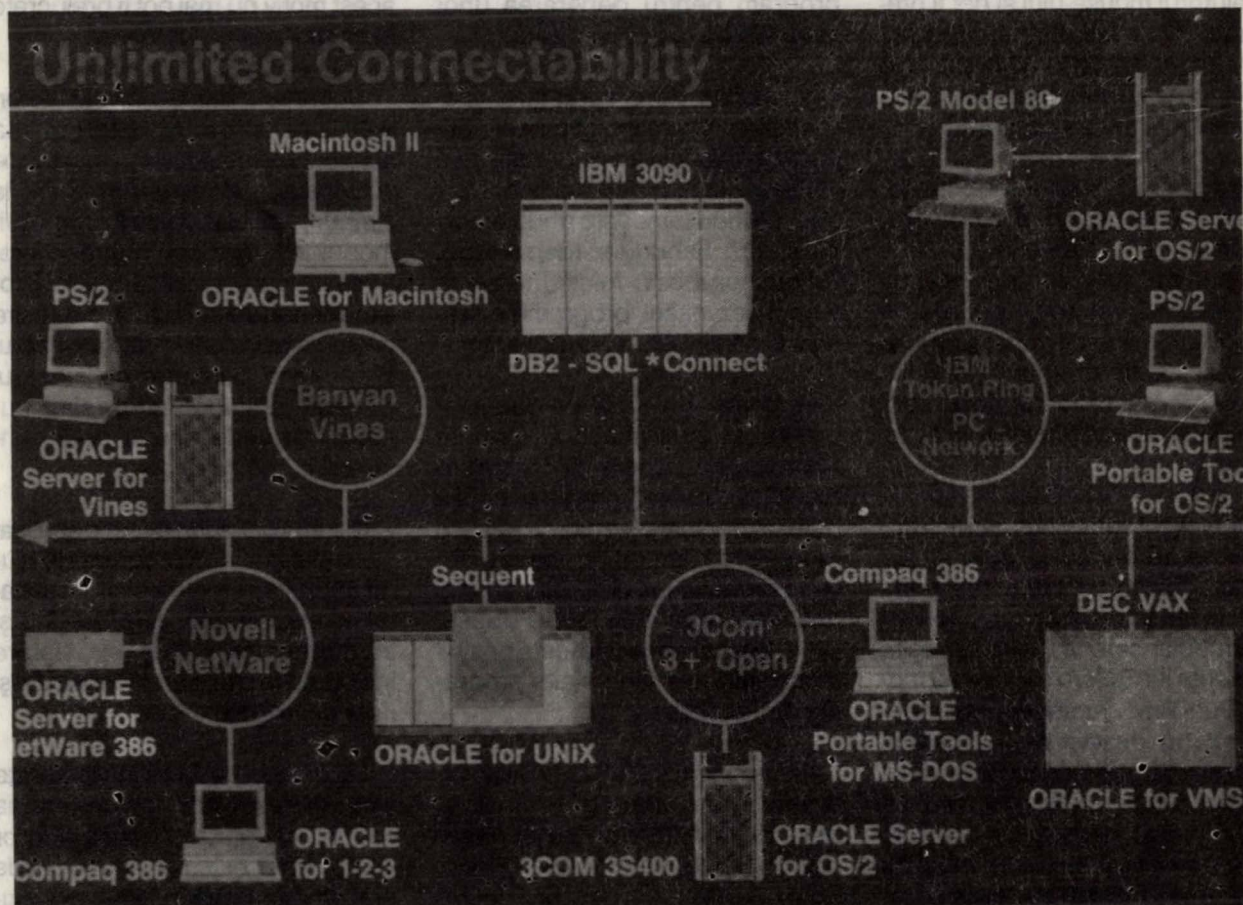
- faze de test detaliate
- instalare
- organizarea fluxului de lucru
- întreținere și - suport.

Prin intermediul aplicațiilor distribuite programatorii pătrund în regiuni necunoscute pînă acum.

Premisa esențială este înțelegerea și acomodarea cu noile instrumente și tehnici de lucru.

*(Computer Persönlich 22/90,
Steve Guengerich, Dwight Williams)*

Aplicații distribuite



Aplicațiile distribuite utilizează rețelele în mod optim

Fundamentele prelucrării distribuite a datelor se pun în momentul conceperii design-ului. Vă oferim o privire de ansamblu asupra stării actuale a tehnicilor specifice și vă prezentăm câteva instrumente și utilitare destinate dezvoltării de aplicații cu baze de date distribuite.

Tendența de a dezvolta aplicații distribuite este determinată în principal de dorința de a se exploata puterea de calcul a noilor PC-uri aflate pe birourile întreprinderilor private și ale administrației publice. Aplicațiile distribuite oferă în aceste cazuri avantaje importante: datorită păstrării distribuite a datelor în sistemul informatic al întreprinderii multe operații pot fi executate în paralel. Scriind aplicațiile distribuite în așa fel încât datele să se afle

fizic acolo unde este cel mai des nevoie de ele, scad atât cheltuielile pentru comunicații cât și costurile datorate lor într-o întreprindere.

Proiectarea și dezvoltarea aplicațiilor distribuite sînt probleme foarte complexe. Dar în aceeași măsură în care cresc cheltuielile, crește și numărul producătorilor care pun la dispoziție instrumente pentru dezvoltare. Cei mai cunoscuți producători de sisteme de baze de date distribuite sînt: "Empress", "Informix", "Ingres", "Oracle", "Sybase". Și IBM oferă prin specificația APPC (Advanced Program to Program Communications) standarde pentru protocoale ale unei prelucrări distribuite și rutinele necesare pentru a putea exploata protocoalele respective.

Acestea, pentru a simplifica dezvoltarea aplicațiilor distribuite, sînt integrate în sistemul de operare OS/2 Extended Edition, al lui IBM.

Din punct de vedere al modelului există trei forme de prelucrare a datelor: centralizat, descentralizat și distribuit. Sistemele pur centralizate sînt folosite pentru aplicațiile întreprinderilor mari, care se prelucrează pe mainframe-uri (calculatoare mari). Datele și programele, în majoritatea cazurilor se află în același calculator. Sistemele descentralizate se pretează pentru aplicații care se rulează pe calculatoare distincte, fără să existe un schimb de date între o aplicație și alta. Pe fiecare calculator se găsesc aceleași programe și aceleași structuri de date, nu însă și

aceleași date. Sistemele cu un singur utilizator (single-user) au apărut odată cu PC-urile. Cu toate că tendința de a utiliza rețele este tot mai mare, în continuare cele mai multe programe se scriu pentru sisteme single-user, mai ales pentru micii meseriași și pentru întreprinderile mici din sfera serviciilor.

Sistemele distribuite sînt un amestec de aplicații centralizate și descentralizate. Ele înglobează avantajele celor două forme anterioare. În cazul lor programele și datele se pot găsi în orice loc: centralizat pe un mainframe, pe calculatorul dintr-o secție, pe fileserver-ul unei grupe de lucru, sau pe PC-ul unui specialist.

Sistemele distribuite sînt cele mai apropiate de atingerea scopului - realizarea unui echilibru între o viteză mare de prelucrare a datelor și realizarea unui control permanent asupra consistenței datelor. Instanțele de control sub formă de backup, restart, recovery și management al rețelei sînt centralizate în cazul lor. Viteza mare de lucru se atinge, în acest caz, datorită faptului că programele și datele se află acolo unde sînt necesare cel mai des. Aceasta garantează timpi de răspuns scurți la aplicațiile zilnice.

Dezavantajul sistemelor de baze de date distribuite îl constituie costul mare al echipamentelor hard, deoarece aplicațiile distribuite au nevoie de inteligență proprie și de o memorie externă relativ mare. Și costurile pentru soft pot fi mari, dacă aplicațiile și suprafețele utilizator necesită adaptarea driverelor, conversii și comunicații de date. La planificarea unei aplicații distribuite este hotărîtoare numirea unui angajat propriu care să răspundă de produsul final. Acesta trebuie să fie cineva care să supravegheze activitățile efectuate cu datele. El trebuie să aibă o privire de ansamblu asupra drepturilor de acces la date și este responsabil pentru operațiile de salvare (backup) și restaurare (recovery) în caz

de necesitate. În domeniul lui de activitate intră și administrarea structurii bazei de date și măsurile de acordare cu DBMS-ul (Data Basis Management System) cu care a fost dezvoltată aplicația. Această persoană, numită și administratorul bazei de date (DBA - Data Basis Administrator), joacă un rol decisiv atunci cînd distribuirea prelucrărilor se vrea să fie un succes. Ideea de a crea o instanță independentă de administrare și control este străină celor mai mulți programatori, dar elaborarea unor aplicații distribuite este inefficientă fără un DBA. Dar pentru că multe drumuri duc la Roma, punctele decisive în cazul unei aplicații distribuite sînt designul și posibilitatea de restructurare. Din acest motiv este hotărîtor ca proiectantul să cunoască exact cerințele de specialitate impuse de o bază de date distribuită. La analiză două puncte stau în fruntea listei de probleme:

- care date să fie stocate local și care global
- cum trebuie legate componentele între ele.

Conceptul de date locale și date globale este fundamental în cazul aplicațiilor distribuite. În această accepțiune, datele locale sînt acele date care sînt necesare într-un singur modul de prelucrare pe calculatorul de la un loc de muncă, pe un file-server sau pe calculatorul unei secții. Datele globale, care de cele mai multe ori sînt decisive pentru rularea corectă a unei aplicații, sînt date care sînt prelucrate de două module aflate pe același nivel ierarhic, ambele fiind legate de treapta de prelucrare superioară. Datele globale pot fi memorate pe un file-server și folosite de două stații de lucru din rețea, sau pot fi memorate pe calculatorul unei secții care deservește două servere de baze de date din rețea. Un exemplu va clarifica diferența dintre datele globale și cele locale și importanța cunoașterii, încă din faza de analiză, a punctelor de vedere locale și globale.

Să presupunem că o agenție de aviație își planifică un sistem contabil, care să poată fi utilizat gratuit de către agențiile independente de turism pentru rezervarea de locuri și că în plus ea dorește să-și dezvolte un sistem de evidență a personalului și de calculul salariilor. La analiză se va ajunge la concluzia că majoritatea datelor personale ale angajaților din birourile de turism vor trebui stocate și ținute la îndemîna local. Singurele date personale care trebuie stocate global sînt numerele de înmatriculare ale angajaților. În schimb datele personale ale angajaților agenției aviatice sînt date globale, aceasta facilitînd mai ales planificarea schimburilor.

Dacă patronul unui birou de turism dorește să cunoască activitatea unuia din angajații săi, atunci prin intermediul "numărului de marcă" al angajatului își poate construi o legătură la mainframe și poate obține statistica cerută. Pentru că toate celelalte date, cum ar fi: numele angajatului, data nașterii, numărul cărții de asigurări sociale, etc., sînt necesare într-un singur loc, și anume la birou, acestea pot fi stocate local, indiferent în ce format. Aceasta înseamnă că formatul local al datei de naștere poate fi "LLAAZZ" într-un birou de turism, iar într-altul "AAAZZLL".

După cum se poate vedea, din acest exemplu simplu, este necesară o analiză preliminară detaliată înainte de a putea spune despre o dată că este locală sau globală. Dacă majoritatea datelor sînt declarate locale, atunci trebuie să existe mai multe programe "locale" pentru administrarea acestor date și pentru a putea obține rapoarte despre ele, în același timp însă vor crește cheltuielile de dezvoltare și de întreținere a sistemului. În exemplul dat anterior calculul salariilor se face în birourile locale de turism. Necesară este doar stabilirea unui interval de numere între care să se dea numerele de înmatriculare personale pentru fiecare birou de turism.

Aspectele analizei datelor locale și globale poate depăși limitele organizatorice. De exemplu într-un sistem de rezervare și control al biletelor de avion pot exista și agenții cu birouri în toată lumea. Fiecare agent ar putea, în acest caz, să hotărască liber, deci local, asupra unui segment al contingenței de rezervări. Modulele care se referă la controlul zborurilor ar trebui să ofere informații continue, actualizate, despre fiecare aeroport în parte, deci trebuie să aibă acces la date globale.

melor și de multitudinea cerințelor individuale pe care trebuie să le satisfacă programul. Rezultatul este că vor trebui dezvoltate multe bucați mici de programe dificile.

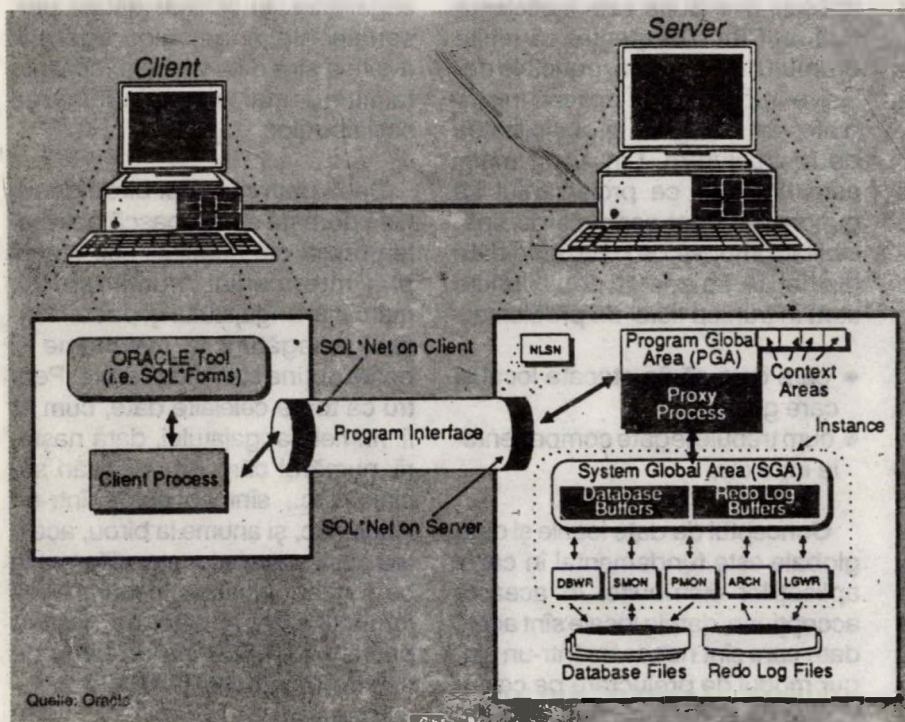
De multe ori diferențele dintre programe constau doar în faptul că introducerea datelor pentru cîmpuri identice se face din poziții de pe ecran diferite. Mulți programatori cu experiență sînt tentați să ocolească aceste diferențe și să încerce să dezvolte module multifuncționale. Aceasta poate duce la

pînă la integrarea programelor într-o aplicație completă.

Cuplarea diferitelor componente într-o aplicație se poate face în variante treptate. La o aplicație cu legături slabe fiecare componentă este un atom, și numai funcțiile care comandă acea componentă sînt legate cu aplicația. O aplicație cu legături strîns se caracterizează prin aceea că diferitele componente sînt grupate în mod logic și funcțiile care acționează asupra componentelor sînt asociate cu grupele care le supraordonează. Această metodă de design reduce puternic flexibilitatea și capacitatea de întreținere a unei aplicații. Dacă de exemplu funcțiunea de rezervare de locuri ar fi strîns legată cu orarul de control al zborurilor, atunci programele aplicației ar fi prea strîns legate între ele. Cu toate că această combinație ar reduce necesitatea proiectării și codificării diferite a măștilor de ecran, aplicația ar putea funcționa doar atunci cînd derularea procesului din ambele domenii ar fi absolut identică tastă cu tastă.

Dacă cerințele asupra unei rezervări s-ar schimba peste noapte, ar trebui verificate și re-proiectate părți ale aplicației, problemă care se ivește întotdeauna cînd componentele aplicației sînt prea strîns legate între ele. În plus exemplul aproape că depășește deja imaginația unui programator prevăzător. Dar practica este de regulă încă mult mai complexă decît teoria.

Alte puncte de vedere de care trebuie să ținem cont încă din faza de analiză sînt competența operațională și funcțională ale unui utilizator. Ne referim aici la dreptul de acces la anumite date, la stabilirea grupei de utilizatori, felul în care se va face manipularea datelor permise unei anumite persoane, etc. Este foarte important ca aceste lucruri să fie stabilite din timp.



Arhitectură multi-proces în cazul unui server Oracle

Cel de-al doilea criteriu important, de aceeași importanță ca și definirea datelor locale și globale, este stabilirea modului de interconectare a componentelor. Cu toate că accentul în acest caz cade pe programe și nu pe date, și în acest caz cel mai bun coordonator este administratorul bazei de date, pentru că el răspunde, printre altele, de design-ul bazei de date. Cerințele unei aplicații distribuite seamănă mai mult cu cele ale unui magazin universal, decît cu cerințele tradiționale ale unei aplicații, deoarece trebuie să se țină seama de puterea diferită de calcul a siste-

obținerea unor module atît de inflexibile încît ele nu vor mai putea fi adaptate la situații noi și vor trebui rescrise. Scopul dezvoltării unei aplicații distribuite este de a reduce fiecare cerință la componentele ei elementare. Fiecare componentă trebuie să conțină numai acele instrucțiuni de care are nevoie în timpul lucrului.

Această tehnică de atomizare, care este de fapt o analiză top-down, s-a răspîndit la toate nivelele programării, începînd de la scrierea de rutine pentru module, construirea programelor din module și

Pe lângă aceste aspecte ale protecției datelor, de mare importanță este și verificarea corectitudinii datelor încărcate. Aici nu ne referim doar la încadrarea valorilor câmpurilor în anumite limite. Multe aplicații sînt prevăzute cu câmpuri de text vide (zone de rezervă). Așa se poate întâmpla ca după un timp în aceste câmpuri să se adune informații total diferite. Astfel într-un birou câmpul de rezervă poate conține stadiul datoriilor din ultima seară de popice, în timp ce în alt birou același câmp poate conține date despre locul de parcare al angajatului în parcare a firmei.

Diferența hotărîtoare între designul unei aplicații tradiționale și cel al unei aplicații distribuite este gradul mult mai ridicat de complexitate. Într-un sistem tradițional programele și datele se află de regulă într-un singur loc și acolo sînt și folosite. Un sistem distribuit se face remarcat prin aceea că programele și datele sînt memorate în locuri diferite și rolul programelor poate varia în funcție de aplicația care tocmai se află în execuție.

Cu gradul de complexitate crește și capacitatea. Dacă aspectele de: integritate a datelor, controlul accesului simultan la date, tratarea erorilor, operațiile de backup și recovery sînt tratate ordonat, atunci o aplicație distribuită poate aduce avantaje semnificative în comparație cu un sistem tradițional. O condiție este, ca de exemplu, accesul la datele globale să fie cît mai redus. Dacă se ține cont de această indicație, atunci utilizatorul poate lucra cu datele sale locale și atunci cînd serverul sau calculatorul mare nu este operațional. Acesta este un avantaj mare față de sistemele centralizate, pentru că la acestea dacă "pică" server-ul, toate lucrările stau.

Aspectele centrale ale unei aplicații distribuite sînt integritatea datelor și controlul accesului simultan asupra datelor. Există o serie de tehnici, pe care programatorul le

poate folosi, pentru a proteja datele înainte, în timpul sau după prelucrarea propriu-zisă. Integritatea datelor cuprinde: exactitatea, valabilitatea și protecția datelor. Într-o aplicație distribuită integritatea datelor este asigurată de către programe. Într-un sistem de rezervare de locuri la avion, agenții din mai multe orașe se ocupă de rezervarea locurilor pentru aceeași cursă. Pentru a asigura integritatea datelor trebuie găsită o metodă care să aibă grijă ca același loc să nu fie rezervat decît o singură dată. Din motive de siguranță, la anumite date, unii utilizatori trebuie să aibă acces numai în citire. Modificări asupra datelor se vor permite numai anumitor grupe de utilizatori. Controlul acceselor simultane la date se ocupă cu modul în care aplicația tratează cererile de interogare. Problemele serioase puse de accesul simultan la date pot fi depistate numai cu tehnici de proiectare încercate. Un domeniu de probleme este ridicat de succesivitatea în care resursele aplicației, cum ar fi: timp CPU, loc în memorie sau pe harddisk, imprimantă, sau chiar înregistrările individuale ale unei tabeli, pot fi solicitate.

În timpul folosirii acestor resurse de lucru se poate ajunge la un "deadlock" - un blocaj total, reciproc. Un astfel de blocaj poate să apară, de exemplu, în cazul în care un program așteaptă eliberarea unei resurse folosite de un alt program, iar acest al doilea program, așteaptă o resursă acordată primului program.

O altă problemă este cunoscută sub numele de "lock out" - închis pe dinafară. Această situație poate să apară atunci cînd un program folosește în mod exclusiv o resursă, care este necesară și unui alt program, care își poate continua execuția abia după accesul la această resursă. Prin această excludere nu numai că sistemul va merge mult mai încet, dar sistemul poate și "claca". Dacă un angajat al unui birou de turism își propune o rezervare, dar lasă tranzacția neterminată și merge să ia masa de prînz, atunci ne putem închipui că tranzația blochează resurse critice și că în acest timp nu se pot face alte rezervări de locuri.

Există tabele care oferă protecție împotriva fenomenelor de "deadlock" și "lock out", ele acordînd programelor un timp maxim de prelucrare pentru o anumită resursă. Dacă acest timp este depășit și dacă nu urmează o altă cerere semnificativă, resursa este eliberată. Ca exemplu dăm situația rezervării de loc neterminat. Prin folosirea unei tabeli de alocare a resurselor, la început tranzacția va fi întreruptă. În pasul următor trebuie să ne îngrijim ca starea anterioară să fie refăcută, iar la urmă trebuie să redăm folosirii generale toate resursele care au fost blocate prin tranzacție.

Design-ului îi urmează programarea. Există mai multe modalități de a dezvolta o aplicație, variantele diferențiindu-se în funcție de mediul de dezvoltare și instrumentele alese. Două probleme fundamentale, care joacă momentan un rol important în discuțiile despre bazele de date distribuite sînt: arhitectura client-server și bazele de date bazate pe SQL. În cadrul dezvoltării aplicațiilor distribuite modelul client-server prezintă cel mai mare interes. În acest caz programele se află pe calculatorul "client" (de obicei un PC), legat într-o rețea LAN cu "server"-ul bazei de date. Server-ul prelucrează interogările clientului, pregătește rezultatele și răspunde cererii.

Există trei modele de bază: - clienți inteligenți și serveri proști - clienți proști și serveri inteligenți - clienți inteligenți și serveri inteligenți

În cazul clienților inteligenți și a serverilor proști, pregătirea datelor și interogările au loc în întregime pe calculatorul client, server-ul fiind folosit doar pentru introducerea și extragerea datelor. Această variantă este foarte răspîndită la mul-

te sisteme de management a bazelor de date de tipul "Low-End-PC". Terminalele proaste și server-ele inteligente acoperă celălalt capăt al ierarhiei; această constelație domină în cazul sistemelor centralizate cu mainframe-uri sau minicalculatoare. În cazul în care atât server-ele cât și clienții sînt inteligenți, datele sînt prelucrate cu o integritate îmbunătățită și resursele rețelei sînt exploatate mai eficient.

Avantajele ultimului model numit pot fi exemplificate printr-o interogare simplă ad-hoc. Un angajat al unei firme de turism dorește să confirme cererea de rezervare a unui loc pentru domnul Leon Moraru și introduce numele "Moraru" în câmpul "Nume" din macheta de pe ecran, ca fiind singurul criteriu de căutare.

În cazul unui client inteligent și a unui server prost, server-ul trebuie să-i trimită clientului întregul fișier de rezervări, iar acesta trebuie să caute singur înregistrarea respectivă. În varianta cu un client prost și un server inteligent, server-ul ar trimite clientului toate înregistrările care conțin valoarea "Moraru" în câmpul "Nume". Numele "Moraru" fiind destul de frecvent, pe ecran este posibil să apară destul de multe înregistrări. Interogarea ar trebui detaliată specificid, de exemplu, și prenumele domnului "Moraru". În acest caz însă înregistrările transmise deja clientului devin fără valoare și căutarea trebuie reluată de la început.

În cazul clienților inteligenți și a server-elor inteligente, înregistrările deja selectate nu trebuie "aruncate", deoarece atât clientul cât și server-ul știu că ele sînt deja afișate la client. La o detaliere a interogării, clientul va produce un subsșir de înregistrări, din șirul obținut de la prima interogare, subsșir care va conține valoarea "Leon" în câmpul "Prenume".

O altă tendință în dezvoltarea aplicațiilor cu baze de date este

folosirea SQL-ului pentru accesul la baza de date, pentru interogarea bazei de date și pentru manipularea datelor. SQL este un limbaj ne-procedural, dezvoltat pentru bazele de date relaționale, care își are rădăcinile în teoria mulțimilor din matematică. SQL a fost obținut în urma unui proiect de cercetare al IBM-ului. Limbajul constă din trei componente esențiale: limbajul de definire a datelor (DDL, Data - Definition-Language), limbajul de control a datelor (DCL, Data - Control-Language) și limbajul de manipulare a datelor (DML, Data-Manipulation-Language).

DDL oferă mecanisme pentru crearea și modificarea structurilor bazei de date. DCL se ocupă de supravegherea valabilității datelor dintr-un mediu multi-user. Componenta cea mai utilizată este DML, componentă care oferă funcțiunile de interogare, adăugare și modificare de date.

Deoarece SQL este recunoscut în lumea calculatoarelor ca un standard general, el a devenit fundamentul multor aplicații cu baze de date distribuite. Cu ajutorul lui programele pot utiliza în comun datele aflate în rețele diferite și sub sisteme de operare de rețea diferite. Producătorii de soft au implementat limbajul SQL într-o multitudine de produse, începînd cu mediul de programare 4GL, cu limbajele de programare și pînă și în pachetele de soft orientate spre utilizator cum ar fi Lotus 1-2-3. Produsele bazate pe SQL au devenit un element de bază pentru dezvoltarea aplicațiilor distribuite, chiar dacă SQL presupune echipamente hard de înaltă calitate, și viteza de lucru pe un PC mai lasă încă de dorit. Există o multitudine de instrumente pentru dezvoltarea aplicațiilor distribuite, baze de date distribuite, servere de baze de date, medii de dezvoltare și alte utilitare. O bază de date distribuită poate funcționa concomitent pe platforme hard diferite și pe sisteme diferite dacă s-a realizat comu-

nicația și compatibilitatea între platforme.

O bază de date este susținută, de regulă, de SQL, de un mediu de dezvoltare 4GL și de mai multe limbaje de programare. În continuare vor fi prezentate pe scurt două exemple de baze de date distribuite: "Oracle" și "Informix". Alte nume de baze de date care ar putea fi enumerate în acest context ar fi: "Empress", "Ingres", "Sybase" sau "Zim".

"Oracle" a oferit prima implementare completă a lui SQL într-o bază de date distribuită deja în 1979. Pentru susținerea limbajelor de programare C, Cobol sau Fortran ca și Front-Ends (comunicație cu exteriorul bazei), Oracle oferă în plus instrumente Front-Ends proprii:

- SQL Forms, un instrument de dezvoltare interactiv
- SQL Plus, un instrument interactiv pentru cereri și liste
- SQL Menu, un instrument pentru proiectarea interactivă a meniurilor
- SQL Star, un instrument pentru obținerea de cod independent de protocol
- SQL Report Writer, un instrument interactiv pentru liste, și
- SQL Graph, un instrument interactiv pentru grafică.

Informix oferă facilități pentru prelucrarea datelor binare, a "regulilor" care trebuie folosite în cadrul înregistrărilor selectate ale unei baze de date și mecanisme pentru ridicarea eficienței operațiilor I/O. Informix se sprijină pe mai multe instrumente de dezvoltare pentru PC-uri, pe o interfață pentru limbajul de programare C, pe pachetul propriu de programe integrate Smart Ware și pe un Add-in Lotus 1-2-3 pentru interogarea datelor.

Server-ele bazelor de date, o altă categorie de instrumente pentru dezvoltarea de aplicații distribuite, se bazează de regulă pe modelul client-server. Cele mai

multe servere de baze de date constau din două componente: partea de "Back-End", care se află pe serverul rețelei și răspunde de integritatea datelor și expedierea răspunsurilor, și partea de "Front-End", care se află pe PC-ul utilizatorului, căruia îi oferă posibilitatea să interogheze baza de date și să reprezinte rezultatele obținute în diferite formate. Se pot deosebi două categorii de servere de baze de date: servere de baze de date pe bază PC și servere de baze de date distribuite. Prima categorie se limitează la sistemele care au fost legate în rețele cu sistemul de operare MS-DOS sau OS/2, în timp ce a doua categorie poate lucra atât cu sisteme de operare de rețea, cât și cu sisteme de operare multiutilizator, cum ar fi Unix sau MVS.

Majoritatea serverelor de baze de date aparțin categoriei de servere bazate pe PC-uri. Așa sînt, de exemplu, produsele "SQL Base" sau "SQL-Server". Integrarea instrumentelor "Front-End" se face prin intermediul interfeței de programare "DB-Library" a server-ului SQL. SQL Base dispune de un mediu de proiectare grafic, care se bazează pe Microsoft Windows și pe Presentation Manager. Facilitățile grafice pot fi utilizate pentru proiectarea de aplicații, care să poată accesa baze de date SQL Base.

Din categoria server-elor de baze de date distribuite pot fi enumerate "Netware SQL" și server-ul "Oracle". Netware SQL oferă facilități pentru servere de baze de date pe platforme DOS, OS/2, Macintosh și VAX/VMS. Novell se concentrează pe dezvoltarea de suporturi pentru partea de "Back-End". Partea "Front-End" este cedată caselor independente de soft. Oracle a ales o altă cale și oferă singură toate instrumentele necesare dezvoltării unei aplicații. Astfel Oracle poate oferi un produs a cărui suprafață utilizator este identică pe toate platformele, de la DOS, Unix, DEC VMS și pînă la MVS al IBM-ului.

O altă categorie de instrumente de dezvoltare o constituie, așa numit-a, "Developers Workbench", asemănătoare cu bancul de lucru al unui tâmplar. La fel cum bancul de lucru al acestuia conține toate sculele de care ai nevoie pentru construirea unei piese de mobilier, și Workbench conține toate instrumentele necesare pentru construirea unei aplicații distribuite. Ofertanți de astfel de colecții de instrumente sînt Novell, Hewlett-Packard și Santa Cruz Operation.

Workbench-ul pentru programatori al lui Novell, care conține un kit de dezvoltare software, este un set de instrumente pentru scrierea de module încărcabile în rețea (Netware Loadable Modules, NLM). Un proiectant poate scrie procese client-server, în forma NLM, care să poată fi rulate sub sistemul de operare de rețea Netware 386. Amintim și setul de instrumente "Remote Procedure Call" al firmei Netwise. El conține funcții care se ocupă de comunicarea interproces între programe care sînt rulate pe calculatoare diferite. Printre protocoalele de transport suportate de RPC se numără și TCP/IP, protocolul OSI și DECnet. RPC poate lucra pe sistemele de operare de rețea Netware, LAN Manager, și altele. Cu instrumentele RPC, proiectantul trebuie să stabilească, mai întîi, care aplicații vor fi executate "de la distanță" și care "local". Urmează apoi o dezvoltare a programului ca o aplicație de sine stătătoare, dar construită modular, pentru a se simplifica procesul de programare. Codul programului va fi scris într-un limbaj standard, cum ar fi C, folosindu-se însă extensiile de limbaj oferite de RPC. În fine, aplicația va fi compilată și link-editată. În execuție RPC interpretează extensiile de limbaj, folosind instrumentul ca pe un fel de preprocesor al aplicației care generează cod standard în C.

Un alt workbench pentru dezvoltarea de aplicații distribuite este NCS (Network Computing System) al firmei Hewlett Packard. El conține un compilator, un "Loca-

tion Broker" și funcții pentru dezvoltarea de RPC. Sînt susținute limbajele de programare C și Pascal, sub sistemele de operare HP/UX, VAX/VMS, Ultrix și Sun OS. Produsul "Open Desktop" al firmei SCO este un software care conține sistemul de operare SCO Unix, suprafața utilizator X-Windows, baza de date Ingres și alte componente de dezvoltare. În sfîrșit, la rubrica unelte de dezvoltare există și grupa de "diverse". În aceasta intră, de exemplu, produse ca "Network File System" (NFS) al firmei Sun, un sistem de gestiune a fișierelor distribuite, care permite accesul la date aflate pe mai multe platforme de sisteme de operare. NFS susține mai multe tipuri de echipamente hard, sisteme soft, medii de dezvoltare, și în fine limbajele de programare și compilatoarele existente pe acestea.

Odată ce aplicațiile distribuite vor fi acceptate în întreprinderi, producătorii vor oferi și mai multe instrumente de dezvoltare a aplicațiilor. În acest context ne putem imagina deja un utilitar care să îmbunătățească comunicarea între procese. Proiectanții așteaptă deja o ofertă mai mare de rutine și biblioteci de programe pentru comunicația low-level.

Tendența de a dezvolta aplicații distribuite influențează și modul de organizare al secțiilor MIS. Multe întreprinderi și-au creat deja noi centre informatice, și noi locuri de muncă, care să fie răspunzătoare pentru realizarea de aplicații distribuite. În sarcina lor intră și instruirea personalului care va intra în contact cu astfel de aplicații. Dacă vrem să stăpînim problemele legate de protecția și integritatea datelor, de accesul simultan la date, atunci crearea de astfel de echipe de lucru este o necesitate. Acest articol nu face decît să arunce o scurtă privire în aspectele centrale ale aplicațiilor distribuite.

(Computer Personal 22/90,
Steve Guengerich, Dwight Williams)

(continuare în pag. 37)

Se poate formata și altfel...

Pe un AT cu o unitate floppy de tip MF, utilizatorul nu poate produce pe dischete simple (2S,DD) dischete de 420 kbyte, 720 kbyte sau 800 kbyte. O alternativă o oferă programul "mformat".

Este o dilemă chinuitoare: să te mulțumești cu dischetele simple și ieftine, însă cu capacitatea rizibilă de 360 kbyte sau să utilizezi dischetele încă destul de scumpe de mare densitate (HD) cu capacitatea de 1,2 Mbyte. Căci formatele suplimentare posibile nu sînt sprijinite de MS-DOS nici pentru unitățile floppy MF de 5,25", nici pentru unitățile noi de 3,5". Aceste formate suplimentare sînt: simplă densitate, 42 piste, 10 sectoare/pistă = 420 kbyte; 80 piste, 9 sectoare/pistă = 720 kbyte și 80 piste, 10 sectoare/pistă = 800 kbyte.

Pentru a formata însă o dischetă cu 720 kbyte sau 800 kbyte trebuiesc îndeplinite două condiții

- 1) ROM-BIOS-ul trebuie să fie de versiune 3.xx sau mai mare (ca să cunoască și noile unități de 3,5") și
- 2) MS-/PC-DOS-ul trebuie să fie cel puțin versiunea 3.2

Numai atunci programul care urmează poate aduce fără limitări dischetele la formatul dorit.

Structura informațiilor de pe o dischetă este următoarea:

- bootsector
- primul FAT (tabela de alocare a fișierelor)
- una sau mai multe copii FAT
- directorul principal sau rădăcină (root)
- zona de date

Versiunile mai vechi de DOS (înainte de 3.2) au folosit pentru recunoașterea formatului unei dischete primul octet din primul FAT, așa-numitul media-byte.

Pe baza acestui media-byte DOS și ROM-BIOS puteau recunoaște tipul dischetei.

După cum se poate vedea din tabelă, media-byte-ul nu mai era suficient după apariția unităților de 3,5", deoarece o aceeași valoare a media-byte-ului descrie formate total distincte pe dischete de 3,5" și pe dischete de 5,25". Trebuia găsită deci o nouă soluție pentru deosebirea formatului.

Versiunile DOS de la 3.2 încoace ne oferă o cu totul altă modalitate de recunoaștere a formatului.

Și anume ele se pot acorda singure pe formatul dischetei cu care se lucrează, exploataînd informațiile din blocul parametrilor discului din cadrul boot-sectorului dischetei DOS.

Acest bloc al parametrilor discului se află în boot-sectorul dischetei, deci în primul sector fizic.

Boot-sectorul dischetei

Adresa	Conținut	Tip
+00h	instrucțiune de salt la rutina de boot	3 byte
+03h	numele producătorului și număr de versiune	8 byte
+0bh	bytes/sector	1 word
+0dh	sectoare/cluster	1 byte
+0eh	numărul sectoarelor rezervate (sectoare înaintea primului FAT)	1 word
+10h	numărul FAT-urilor	1 byte
+11h	numărul maxim al intrărilor în directorul rădăcină	1 word
+13h	numărul sectoarelor de pe dischetă	1 word
+15h	media-descriptor (ca și primul octet din FAT)	1 byte
+16h	numărul de sectoare per FAT	1 word
+18h	sectoare/pistă	1 word
+1ah	numărul capetelor de scriere/citire	1 word
+1ch	numărul sectoarelor ascunse (distanța pînă la primul sector)	1 word
+1eh-1ffh	rutina de boot	

De-abia această metodă permite recunoașterea diferitelor formate de dischetă.

Formatarea unei dischete la 720 kbyte într-o unitate MF (care în mod normal lucrează cu dischete de capacitate mare) este relativ simplă. Sistemului de operare trebuie numai să i se "explice" că este vorba de o dischetă de 3,5" formatată la 720 kbyte. Deoarece DOS, după cum s-a arătat mai sus, face recunoașterea formatului prin intermediul boot-sectorului, e imediată ideea modificării acestuia în așa fel încît DOS să recunoască discheta ca una de 3,5" formatată la 720 kbyte.

Dacă o dischetă ar fi formatată în simplă densitate cu 80 piste, 9 sectoare/pistă, modificarea boot-sectorului ar fi suficientă ca discheta să fie recunoscută de DOS ca dischetă de 720 kbyte.

Dar cum se poate formata o dischetă în simplă densitate cu 80 piste?

Tabela semnificație media-byte

byte:	tip:	fețe	piste	sect /pistă	capacitate(kbyte)
f0h	3,5"	2	80	18	1440
f8h	harddisk				
f9h	5,25"	2	80	15	1200
	3,5"	2	80	9	720
fah	5,25"	1	80	8	320
	3,5"	1	80	8	320
fbh	5,25"	2	80	8	640
	3,5"	2	80	8	640
fch	5,25"	1	40	9	180
fdh	5,25"	2	40	9	360
feh	5,25"	1	40	8	160
ffh	5,25"	2	40	8	320

ROM-BIOS disponibilizează o rutină cu ajutorul căreia se poate modifica câte o pistă a unei dischete (funcția 5, Int 19).

Dacă se încearcă formatarea dischetelor de simplă densitate cu ajutorul acestei rutine la 80 de piste, capul de citire/scriere al unității va ajunge în poziția limită finală deja de pe la pista 42. Aceasta nu se datorează faptului că unitatea poate formata numai 40 de piste în simplă densitate, ci ROM-BIOS-ului, care la formatarea în simplă densitate face totdeauna un dublu pas la trecerea la pista următoare. Dacă acest lucru ar putea fi împiedicat, nu ar mai exista nici o stavilă în calea formătărilor la 80 de piste.

ROM-BIOS-ul "ține minte" dacă trebuie executat un dublu sau simplu pas la trecerea la pista următoare.

Modificarea octetului de memorie aflat la adresa 0040:0090 la valoarea 54h are ca efect executarea unui pas simplu la trecerea la pista următoare.

Adresarea directă a acestei celule de memorie este de altfel singura porțiune dependentă de hard a acestui program. Nu trebuie ascuns faptul că la alte calculatoare, unde comutarea de la simplu pas la dublu pas se face prin intermediul altei celule de memorie, programul nu va putea funcționa în forma prezentată. Dacă însă este cunoscut octetul cu pricina, modificări minore în program pot duce la reușită.

După formatare nu mai trebuie apoi scrise pe dischetă decât boot-sectorul cu blocul parametrilor disc, sectoarele FAT și sectoarele directorului rădăcină eventual cu un nume pentru volum. Dacă discheta urmează să fie folosită ca dischetă sistem pentru încărcare, la terminarea programului ar trebui utilizată comanda DOS "sys a:", pentru a transpune fișierele sistem pe dischetă.

Programul se startează la nivel DOS cu comanda "MFORMAT". După scurt timp pe ecran apare meniul principal:

Meniul principal MultiFormat pe unitati de 5,25"
if Freeware

Sint disponibile urmatoarele optiuni:
(sus-jos alege, Enter starteza)

Unitatea : A

Verificare: da

360 kbyte

-----420 kbyte-----

720 kbyte

800 kbyte

1.2 Mbyte

1.4 Mbyte

Terminare

Cu ajutorul tastelor cursor (sus-jos), bara colorată poate selecta punctul de meniu dorit. O apăsare pe tasta RETURN determină selecția punctului de meniu respectiv. Se poate lucra după dorință pe discheta A: sau B:, cu sau fără verificare. Din motive de siguranță, se recomandă să se lucreze cu opțiunea de verificare, pentru a asigura o mai mare securitate a datelor. Dacă

există însă certitudinea că discheta este completamente fără erori, se poate renunța la verificare, accelerând astfel formatarea. Porțiunea deja formatată a discului este afișată de acest program sub formă de procente.

La terminarea procesului de formatare, programul cere un nume pe care să-l primească discheta. La introducerea unui nume, programul stochează pe dischetă odată cu acesta și data și ora curentă. Dacă discheta urmează să fie folosită ca și dischetă sistem, în acest loc nu trebuie dat un nume ci doar acționată tasta RETURN. După terminarea programului de formatare se pun și fișierele sistem pe dischetă, cu comanda "sys".

Programul a fost astfel conceput încât o modificare a formatelor de dischetă este ușor de operat. De exemplu, se pot modifica ușor numărul de piste respectiv al numărului de sectoare/pistă în apelul rutinelor corespunzătoare de formatare.

Fără o modificare suplimentară în program însă nu se pot formata decât cel mult 10 sectoare/pistă. Dacă se dorește o formatare cu mai mult de 10 sectoare/pistă, mai trebuie modificată și tabela de parametri ai unității. Această tabelă de 11 octeți este adresată

Tabela de parametri ai unității

Octet	Funcție
0	timpul maximal necesitat de capul de citire scriere pentru a țrece de la o pistă la alta (biții 0-3); timpul de descărcare al capului (biții 4-7)
1	bit 0: 1- controller-ul unității folosește DMA; biții 2-7: timpul de încărcare al capului
2	specifică timpul cât motorul unității rămâne încă activ după efectuarea unei operații, folosind un tact de timer de 18,2 impulsuri/secundă. O valoare de 36 de exemplu determină un timp de cca. 2s pînă la deselectarea motorului
3	numărul de octeți/sector 0 - 128 octeți 1 - 256 octeți 2 - 512 octeți 3 - 1024 octeți
	pentru dischete și unități uzuale se folosește valoarea 2.
4	numărul maxim de sectoare/pistă
5	intervalul logic (gap) pînă la următorul sector; definește un interval care trebuie să treacă înainte ca următorul sector să poată fi accesat sub capul de citire/scriere.
6	lungimea transferului de date; definește lungimea maximă a unui transfer pentru cazurile în care acesta nu e specificat în comandă.
7	intervalul fizic (gap la formatare) pînă la următorul sector; trebuie să fie totdeauna mai mare decât intervalul logic!
8	caracterul cu care sînt inițializați toți octeții de date în cursul formătărilor; de obicei, 0f6h.
9	definește timpul de stabilire al capului după o schimbare de poziție, în unități de 1 ms; depinde de tipul unității de dischete.
10	intervalul care trebuie să treacă înainte ca motorul să ajungă la turația nominală, în unități de 1/8 secunde.

de Interrupt 1eh. Pointerul către această tabelă se găsește în memorie la adresa 0000:0078. în ea se găsesc parametri controller-ului de floppy asociat.

Dacă se dorește o formatare cu mai mult de 9 sectoare/pistă, trebuie modificate intervalul logic respectiv fizic între sectoare specificat în această tabelă. Valori exacte nu pot fi specificate. Găsirea valorilor potrivite se poate face numai experimentând.

Dischetele formate în acest fel pot fi desigur copiate cu ajutorul programului "DISKCOPY". în acest

caz însă totdeauna și discheta țintă trebuie în prealabil formatată la fel; altfel programul se abortează cu un mesaj de eroare.

Indicație: cei interesați în acest program pot comanda o dischetă HD noname, nouă, cu programul pe ea, la prețul de 300 lei, de la redacția revistei.

(*Computer Persönlich 26/89,*
Völker Bührmann,; Iosif Fettich;

Listing-ul programului Mformat;

```

program mformat
(* if - freeware *)
(* cu afisarea capacitatii formate in procente *)
uses crt, dos;
const
    maxwrite = 12; (* lungimea maxima a mesajelor *)
    reiries = 10; (* 10 incercari la erori de citire *)
    lng = 2; (* lungimea unui sector - totdeauna 512 octeti *)
    bootsec:
        array [1..354] of byte =
($eb,$3e,$90,$4d,$2e,$46,$4f,$52,$4d,$41,$54,$00,lng,$00,$01,$00,
$02,$70,$00,$d0,$02,$fd,$02,$00,$09,$00,$02,$00,$00,$00,$00,$00,
$00,$00,$00,$00,$0f,$00,$00,$00,$00,$01,$00,$00,$00,$00,$00,$00,
$00,$00,$06,$00,$01,$04,$0c,$00,$00,$00,$00,$00,$00,$00,$00,$00,
$fa,$fc,$33,$c0,$8e,$d0,$bc,$00,$7c,$36,$c5,$36,$78,$00,$1e,$56,
$8e,$c0,$8d,$3e,$20,$7c,$b9,$0b,$00,$ac,$26,$80,$3d,$00,$74,$03,
$26,$8a,$05,$aa,$e2,$f3,$33,$c0,$8e,$d8,$a3,$7a,$00,$c7,$06,$78,
$00,$20,$7c,$fb,$cd,$13,$73,$03,$e9,$b4,$00,$8a,$2e,$30,$7c,$8a,
$0e,$32,$7c,$8a,$36,$31,$7c,$b2,$00,$bb,$00,$05,$b8,$01,$02,$cd,
$13,$72,$e8,$8d,$36,$62,$7d,$8b,$fb,$b9,$0b,$00,$f3,$a6,$75,$0d,
$bf,$20,$05,$b9,$0b,$00,$f3,$a6,$74,$1a,$e9,$89,$00,$8d,$36,$78,
$7d,$8b,$fb,$b9,$0b,$00,$f3,$a6,$75,$7c,$bf,$20,$05,$b9,$0b,$00,
$f3,$a6,$75,$72,$33,$d2,$a1,$1c,$05,$f7,$36,$0b,$7c,$fe,$c0,$a2,
$38,$7c,$bb,$00,$07,$8a,$2e,$33,$7c,$8a,$0e,$35,$7c,$8a,$36,$34,
$7c,$b2,$00,$a1,$18,$7c,$2a,$c1,$fe,$c0,$50,$b4,$02,$cd,$13,$58,
$72,$3d,$28,$06,$38,$7c,$76,$1a,$b4,$00,$52,$f7,$26,$0b,$7c,$5a,
$03,$d8,$b1,$01,$fe,$c6,$3a,$36,$1a,$7c,$72,$d7,$fe,$c5,$b6,$00,
$eb,$d1,$cd,$11,$d0,$e0,$d0,$c0,$25,$03,$00,$75,$01,$40,$40,$8b,
$c8,$b8,$00,$00,$b2,$00,$8b,$1e,$36,$7c,$ea,$00,$00,$70,$00,$8d,

```

```

$36,$8e,$7d,$eb,$05,$90,$8d,$36,$a2,$7d,$ac,$0a,$c0,$74,$09,$bb,
$07,$00,$b4,$0e,$cd,$10,$eb,$f2,$8d,$1e,$c2,$7d,$3b,$f3,$77,$04,
$8b,$f3,$eb,$e6,$32,$e4,$cd,$16,$8f,$06,$78,$00,$8f,$06,$7a,$00,
$cd,$19);
boottext1:
    array [1..136] of char =
        'IBMBIO COM'+ 'IBMDOS COM'+
        'IO SYS'+ 'MSDOS SYS'+
        '#10#13'DiskBootFailure'#00#10#13+
        'Non-system disk or disk error'#00#10#13+
        'Replace and press any key when ready'#10#13;

    (**** constante pentru dischete de 360 kbyte ****)
    form1 = 2; (* 360 kbyte in unitate de 1,2 Mbyte *)
    media1 = $fd; (* mediabyte *)
    secperclu1 = 2; (* sectoare per cluster *)
    radint1 = $070; (* intrari in radacina *)
    secperfat1 = 2; (* sectoare per FAT *)
    capete1 = 2; (* numarul de capete (fete) *)
    (**** constante pentru dischete de 720 kbyte ****)
    form2 = 2; (* 360 kbyte in unitate de 1,2 Mbyte *)
    media2 = $f9; (* mediabyte *)
    secperclu2 = 2; (* sectoare per cluster *)
    radint2 = $070; (* intrari in radacina *)
    secperfat2 = 3; (* sectoare per FAT *)
    capete2 = 2; (* numarul de capete (fete) *)
    (**** constante pentru dischete de 1,2 Mbyte ****)
    form3 = 3; (* 1,2 Mbyte in unitate de 1,2 Mbyte *)
    media3 = $f9; (* mediabyte *)
    secperclu3 = 1; (* sectoare per cluster *)
    radint3 = $0e0; (* intrari in radacina *)
    secperfat3 = 7; (* sectoare per FAT *)
    capete3 = 2; (* numarul de capete (fete) *)
    maxsec = 17; (* numarul maxim de sectoare *)

type
    disctab = array [1..11] of byte;
    disctampon = array [1..512*maxsec] of byte;
    progmenu = array [1..9] of string [20];

var
    tabvechi : pointer; (* pointer la vechea tabela a unitatii *)
    eroare : boolean; (* indicatia de eroare *)
    verify : boolean; (* verificare discheta *)
    coderoare : byte; (* codul de eroare *)
    unitate : byte; (* numarul unitatii de dischete *)
    selector : byte; (* selectia din meniu *)

```

```

punctmeniu : byte;          (* punctul de meniu actual *)
meniu : progmeniu;        (* linia meniu *)
tabunit : disctab;       (* noua tabela a unitatii *)
tampon : disctampon;     (* tampon de 15 sectoare *)
pista1,
pista2,
pista3 : byte;           (* distzele diverselor formate *)
secperf1,
secperf2,
secperf3: byte;         (* sectoare per pista *)

```

```

(.....
Subrutine .....*)

```

```

(* ..... *)
procedure rama (x1,y1,x2,y2 : byte; nume : string);
const
  ss = $d5;
  ds = $b8;
  sj = $d4;
  dj = $be;
  oz = $cd;
  vr = $b3;

var
  i : byte;
  lng : integer;          (* lungimea sirului cu numele ferestrei *)
  height,
  width : byte;          (* inaltimea si respectiv largimea ferestrei *)
  titlu : boolean;

```

```

begin
  window(1,1,80,25);    (* tot ecranul *)
  textbackground(black);
  width := x2-x1;        (* largimea ferestrei *)
  height := y2-y1;      (* inaltimea ferestrei *)
  titlu := true;        (* fereastra sa primeasca un nume *)
  lng := length(nume);
  if lng width then
    titlu := false;     (* renuntam la nume pentru fereastra *)
  textcolor(white);
  if titlu then         (* cind fereastra are titlu *)
    begin
      gotoxy(x1,y1);
      write(ss);        (* coltul din stanga sus *)
      for i := 2 to width do
        write(oz);
      write(ds);
      gotoxy((x1-round(width/2)-round(lng/2)+1,y1);
      write(nume);
    end
  else
    begin
      gotoxy(x1,y1);
      write(ss);
      for i := 2 to width do
        write(oz);
      write(ds);
    end;
  for i := 1 to height-1 do
    begin
      gotoxy(x1,y1+i);
      write(vr);
      gotoxy(x2,y1+i);
      write(vr);
    end;
  gotoxy(x1,y2);
  write(sj);            (* coltul din stanga jos al ferestrei *)
  for i := 2 to width do
    write(oz);
  write(dj);

```

```

window(x1+1,y1+1,x2-2,y2-1);
textbackground(red);
clrscr;
end;

```

```

(* ..... *)

```

```

function select(x1,y1,st,ed : byte; var punctmeniu : byte) : byte;

```

```

var
  i : byte;
  lng : byte;
  poz : byte;
  ch : char;

begin
  lng := ed-st;
  textbackground(red);
  textcolor(yellow);
  window(x1,y1,x1+20,y1+lng+1);
  for i := st to ed-1 do
    writeln(meniu [i]);
  write(meniu [ed]);
  poz := (punctmeniu-st)+1;
  (* pozitia relativa la incep. meniului *)
  gotoxy(1,poz);
  textbackground(white);
  textcolor(black);
  write(meniu [punctmeniu]);
  repeat
    ch := readkey;
    i := 0;
    case ch of
      #80:
        begin (* sageata in jos *)
          gotoxy(1,poz);
          textbackground(lightred);
          textcolor(yellow);
          write(meniu [st+poz-1]);
          inc(poz);
          if poz lng+1 then
            poz := 1;
          gotoxy(1,poz);
          textbackground(white);
          textcolor(black);
          write(meniu [st+poz-1]);
        end;
      #72:
        begin (* sageata in sus *)
          gotoxy(1,poz);
          textbackground(lightred);
          textcolor(yellow);
          write(meniu [st+poz-1]);
          dec(poz);
          if poz 1 then
            poz := lng+1;
          gotoxy(1,poz);
          textbackground(white);
          textcolor(black);
          write(meniu [st+poz-1]);
        end;
    end;
  until ch=#13;        (* RETURN *)
  punctmeniu := st+poz-1;
  select := punctmeniu;
end;

```

```

(* ..... *)

```

```

procedure clearcur;

```

```

var
  cpu : registers;
begin

```

```

cpu.ah := 1;      (* forma de aparitie a cursorului *)
cpu.ch := 15;    (* linia de inceput a cursorului *)
cpu.cl := 0;     (* linia de sfirsit a cursorului *)
intr($10,cpu);  (* cursorul *)
end;

(* ----- *)

procedure setcur;

var
  cpu : registers;

begin
  cpu.ah := 1;
  cpu.ch := 11;
  cpu.cl := 12;
  intr($10,cpu);
end;

(* ----- *)

procedure selectdisc(var unitate : byte; var meniu: progmeniu);

begin
  if unitate = 0 then
    begin
      unitate := 1;
      meniu[1] := ' Unitatea : B
    end
  else
    begin
      unitate := 0;
      meniu[1] := ' Unitatea : A
    end;
end;

(* ----- *)

procedure verifdisc(var verify : boolean; var meniu : progmeniu);

begin
  if verify then
    begin
      verify := false;
      meniu[2] := ' Verificare: nu
    end
  else
    begin
      verify := true;
      meniu[2] := ' Verificare: da
    end;
end;

(* ----- *)

procedure tabelunitatenou (nrsect, fizic : byte);
(* creeaza nou tabel ptr. unitate *)

begin
  tabunit[1] := $df; (* timpul maximal ptr. schimbarea pistei *)
  tabunit[2] := 2;   (* modul DMA *)
  tabunit[3] := $25; (* cit mai ramine activ motorul *)
  tabunit[4] := lng; (* 512 octeti per sector *)
  tabunit[5] := nrsect; (* cite sectoare / pista *)
  tabunit[6] := $1b; (* intervalul logic pina la urmatorul sector *)
  tabunit[7] := $ff; (* lungimea datelor transferate *)
  tabunit[8] := fizic; (* intervalul fizic pina la urmatorul sector *)
  tabunit[9] := $f6; (* octetul de umplere *)
  tabunit[10] := 1;  (** *)
  tabunit[11] := 8; (* alti octeti de comanda *)
  getintvec($1e,tabvechi);
  setintvec($1e,@tabunit);

```

```

end;

(* ----- *)

procedure tabvechiunit;

begin
  setintvec($1e,tabvechi);
end;

(* ----- *)

procedure diskreset;

var
  contor : byte;
  cpu : registers;

begin
  contor := retries;
  repeat
    cpu.ah := 0; (* resetarea unitatilor atasate *)
    cpu.dl := 0;
    intr($13,cpu);
    dec(contor);
  until (contor 1) or ((cpu.flags and 1) = 0);
end;

(* ----- *)

procedure formatdisc(form : byte);

var
  cpu : registers;
  contor : byte;

begin
  contor := retries;
  repeat
    cpu.ah := $17; (* stabilirea formatului dischetei *)
    cpu.al := form; (* felul formatarii *)
    intr($13,cpu);
    dec(contor);
    if (cpu.flags and 1) = 1 then
      diskreset;
    until (contor 1) or ((cpu.flags and 1) = 0);
end;

(* ----- *)

function formatare(nrpiste, nrsectoare : byte; titlu : string;
var tampon: disctampon; var errcode : byte) : boolean ;

type formatrec = record
  trackdisk,
  fatadisk,
  sectordisk,
  octetcontor : b yte;
end;

var
  cpu : registers;
  contor : byte;
  pista : byte;
  sector : byte;
  fata : byte;
  formproc: byte;
  secformat: word;
  formmax : word;
  formattab : array[1.. maxsec] of formatrec;

begin
  rama(22,19,58,23,titlu);
  formmax := nrpiste * nrsectoare * 2;

```

```

(* nr. max. sectoare pe discheta *)
secformat := 0; (* nr. sectoarelor formate *)
gotoxy(7,2);
write('% din discheta format');
for pista := 0 to nrpiste-1 do
begin
for fata := 0 to 1 do (* doua fete *)
begin
contor := retries;
for sector := 1 to nrsectoare do
begin
formattab [sector] .trackdisk := pista;
formattab [sector] .fatadisk := fata;
formattab [sector] .sectordisk := sector;
formattab [sector] .octetcontor := lng; (* 512 bytes/sector *)
end;
textcolor(white);
inc(secformat,nrsectoare);
formproc := round(secformat/formmax*100);
gotoxy(3,2);
write(formproc:3);
repeat
(* formatare pista *)
cpu.ah := 5; (* numarul unitatii *)
cpu.dl := uniteate;
cpu.dh := fata;
cpu.ch := pista;
cpu.al := nrsectoare;
cpu.es := seg(formattab);
cpu.bx := ofs(formattab);
intr($13,cpu);
if verify then
begin
cpu.ah := 4; (* verificare pista *)
cpu.dl := uniteate;
cpu.dh := fata;
cpu.cl := 1; (* incepind cu sectorul 1 *)
cpu.al := nrsectoare; (* verifica toate sectoarele *)
cpu.es := seg(tampon);
cpu.bx := ofs(tampon);
intr($13,cpu);
end;
dec(contor);
if (cpu.flags and 1) = 1 then
diskreset;
until (contor = 1) or ((cpu.flags and 1) = 0);
if (cpu.flags and 1) = 1 then
begin
coderoare := cpu.ah;
formatare := true;
exit; (* retur in programul apelant *)
end;
end;
end;
formatare := false;
end;
(* ----- *)

```

```

function disc_prep(media,secperclu,radint : byte; nrsectoare : word;
secperfat,secperpista : byte; var errcod : byte) : boolean;

```

```

var
contor : byte;
numar : byte;
i : integer;
pointer : integer;
cpu : registers;

begin
numar := secperfat*2+round(radint*32/512);
(* sectoare FAT si root *)
for i := 1 to secperpista * 512 do
tampon [i] := 0; (* sterge tamponul *)
for i := 1 to 354 do

```

```

tampon [i] := bootsec [i];
tampon [14] := secperclu; (* sectoare per cluster *)
tampon [18] := lo(radint);
tampon [19] := hi(radint); (* nr. inregistrarilor in radacina *)
tampon [20] := lo(nrsectoare);
tampon [21] := hi(nrsectoare); (* nr. total de sectoare *)
tampon [22] := media; (* mediabyte-ul *)
tampon [23] := lo(secperfat);
tampon [24] := hi(secperfat);
tampon [25] := secperpista;
(* sectoare pe o fata intr-o pista *)
for i := 1 to 136 do
tampon [i+354] := ord(boottext1 [i]);
tampon [511] := $55;
tampon [512] := $aa; (* sfirsitul sectorului de boot *)
contor := retries;
repeat
cpu.ah := 3; (* scrierea bootsectorului pe discheta *)
cpu.dl := uniteate;
cpu.dh := 0; (* fata bootsectorului *)
cpu.ch := 0; (* pista bootsectorului *)
cpu.cl := 1; (* numarul bootsectorului *)
cpu.al := 1; (* numarul de sectoare *)
cpu.es := seg(tampon);
cpu.bx := ofs(tampon);
intr($13,cpu); (* scriere bootsector *)
dec(contor);
if (cpu.flags and 1) = 1 then
diskreset;
until (contor = 1) or ((cpu.flags and 1) = 0);
if (cpu.flags and 1) = 1 then
disc_prep := true (* a aparut o eroare *)
else
begin
for i := 1 to 512 do
tampon [i] := 0;
(* sterge bootsectorul in tampon *)
tampon [1] := media;
tampon [2] := $ff;
tampon [3] := $ff;
pointer := secperfat * 512;
tampon [pointer+1] := media;
tampon [pointer+2] := $ff;
tampon [pointer+3] := $ff;
cpu.ah := 3; (* scriere sectoare pe discheta *)
cpu.dl := uniteate;
cpu.dh := 0; (* fata dischetei *)
cpu.ch := 0; (* numarul pistei *)
cpu.cl := 2; (* incepind cu sectorul 2 *)
cpu.al := secperpista - 1;
(* numarul sectoarelor ce trebuiesc scrise *)
cpu.es := seg(tampon);
cpu.bx := ofs(tampon);
intr($13,cpu);
dec(numar,secperpista-1);
for i := 1 to 512*numar do
tampon [i] := 0; (* sterge tampon *)
cpu.ah := 3; (* scriere sectoare pe discheta *)
cpu.dl := uniteate;
cpu.dh := 0; (* fata dischetei *)
cpu.ch := 1; (* numarul pistei *)
cpu.cl := 1; (* incepind cu sectorul 1 *)
cpu.al := numar;
cpu.es := seg(tampon);
cpu.bx := ofs(tampon);
intr($13,cpu);
end;
if (cpu.flags and 1) = 1 then
begin
disc_prep := true;
diskreset;
end

```

```

else
    disc_prep := false;
end;

(* ----- *)
procedure numedisc(secperfat,secperpista : byte);
var
    key : char;
    nume : string [ 11 ]; (* numele dischetei *)
    i : integer;
    cpu : registers;
    sector: b byte;
    fata : byte;
    an,
    luna,
    zi,
    ziasaptaminii,
    ora,
    minut,
    secunda,
    sutime : word;
    data,
    time : word;
begin
    window(22,18,58,22);
    textbackground(black);
    clrscr;
    rama(16,19,63,25,'Formatare terminata cu succes. ');
    gotoxy(4,2);
    write(disksize(unitate+1),' byte disponibili pe discheta ');
    gotoxy(12,4);
    write('Continua apasind Enter ');
    while key #13 do
        key := readkey;
    end;
    clrscr;
    setcur;
    gotoxy(6,3);
    write('Numele dischetei: [    ] ');
    gotoxy(25,3);
    readln(nume); (* preia numele dischetei *)
    clearcur;
    if length(nume) > 0 then
        begin
            for i := 1 to 512 do
                tampon [ i ] := 0; (* sterge tampon *)
            end;
            for i := 1 to length(nume) do
                tampon [ i ] := ord(nume [ i ]);
                (* trece numele in tampon *)
            end;
            tampon [ 12 ] := 8; (* atributul pentru nume *)
            getdate(an,luna,zi,ziasaptaminii);
            (* preia data din sistem *)
            gettime(ora,minut,secunda,sutime);
            (* preia ora din sistem *)
            time := (ora shl 11) or (minut shl 5) or secunda;
            dec(an,1980);
            data := (an shl 9) or (luna shl 5) or zi;
            tampon [ 23 ] := lo(time);
            tampon [ 24 ] := hi(time);
            tampon [ 25 ] := lo(data);
            tampon [ 26 ] := hi(data);
            fata := 0; (* prima fata a dischetei *)
            sector := 2*secperfat+2;
            (* primul root-sector *)
            if sector < secperpista then
                begin
                    fata := 1; (* fata a doua a dischetei *)
                    dec(sector,secperpista);
                    (* sector pe fata a doua a dischetei *)
                end;
            end;
            cpu.ah := 3; (* scriere sectoare pe discheta *)
            cpu.dl := unitate;
        end;
    end;
end;

```

```

cpu.dh := fata; (* fata dischetei *)
cpu.ch := 0; (* numarul pistei *)
cpu.cl := sector; (* incepind cu sectorul 2 *)
cpu.al := 1;
(* numarul sectoarelor ce trebuiesc scrise *)
cpu.es := seg(tampon);
cpu.bx := ofs(tampon);
intr($13,cpu);

end;
window(16,19,63,25);
textbackground(black);
clrscr;
end;

(* ----- *)
procedure mesajeroare(coderoare : byte);
var
    key : char;
begin
    window(22,18,58,22);
    textbackground(black);
    clrscr;
    rama(22,19,58,23,'Eroare!!!');
    gotoxy(8,2);
    if coderoare = 3 then
        begin
            gotoxy(6,2);
            write('Discheta e protejata la scriere');
        end
    else
        begin
            gotoxy(6,2);
            write('Discheta e inutilizabila');
            gotoxy(2,3);
            write('sau nu e de tipul cerut (DD,HD).');
        end;
    end;
    key := readkey;
    window(22,19,58,23);
    textbackground(black);
    clrscr;
end;

(* ----- *)
procedure format360(pista1,secperpista1,fgap : byte; titlu : string);
(* max. 42 piste *)
var
    nrsectoare : word;
begin
    nrsectoare := capete1*secperpista1*pista1;
    (* nr. total de sectoare *)
    tabelunitatenou(secperpista1,fgap);
    diskreset;
    formatdisc(form1);
    eroare := formatare(pista1,secperpista1,titlu,tampon,coderoare);
    if eroare then
        mesajeroare(coderoare)
    else
        begin
            eroare := disc_prep(media1,secperclu1,radint1,nrsectoare,
            secperfat1,secperpista1,coderoare);
            if eroare then
                mesajeroare(coderoare)
            else
                begin
                    numedisc(secperfat1,secperpista1);
                end;
            end;
        end;
    end;
    tabvechiunit;
end;

```

```

end;

(* ----- *)

procedure format720(pista2,secperpista2,fgap : byte; titlu : string);
  (* max. 84 piste *)

var
  nrsectoare : word;

begin
  nrsectoare := capete2*secperpista2*pista2;
    (* nr. total de sectoare *)
  tabelunitatenou(secperpista2,fgap);
  diskreset;
  formatdisc(form2);
  mem [$0040:$0090+unitate] := $54;
    (* comutare pe pas simplu *)
  eroare := formatare(pista2,secperpista2,titlu,tampon,coderoare);
  if eroare then
    mesajeroare(coderoare)
  else
    begin
      eroare := disc_prep(media2,secperclu2,radint2,
        nrsectoare,secperfat2,secperpista2,coderoare);
      if eroare then
        mesajeroare(coderoare)
      else
        begin
          numedisc(secperfat2,secperpista2);
        end;
    end;
  end;
  tabvechiunit;
end;

(* ----- *)

procedure format12m(pista3,secperpista3,fgap : byte; titlu : string);
  (* max. 42 piste *)

var
  nrsectoare : word;

begin
  nrsectoare := capete3*secperpista3*pista3;
    (* nr. total de sectoare *)
  tabelunitatenou(secperpista3,fgap);
  diskreset;
  formatdisc(form3);
  eroare := formatare(pista3,secperpista3,titlu,tampon,coderoare);
  if eroare then
    mesajeroare(coderoare)
  else
    begin
      eroare := disc_prep(media3,secperclu3,radint3,
        nrsectoare,secperfat3,secperpista3,coderoare);
      if eroare then
        mesajeroare(coderoare)
      else
        begin
          numedisc(secperfat3,secperpista3);
        end;
    end;
  end;
  tabvechiunit;
end;

(* ----- *)

```

```

(*-----*)
Programul principal
(*-----*)

begin
  clearcur;
  textbackground(black);
  clrscr;
  rama(20,1,60,17,'Meniulprincipal');
  textcolor(yellow);
  gotoxy(5,2);
  write('MultiFormat pe unitati de 5,25"');
  textcolor(white);
  gotoxy(14,3);
  write('if Freeware ');
  textcolor(yellow);
  gotoxy(2,4);
  write('Sint disponibile urmatoarele optiuni:');
  gotoxy(4,5);
  write('sus-jos alege, Enter starteaza');
  menu 1 := ' Unitatea : A';
  menu 2 := ' Verificare: da';
  menu 3 := ' 360 kbyte';
  menu 4 := ' 420 kbyte';
  menu 5 := ' 720 kbyte';
  menu 6 := ' 800 kbyte -DD';
  menu 7 := ' 1.2 Mbyte';
  menu 8 := ' 1.4 Mbyte -HD';
  menu 9 := ' Terminare';
  punctmenu := 3;
  unitate := 0;
  verify := true;
  checkbreak := false;(* ctrl-break oprit pe durata formatarii *)
  repeat
    selector := select(30,7,1,9,punctmenu);
  case selector of
    1:
      selectdisc(unitate,menu);
    2:
      verifdisc(verify,menu);
    3:
      format360(40,9,$54,'Formatare discheta 360 kbyte');
    4:
      format360(42,10,$33,'Formataredischeta420 kbyte');
    5:
      format720(80,9,$54,'Formataredischeta 720 kbyte');
    6:
      format720(80,10,$33,'Formataredischeta800 kbyte');
    7:
      format12m(80,15,$54,'Formataredischeta 1,2 Mbyte');
    8:
      format12m(82,17,$18,'Formataredischeta 1,4 Mbyte');
    9:
      end;
  until selector = 9;
  window(1,1,80,25);
  textbackground(black);
  textcolor(lightmagenta);
  clrscr;
  setcur;
end.

```

Prețuri, prețuri

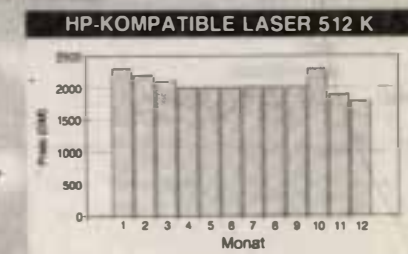
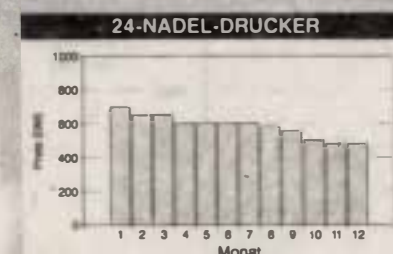
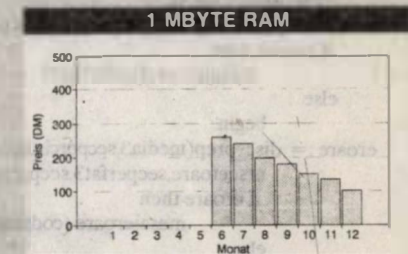
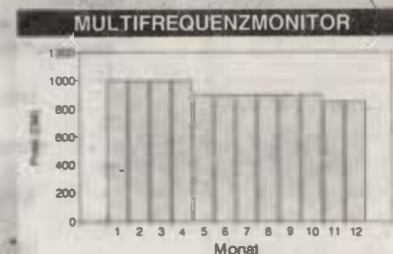
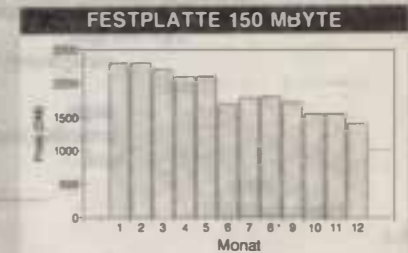
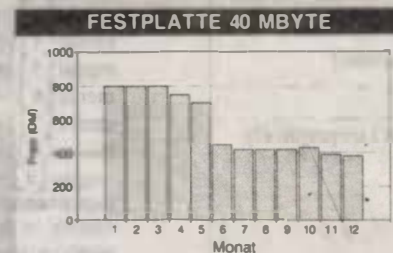
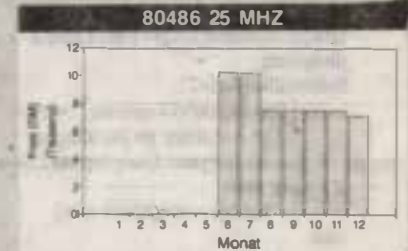
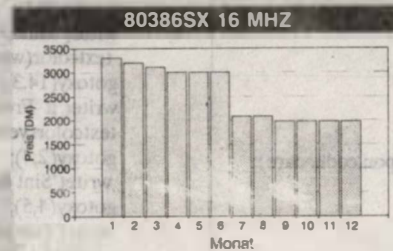
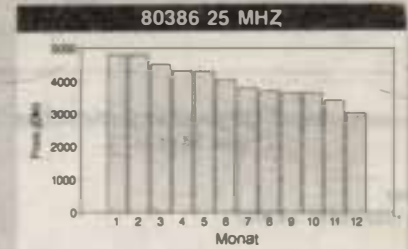
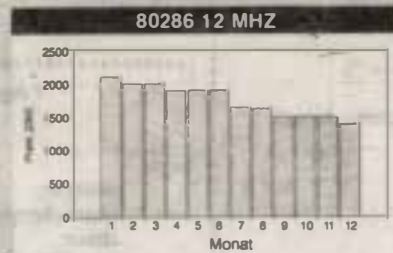
Rubrica aceasta vă oferă o sinteză a prețurilor pe piața germană. În diagramă sînt notate sînt de obicei cele mai joase prețuri relative la un produs, așa -numite prețuri de 'stradă' și de aceea ele pot să se abată mult de la recomandările de preț ale producătorilor. Prețurile relative la PC-uri se referă la următoarele configurații complete, cu placă Hercules și monitor monocrom: 80286 (12MHz) cu 512 kbyte RAM și harddisk de 20 Mbyte, 80386SX (16MHz) cu 1 Mbyte RAM și harddisk de 20 Mbyte precum și 80386 sau 80486 (25 MHz) cu 2 Mbyte RAM și harddisk de 40 Mbyte

Vă rugăm să aveți în vedere că produsul cu prețul cel mai avantajos deseori nu este echivalent în ceea ce privește compatibilitatea, calitatea și siguranța în funcționare cu produse funcțional echivalente, dar mai scumpe.

Tendențe

În domeniul imprimantelor Laser compatibile HP este iminentă o cădere a prețurilor. Mulți distribuitori oferă imprimante Laser sub 2000 DM. Unii producători vor scoate pe piață imprimante Laser noi, ieftine în comparație cu performanțele relativ ridicate de care vor fi capabile, și vor să scape de stocurile de modele vechi. Clientul potențial poate profita acum de scăderile de prețuri, deoarece noile modele probabil nu vor 'ieși' înainte de primăvara '91, cînd prețurile vor fi iarăși puțin mai ridicate.

Mic dicționar german - român:
 Farbmonitor = monitor color
 Festplatte = harddisk
 Nadel-Drucker = imprimantă matriceală
 APC + Technik, 1/91



Vreau să cumpăr un AT - de unde și ce pot să cumpăr?

Aceasta a fost întrebarea cu care am pornit un sumar raid-anchetă telefonic. Cu intenția de a afla cam care sînt prețurile și cam cum stă piața, am contactat firmele pe care le veți găsi înșirate mai jos. Desigur, acest raid nu este nici pe departe exhaustiv, și poate nici foarte reprezentativ: pur și simplu am testat firmele de a căror existență într-un fel sau altul am aflat. Ne va face plăcere să facem publice și alte oferte, în măsura în care ne vor parveni. Cele de mai jos sînt date relativ la care nu ne asumăm nici un fel de răspundere - le-am notat în dimineața zilei de 10 ianuarie 1991, discutînd de la anonim la anonim.

Firma	Uc cu	Tact UC (MHz)	Memo UC (Mbyte)	Hard disk (Mbyte)	Ecran	Floppy	Garanție	Preț
1)ABTech								
	286	16	1	40	VGA color	1x5,25"+ 1x3,5"	1 an	cca. 550.000
	386	25/33	2	40	VGA color	"	"	cca. 650.000
Alte amănunte: inclusiv mouse; producție S.U.A.; calculatoarele vor fi trecute pe la Comisia națională de standarde								
2)ADMEX								
	286	12	1	40	VGA color	?	1 an	432.000 lei
Alte amănunte: suplimentar, imprimante 9 ace 80 col. - 72.000 lei								
3)Attigram								
	286	12	1	40	VGA color	1x3,5" + 1x5,25"	1an	390.000 lei + 60.000 lei
Alte amănunte: inclusiv maus, MS-DOS 4.01 și Windows 2.0								
4)BIPO International								
								90/17.06.26 <i>19.30.89</i> <i>Laurin Popa</i>
Alte amănunte: pe moment nu livrează nimic, datorită grevei TAROMS.R.L.								
5)Brodeco								
	286	12	1	20	EGA mono	1x5,25"		229.000 lei
					VGA mono	"		264.000 lei
					VGA color	"		304.000 lei
		16		40	EGA mono	"		320.000 lei
					VGA mono	"		361.000 lei
					VGA color	"		400.000 lei
		25	2	80	VGA color	1x5,25"+ 1x3,5"		692.000 lei
Alte amănunte: inclusiv mouse; 386 are și 64 kbyte cache; suplimentar se pot achiziționa: 8087 - 32.000 lei, 80387/16 - 86.000 lei, 80387/25 - 110.000 lei și imprimante Panasonic 80 col. 192 car./sec. - 57.000 lei, Canon- Epson 132 col. 160 car/sec - 85.000 lei*								
6)Datacom								
	286			40	VGA color	1x5,25"	6 luni	500.000 lei
				80	"	"	"	580.000 lei
Alte amănunte: inclusiv o imprimantă Epson cu 9 ace								
7)Infocomp								
	286	16	1	40	VGA color	1x5,25"+ 1x3,5"	1 an	1600 \$ sau echivalentul în lei
8)Micro ATCI								
	286	16	1	40	EGA color	1x5,25"	1 an	510.000 lei
Alte amănunte: inclusiv o imprimantă Epson FX-1000, mouse și două cutii cu dischete (1 DD, 1 HD)								
9)Microcomputer Service Ltd.								
	286	16	1	40	VGA color	1x5,25"+ 1x3,5"	1 an	650.000 lei
Alte amănunte: inclusiv mouse și o săptămînă de școlarizare; în plus, diverse: scannere, streamere etc.								
10)Mons Medius								
	286	8/12	1	40		1x5,25"+ 1x3,5"		210.240 lei
					+ Hercules			29.000 lei
					+VGA color			101.000 lei
Alte amănunte: calculatoarele sînt de producție RFG, Aquarius Systems International); suplimentar, toată gama de imprimante Star, de la Star SR 10 (48.000 lei) pînă la Star X/2415 (154.000 lei)								
11)Salto S.R.L.								
	286	16	1	40	VGA color	1x5,25"+ 1x3,5"	6 luni	710.000 lei
Alte amănunte: inclusiv imprimantă Epson LX800; separat, mouse 12.000 lei								
12)Romtrust								
								90/17.08.40
"standard"								
								2400 \$ (cca 450.000 lei)

Mică publicitate

Poșta redacției

Cumpăr HD cu controler IBM-PC compatibil, monitor EGA, 4 MP 8087-2. **Vind** monitor Orange, set programe, peste 140 (WS, DTP, CAD, Lotus, Harvard Graphics, 3D, Șah, dBase 3+, Mathe, GEM Paint, Write, Ediinf, Fox, Pascal, Prolog, simulator zbor, jocuri, utilitare) 90/717296, 18-22h

"Sava Soft" Buzău vă oferă:

- salarii CP/M, MS-DOS
 - balanța de verif..... CP/M, MS-DOS
 - mijloace fixe CP/M, MS-DOS
 - facturi-livrări CP/M, MS-DOS
 - stocuri (în preg) MS-DOS
- ing. Sava Stan, tel. 974/12760

Doresc colaborare sau scurte consultații pt. **Commodore 64** în special probleme legate de cod mașină Florin 90/803384

A apărut revista lunară de High Fidelity, electronică audio și software pentru generare de sound și compoziție muzicală. Abonamente la orice oficiu poștal. Adresa redacției: Trever Emil, str. Toplița nr.7, 1900 Timișoara, jud. Timiș, tel. 961/48631. Citiți și veți fi mulțumiți: color, 36 pagini.

SINTES 6

Universitatea din Craiova, Facultatea de automatică și calculatoare, organizează ediția a șasea a simpozionului național de teoria sistemelor, robotică, calculatoare și informatică de proces. Simpozionul se va desfășura în perioada 14 - 15 (!) martie 1991; Tematica destul de largă cuprinde printre altele:

- conducerea adaptivă a roboților
 - echipamente pentru testare metrologică
 - conducerea optimă, cu calculatoare numerice, a proceselor din energetică și industria chimică
 - procesoare dedicate pe aplicații
 - sisteme numerice cu prelucrare paralelă, transputere
- În timpul simpozionului, firmele Tektronix, Hewlett-Packard precum și firmele sponsorizatoare vor expune unele din produsele lor.

Alte informații: Cernian Oleg, Universitatea din Craiova, Facultatea de Automatică și Calculatoare, Catedra de Calculatoare, str. Lăpușului nr. 5, 1100 Craiova, tel. 941/45724, int. 168,169

Din cuprinsul numărului viitor:

- Programe de calcul tabelar (spreadsheet-uri)
 - Microprocesorul RISC intel 860
 - State-of-the-art: cele mai novatoare 5 concepte hard și soft ale anului 1990: digital video interface, TFT-LCD, MS-Windows 3.0, tehnologia flash, programarea orientată obiect
- Și încă multe altele!...**

Începem prin a ne scuza de două ori: întâi, în fața colaboratorului nostru, ing. Doboș Ștefan, care a făcut toate desenele din revistele noastre și pe care l-am omis sistematic din caseta redacției; apoi, în fața cititorilor, deoarece din considerente tehnice articolul relativ la microprocesorul RISC intel860 va apare abia în numărul viitor și nu în acest număr, cum a fost anunțat.

Mulțumim tuturor celor care ne-au scris de anul nou - și tuturor celor care ne scriu în general. Nu numai că ne ajută, dar ne sînt chiar indispensabile reacțiile la ceea ce punem în revistă: doar o facem să fie citită și, mai ales, să fie utilă.

Pentru a simplifica procedura de expediere, destul de laborioasă la multe expedieri cu ramburs, dorim să simplificăm procedura și să trecem la expedierea de recomandate cu confirmare de primire. Pentru aceasta, rugăm abonații noștri să ne trimită contravaloarea revistelor în avans, pentru numărul de luni care îi interesează. Plecăm de la premisa că vom reuși să păstrăm prețul revistei la 39 lei pe număr. Cei care doresc să se aboneze în acest fel sînt rugați să trimită mandate poștale cu confirmare de primire pe adresa redacției, pentru Ingrid Maier. (Confirmările ar trebui păstrate, ele fiind "chitanța" de abonament). O mențiune suplimentară pe mandat, relativă la faptul că cel care-l trimite e un abonat mai vechi care renunță la forma veche de abonament și trece pe forma nouă sau este un abonat nou poate numai să ne ajute.

Va fi mai simplu în acest fel și la expediere, dar și la primire, deoarece recomandatele (mai) vin acasă, fără să necesite (totdeauna) un drum pînă la poștă. Îi mai rugăm pe cei care doresc să facă abonamente să le facă pe 4 sau pe 10 luni, abonamentul fiind astfel valabil pînă la jumătatea anului sau respectiv pe tot anul 1991; primul număr care va veni pe abonament va fi numărul 3/1991; pînă atunci, nu se schimbă nimic. Mulțumim celor care ne vor ajuta să mai scăpăm un pic de formularistică, plătind anticipat, dar și tuturor celor care ne pot ajuta cu sugestii sau propuneri valabile relativ la simplificarea difuzării. Un caz fericit, dar singular, îl avem la Baia Mare, unde mai mulți cititori s-au abonat la un difuzor local (dl. Cudalbu Octavian, tel. 994/36433); și pentru noi, acest lucru reprezintă o ușurare semnificativă, deoarece nu mai trimitem 50 de plicuri ci un singur colet, iar în evidențele noastre figurează un singur abonat.

(ing. Iosif Fettich)

6. Urmatoarele articole nu mi-au plăcut:

.....
.....
.....

datorită | | subiectului | | formei de prezentare

7. Calificativul acordat acestui număr:

8. Propuneri de îmbunătățire:

.....
.....
.....

9. Mi-au displicut mai multe exprimări/termeni folosiți în revistă. Propun ca în locul lor să se folosească un alt echivalent.

Termen

Echivalent

.....
.....
.....
.....
.....
.....
.....

Data..... Semnătura..... If 1 / 91

**Informații la zi din
lumea calculatoarelor
personale puteți
obține numai citind
regulat revista**

"if" !

Micro ATCI

C.P. 64, O.P. 1

RO - 4300 Tîrgu Mureș

Aveți ceva de vîndut?

Doriți să

cumpărați ceva ?

Vreți să vă oferiți

serviciile sau aveți

nevoie de ajutor într-o

problemă ?

Folosiți mica publicitate

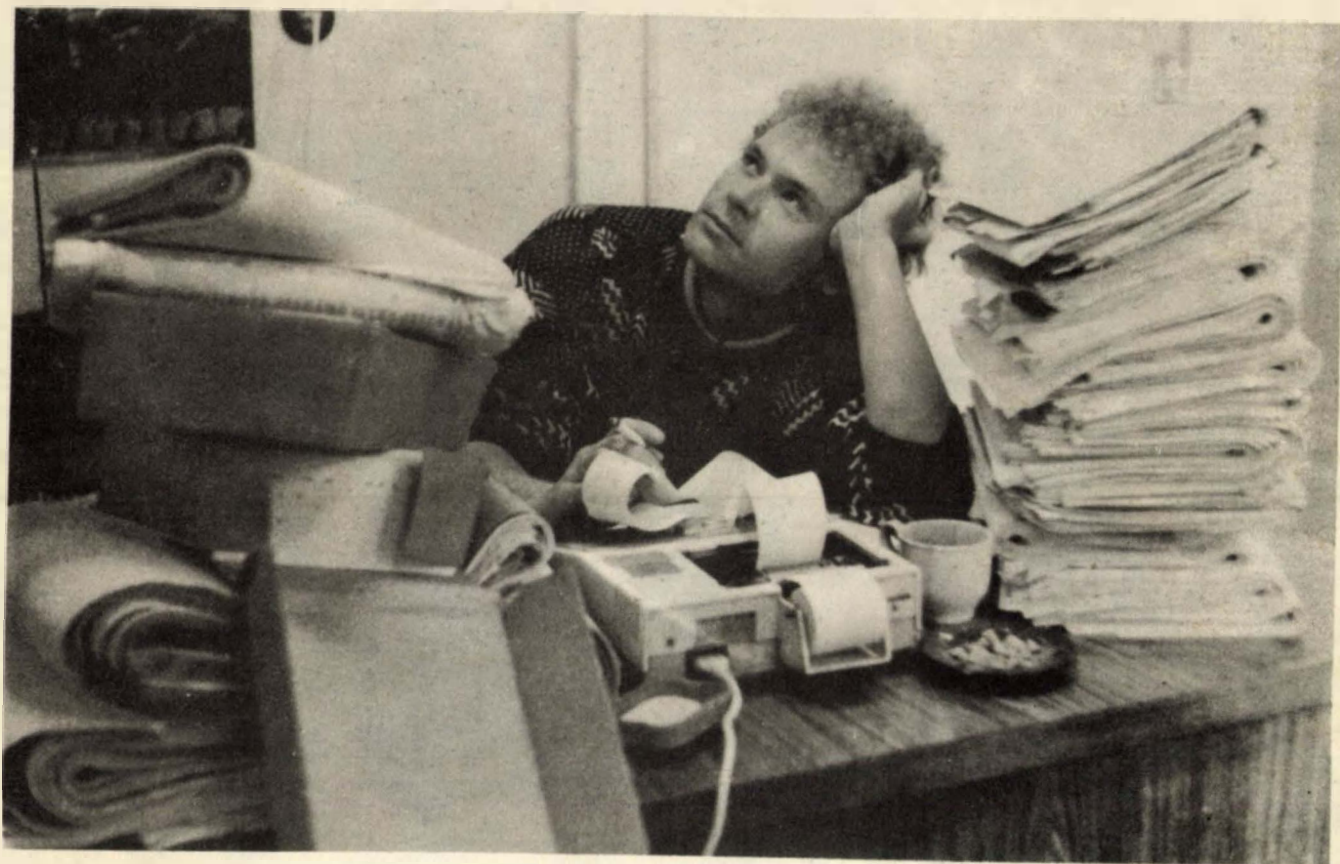
de specialitate din

"if" !

Micro ATCI

C.P. 64, O.P. 1

RO - 4300 Tîrgu Mureș



Don't worry, be happy !

Micro ATCI Tirgu Mures

va ofera o alternativa !

**Daca doriti sisteme IBM PC la cheie
(consulting, instalare, asistenta tehnica)
contactati-ne la adresa :**

Micro ATCI Tirgu Mures

C.P. 64

RO - 4300 Tirgu Mures

Tel. 954 / 17024

UNIX

OS/2

AIX

BS/2

Sun OS

PC-D

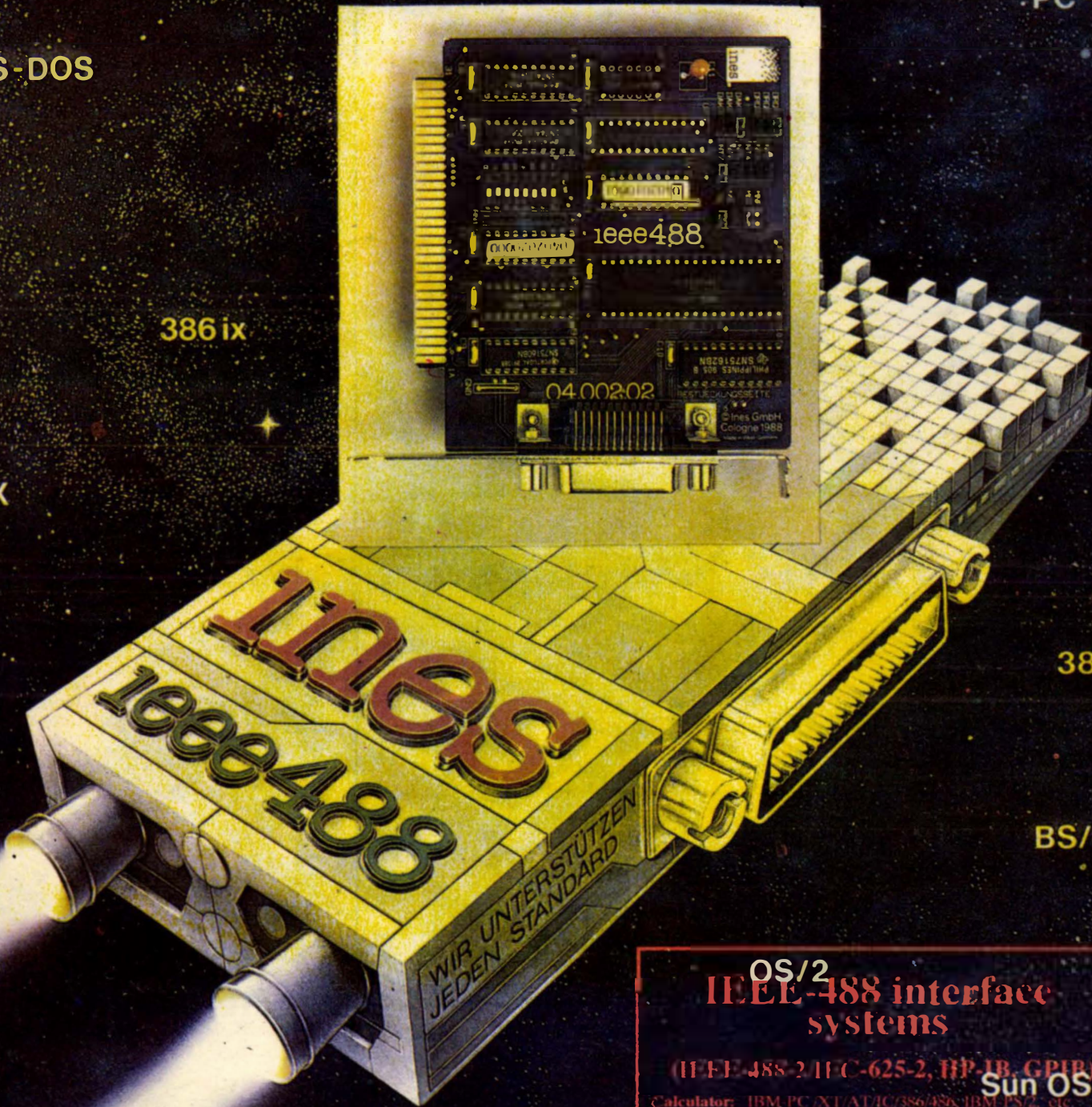
MS-DOS

386 ix

UNIX

38

BS/2



OS/2
IEEE-488 interface systems

(IEEE 488-2, IFC-625-2, HP-IB, GPIB)

Calculator: IBM PC XT/AT/IC/386/486-IBM PS/2, etc.

Sistem de operare: MS-DOS, OS/2, 386/ix, UNIX, XENIX etc.

Compatibil cu HP-Basic, ASYST/ASYSTANT GPIB,

Lotus Measure, GURU, TabWindow, DaDisp 488et

Rate de transfer: 100-200kbyte/sec (DOS)

35kbyte/sec (OS/2, UNIX)

330-500kbyte/sec (8-bit DMA)

1Mbyte/sec (16bit DMA)

Garantie: 5 ani

ines - INnovativé Elektronik Systeme GmbH
 Neuenhöfer Allee 45, 5000 Köln 41, Germany

PC-DOS

Sun OS

MS-D