

NAGY GÁBOR

**TÖMÖR
GYÖNYÖR**

ALAPLAP KÖNYVEK  **2**

NAGY GÁBOR

Tömör gyönyör

avagy a tömörítés magasiskolája
DOS alatt működő PC-ken
(PC/XT/AT/STB)

NAGY GÁBOR

Tömör gyönyör

avagy a tömörítés magasiskolája
DOS alatt működő PC-ken
(PC/XT/AT/STB)



CÉDRUS KIADÓ

ISSN 0866-434X
ISBN 963 7429 00X

© Nagy Gábor, 1991
Cédrus Informatikai Részvénytársaság
Felelős kiadó: Sebestyén Ilona
Szerkesztette: Faklen Pál
Felelős vezető: Vékony Tamás
Nyomtatás: Sprint^R – Hungaroprint
Felelős vezető: Tóth Éva
Készült a Stoffel & Tsai. gondozásában

A könyvet mindazoknak ajánlom,

- akik most jutottak PC-hez és meg akarnak ismerkedni a különféle adattömörítő programok lehetőségeivel,
- akik már begyűjtöttek egy vagy több programot, de dokumentációt nem kaptak hozzá, vagy nincs idejük az angol leirással bíbelődni,
- akik jól kezelnek néhány tömörítőt, de azok rejtettebb szolgáltatásaival szintén meg akarnak ismerkedni,
- akik új tömörítőprogramokat szeretnének megismerni,
- akik a feladataikhoz legalkalmasabb tömörítő programot szeretnék kiválasztani,
- akik még csak most gondolkodnak azon, hogy tömörítő programot is érdemes lenne használni,
- akik az ismerősöknél, a BBS-ekben vagy a shareware könyvtárakban látott önkicsomagoló állományokhoz hasonló SFX-eket maguk is szeretnének készíteni,
- akik dugig megtelt winchesterükön egy kis szabad helyet szeretnének szorítani,
- akik úgy akarják raktározni bizalmas adataikat és programjaikat, hogy mások ne nézhessenek bele,
- akik szeretik anyagaikból a korábbi verziókat is megőrizni,
- akik programokat írnak és azokba tömörített biztonsági mentéseket terveznek,
- akik nagyobb beruházás és szerelés nélkül akarják megduplázni merevlemezük kapacitását.

Mottó #1

A DOS alatt működő PC-ken az eddigi leghatékonyabb tömörítési mód a DEL, vagy ERASE parancs.

Mottó #2

Ha egy tömörített állomány 0 bájt hosszú, az tovább már nem tömöríthető.

Bevezetés

Az adattömörítő programcsomagok mindegyikét aligha lehetne bemutatni, mert már követhetetlenül sok van belőlük. Közülük leginkább azokra igyekeztem figyelni, amelyek a legjobban sikerültek és szinte szabványt teremtettek, így összességében remélhetőleg átfogó képet tudtam adni a tömörítők kínálatáról és használatáról.

Az ismertetett programok egy része évek óta a shareware szabadszoftverek közé tartozik, mint például a PKARC, az LHARC különböző verziói, a PKZIP és függelékei. Ezeket bármelyik PC-s klubban megkaphatjuk vagy ismeretségi körünkben valakitől elkérhetjük. A többi is nagyon olcsón beszerezhető az immár Magyarországon is hozzáférhető shareware programkönyvtárakban (SolarSoft, Pannonsoft stb.). A winchesterbővítők (Stacker, Expanz!, stb.) kereskedelmi szoftverek és áruk miatt sajnos nem mindenki számára ilyen könnyen hozzáférhetőek, de a vállalati és hivatali PC-kben is egyre inkább szükség lesz rájuk.

A kötet remélhetőleg a PC-használók egyik kézikönyve lesz, amelynek későbbi kiadásai mindig követhetik a tömörítőprogramok újdonságait. Ehhez nagy örömmel felhasználom az olvasóktól beérkező észrevételeket, ötleteket, tanácsokat is.

Budapest, 1991. augusztus 21.

Nagy Gábor

Alapfogalmak, értelmezések

Írásmód

A tömörítő, kibontó és archívkezelő programokat az egyszerűség kedvéért a továbbiakban általában tömörítőknak nevezem. Ahol erre szükség van, ott természetesen különbséget teszünk közöttük, és egyértelműen jelzem, hogy mit kell használni. A tömörítő programok nevét a kiemelés érdekében a szövegben nagybetűvel írom, a példákban pedig kisbetűvel, bár a számítógépnek csak a programok különböző paramétereinél lehet fontos, hogy kis-, vagy nagybetűt használunk.

A tömörített, bepakolt, archivált állományok általános nevében mindig jelzem a tömörítő típusát, az önkicsomagoló állományok esetén is. Például: ZIP fájl, LZH fájl, ZIP-SFX fájl, LZH-SFX fájl, stb. A kiterjesztéseket nem írom be, a tömörítő alapértelmezésként ugyanis mindig a saját kiterjesztését veszi. Ahol eltérés van ettől, ott külön jelzem. A könyvben felváltva használom a fájl és magyar megfelelője, az állomány kifejezést.

Path

Magyarul elérési útvonal. Ha egy programról azt mondjuk, hogy path-on van, annyit jelent, hogy a DOS megtalálja, bárhol is hívjuk meg, s nem kell a programnév elé minden meghíváskor odabiggyeszteni az elérési útvonalat. A gyakran használt programokat, segédprogramokat, így a különböző tömörítő programcsomagokat is célszerű egy vagy több olyan könyvtárban (\ZIP, \ARC, \UTIL, \SEGED, stb.) elhelyezni, amelyet path-ra tettünk már az AUTOEXEC.BAT-ban a

```
PATH=c:\c:\dos;c:\util; ... , a SET PATH=c:\c:\dos;c:\util; ...  
vagy hasonló paranccsal.
```

Parancssor

Amennyiben a programokat DOS promptról hívjuk meg, azt a sort, amelyben a sor elején a prompt után a tömörítő nevét, a paramétereket és az archív nevét beírtuk, az angol command line megfelelőjeként parancssornak hívom. A paraméterek elnevezésében az eredeti leírások sem következettek. Én a főbb paramétereket parancsoknak, a parancsok végrehajtását befolyásoló paramétereket pedig opcióknak vagy kapcsolóknak nevezem.

Munkaterületek

A parancsot munkaterületekre is oszthatjuk. A legelső munkaterületen maga a tömörítő program neve található, amelynek tartalmaznia kell elérési útvonalát (path) is, ha az nincs az aktuális könyvtárban vagy egy path-on lévő könyvtárban.

A második munkaterület a parancs(ok) és kapcsolók helye. Egyes programoknál az egyszerre beírható parancsok, kapcsolók száma és azok parancssoron belüli helye szigorúan rögzített, máshol szabadon variálható.

A harmadik terület az archív fájl neve. Természetesen ez is tartalmazhat elérési útvonalat. Általános szabály, hogy becsomagoláskor nem adhatjuk meg dzsóker (* és ?) karakterekkel az archív nevét, ezen kívül a DOS általános névadási szabályai érvényesek. Kinyitáskor a legtöbb program megengedi a * és ? használatát az archív nevében is, de célszerű ezt az adott program ismertetőjében ellenőrizni. Az egyes tömörítők mindig saját kiterjesztésüket használják alapértelmezésként. Ahol ettől el lehet, vagy el kell térni, ott mindig jelzem.

A negyedik terület a célkönyvtár, ahova a kibontott állományok kerülnek, illetve bepakoláskor az a könyvtár, ahonnan a forrásfájlokat a tömörítő bepakolja. Egyes programoknál (PKZIP, PKARC) ez beépül a fájlnevbe, máshol (LHA, ARJ, PAK) élesen elkülönül.

Az ötödik munkaterület a fájl lista. Ez gyakran tartalmaz * és ? karaktereket. A fájl lista a legtöbb tömörítőnél megadható listafájl segítségével. Ez előtt többnyire egy @ karaktert kell írunk bevezető karakter gyanánt, az ARJ azonban ! jelet használ erre.

A hatodik területet tulajdonképpen nem is a tömörítő program, hanem a DOS használja: ez az átírányítás a >, illetve >> DOS operátorokkal pl. nyomtatóra (>PRN), lemezre (>fájl1.kit, illetve >>fájl2.kit, ha a meglévő állományt nem felülírni, hanem bővíteni akarjuk).

Bepakolás

Az az eljárás, amikor a forrásállományt a tömörítő program segítségével archív állománnyá alakítjuk át. A becsomagolás legtöbbször tömörítve történik, de néhány program felkínálja azt a lehetőséget is, hogy tömörítés nélkül készíti el az archívot.

Tömörítés

A tömörítő program megfelelően megválasztott módszerekkel a forrásállomány méretét adatvesztés nélkül lecsökkenti, miközben elkészít belőle egy archív állományt. Megfelelő kibontó programmal az eredeti állapot hiánytalanul visszaállítható.

Munkakönyvtár

Bepakoláskor a legtöbb tömörítő különleges átmeneti állományokat hoz létre, amelyeket az új archív létrehozása után letöröl. Ha nincs elegendő szabad lemezterületünk, a legtöbb tömörítőnek kijelölhetünk egy arra alkalmas átmeneti munkakönyvtárat is, lehetőleg a winchesteren vagy a ramdisk-en.

Környezeti változók (environmental variables)

Az alapértelmezés kialakításához a tömörítő programok majd mindegyike felhasználhat előzőleg definiált környezeti változókat. Célszerűen az AUTOEXEC.BAT-ban vagy egyéb batch fájlban adhatunk meg értéket. Egészen különleges változó az ERRORLEVEL, amelynek értékét a tömörítő programok a futás befejeztével módosítják, így téve lehetővé a biztonságos batch (köteget) alkalmazást.

Konfigurációs állomány

Egyes tömörítők saját konfigurációs állományt is képesek alkalmazni, miáltal komfortosíthatjuk a gyakran használt parancsok és kapcsolók alapértelmezésének beállítását.

Archív katalógus

Minden tömörített állományban van egy belső archív katalógus, amelyet a tömörítő program folyamatosan karban tart. Ebben szerepel minden adat, amely a becsomagolt fájl kibontásához szükséges, és gyakran még megjegyzéseket is tartalmazhat.

Listafájl

A listafájl egy egyszerű szöveges állomány, amely felsorolja a be-, illetve kipakolandó állományokat. Egyes programoknál kapcsolókat, sőt parancsokat is tartalmazhat. Rendszerint egy @ karakter vezet be, amelyről a tömörítő felismeri, hogy listaállománnyal van dolga.

Kibontás

A bepakolt állományok visszaállítása eredeti állapotukba. Gyakran még az eredeti (relatív, vagy abszolút) könyvtárszerkezetet is rekonstruálja az archívba elmentett adatok alapján.

SFX (self extract)

Önkinyitó tömörített archív állomány. Elindításakor kipakolja a benne tömörítve tárolt állományokat. Az SFX állományok használatát, előállítását, lehetőségeit, korlátait szintén az egyes tömörítő programokról és a haladóbbaknak szóló fejezetek ismertetik.

Általános fájl-tömörítők

A könyv túlnyomó része a fájl-tömörítő programokkal és azok segédprogramjaival foglalkozik. Nem véletlenül. Többször hivatkozom a könyvben egy hallgatóságos, iratlan tömörítési „szabványra”, amelyhez kötelessége igazodnia minden fájl-tömörítőnek. Szeretjük ugyanis, ha a program kiszolgál minket és jobban elfogadjuk az újat is, ha az a már (esetleg épp a konkurenciánál) megszokott parancsokat, definíciókat, szerkezeteket elfogadja és ugyanúgy kezeli, nem szorítva bennünket beidegződéseink „átprogramozására”.

Amelyik program tehát az alábbi parancsokat (esetleg kicsit eltérő írásmóddal, vagy a kezdőbetűkre rövidítve) nem ismeri, azokkal nem nagyon érdemes foglalkoznunk. Íme a szabvány értékű lista a fájl-tömörítők általános parancsaiból:

a	= Add	= Sima bepakolás, tömörítéssel, vagy anélkül.
f	= Freshen	= Frissítés (újat nem visz be).
u	= Update	= Aktualizálás (újat is bevisz).
m	= Move	= Áthelyezés, a bevitt forrásállományok törlése.
d	= Delete	= Fájl törlése az archívból.
e, x	= EXtract	= Kibontás, visszaállítás.
t	= Test	= Az archív épségének tesztelése. (A Hyper nem ismeri.)
v	= View	= Az archívkatalógus kifrása.
l	= List	= Az archívkatalógus kifrása.
c	= Console	= Megjelenítés képernyőn.
p	= Print	= Megjelenítés printeren.

A parancsok helye, mint a bevezetőben is említettem, többnyire a tömörítő neve és az archív fájl neve közötti, úgynevezett második munkaterület. Néhány program ennél rugalmasabb, azok parancsai a parancssorban bárhol lehetnek.

A parancsok előtt egyes programoknál egy bevezető -, vagy / karaktert kell írunk, más programoknál a parancs bevezető karakter nélkül áll, csak a kapcsolók elé kell a fenti bevezető karakterek egyikét beírni.

Emellett egy jó program biztosítja megjegyzések hozzáfűzésének lehetőségét (komment, archív komment), és képes önkinyitó SFX (= Self eXtract) archívot is létrehozni. Ezekon kívül egy valamirevaló program legalább egy valamiben többet tud társainál. Az élmézőnynek pedig több területen is kimagasló képességekkel kell rendelkeznie.

Az egyes programcsomagok vizsgálatakor a következő menetet és szempontokat igyekeztem követni:

- Bepakolás.
- Kipakolás, teszt, nyomtatás.
- File törlése az archívból, archív karbantartás.
- Megjegyzések, jelszavas védelem.
- Archívkatalógus.
- Önkinyitó SFX archív készítése és használata.
- Parancsok, kapcsolók összefoglalása.
- Listafájl.

Konfigurációs állomány.

Környezeti változók alkalmazása.

Batch alkalmazás.

Speciális szolgáltatások, a csomag segédprogramjai.

Gyakorlati példák a gyakrabban alkalmazott műveletekre.

A fájl-tömörítő programcsalád képviselői a kiválasztott állományokat, ha kérjük, a könyvtárszerkezettel együtt egy tömörített archív állományba tudják bepakolni. Bepakolásakor ezernyi lehetőségünk van a fájl kiválasztására, törölhetünk, áthelyezhetünk (move), aktualizálhatunk, frissíthetünk, jelszóval titkosíthatunk, kommentezhetünk fájlokat. A kész archívban van egy jól használható tartalomjegyzéke (archív katalógus), amely rendkívül nagy segítség az adatállományok és programok nyilvántartásában.

Az így előállított tömörített archív állományokkal szinte azt csinálunk, amit csak akarunk. Természetesen az archívokból ki lehet pakolni akár mindent, akár csak a kiválasztott állományokat. Lehet bővíteni, aktualizálni, belekukkantani, törölni, nyomtatni, ellenőrizni, stb. És az önkicsomagoló (SFX) állományokká alakításról se feldkezzünk meg.

Az egyes programok, programcsomagok mind rendelkeznek valamilyen teljesen egyedi, extra szolgáltatással, emellett a hagyományos archívkezelő funkciókat magas színvonalon biztosítják. Az extra szolgáltatások felsorolását lásd később, a program kiválasztásáról szóló részben.

A fájl-tömörítők előnye a jó tömörítési hatások, és az, hogy a létrehozott archív állományok könnyebben utaztathatók lemezekre és különféle kommunikációs csatornákon.

A fájl-tömörítők hátránya, hogy aktualizálásakor, vagy használat előtt mindig ki kell nyitni az archívba pakolt állományokat, s ha átmenetileg is, de ilyenkor duplán foglalnak helyet a lemezen.

Visszaüt a tömörítőprogramok sokfélesége, mert néha nehéz, vagy hosszadalmas konvertálni az adatokat a különféle archív formátumok között, s könnyen előfordul, hogy a szükséges tömörítő, kicsomagoló, konvertáló segédprogramot nem ismerjük, vagy nincs elérhető helyen. Szerencsére a különféle tömörítők parancskészlete és szintaktikája az egységesítés irányába halad, s aki egy tömörítő használatát legalább elemi szinten elsajátította, az a többivel is nagyon gyorsan megbarátkozik.

Az elterjedtebb tömörítők

Az egyik legkisebb méretű, s egyben a legnagyobb tömörítést elérő program sokáig az LHA.EXE 2.xx volt. (Most is csak az ARJ tud nála is tömörebb archívokat előállítani.) Paraméterezése az összes tömörítő közül a második legegyszerűbb, mindössze 12 parancsot és 15 kapcsolót kell a program használatához megjegyeznünk. Kezdőknek ideális archiváló program. Nem hipergyors, de a tömörség és egyszerűség a legtöbb felhasználónak sokkal fontosabb erény. SFX állományai igen jók.

A NoGate cég PAK programja meglehetősen terjedelmes, de igen sokat tud. Közvetlenül az LHA után következik tömörítésben, adatkonverziós lehetőségei pedig egyedülállóak. Parancsait mindenki saját igényeihez illesztheti. Állományokat áthelyezni a forrásállományok automatikus törlésével (move) a PAK-nál nemcsak befelé, hanem kifelé is lehet, vagyis a kikapolt állományokat egyetlen paranccsal egyúttal törölni is tudjuk az archívból. Hálózatokon használva a PAK program a szerver meghajtókra nem személtel, oda legfeljebb a végleges állományokat teszi.

A PKZIP együtt nyújtja a gyorsaságot, a nagyfokú tömörítést és az adatbiztonságot. Extra szolgáltatásai, főleg a nyomtatás és a sérült állományok helyreállítása területén különösen jók. Nyomtatásban pedig jelenleg verhetetlen.

Robert K. Jung ARJ programja meglehetősen friss még, de nagy jövő előtt áll. Igaz ugyan, hogy az élmezőnyön belül a lassabbak közé tartozik, de tömörítő képessége és extra szolgáltatásai kiegyenlítik ezt az apró hátrányt. Ahogy a szerző írja, a többrészes archívok révén ez lehet a „szegény ember backup programja”.

A PKPAK a jó öreg PKARC utóda. Egy kicsit elavult már, de még rengetegen használják, és igen sok .ARC állomány kering ma is közkézen. Alapprogramnak számít. Szolgáltatásai néhány kivételtől eltekintve megegyeznek a PKZIP csomagéval.

A ZOO három tulajdonsága miatt főleg a fejlesztők számára lehet érdekes: maximum 64 K hosszú megjegyzéseket is elfogad, állomány-generációkat tárolhatunk egymás mellett, és mellékelték a program forráskódját is.

Az ARC 6.02 minden tesztben hátul végzett. Méltatni — tiszteletre méltó korán kívül — csak azért érdemes, mert archívból tudja futtatni az általa bepakolt .EXE és .COM állományokat.

A DWC 5.01 szintén egy kiveszőben lévő tömörítő, de mivel tőlünk nyugatabbra még jó páran használják, érdemes egy keveset foglalkozni vele.

A HYPER 2.50 programon sokáig töprengtem. Maga a program egy „mélynövésű óriás” a tömörítők között. Kisebb az LHA-nál, sőt kezelése is egyszerűbb, tömörítésben pedig a PKZIP-pel is felveszi a versenyt. Ezzel azonban szemben áll, hogy csak a „szabvány minimumot” tudja, semmi extra szolgáltatása nincs.

A további fejezetekben ismeretett fájl-tömörítők és a róluk kialakult értékítéletem röviden:

Programcsomag	Verzió- szám	Alkotó (Cég)	Vélemény
PKZIP & PKUNZIP	1.10-ig	Phil Katz (PK, PKWare)	*****
LHA/LHICE/LHARC	2.12-ig	Haruyasu Yosizaki (Yoshi)	*****
ARJ	2.10-ig	Robert K. Jung	*****
PAK	2.50-ig	(NoGate Consulting)	*****
HYPER	2.50-ig	P. Sawatzki — K.P. Nischke	****
PKLITE	1.03-ig	Phil Katz (PK, PKWare)	***
ZOO	2.01-ig	Rahul Deshi	**+
LZEXE	0.91-ig	Vernon Bueg	*
PKPAK & PKUNPAK	3.61-ig	Phil Katz (PK, PKWare)	*+
PKARC & PKXARC	3.60-ig	Phil Katz (PK, PKWare)	*+
ARC	6.02-ig	(SEA)	+
DWC	5.01-ig	Dean W. Cooper	*+

(A sok ***** jó véleményt jelent, a + jel az elavult programok mellett áll.)

Tömörítők tesztje

1. *.DOC állományok (43 db, 429 236 bájt) tömörítése:

Program (+ paraméter)	Idő (sec)	Tömörített méret		Lemaradás
		(bájt)	(%)	(%)
ARJ a	90	176 647	41,15	—
LHA a	113	178 964	41,69	1,31
PAK a	94	188 840	43,99	6,90
PKZIP -aex	60	191 232	44,55	8,27
PKZIP -aei	61	191 390	44,59	8,35
PKPAK -a	37	226 061	52,67	27,97
PKZIP -aes	35	226 346	52,73	28,13
ZOO -a	81	231 387	53,91	30,99
ARC a	64	244 452	56,95	38,38

2. *.EXE és *.COM állományok (21 db, 735 700 bájt) tömörítése:

Program (+ paraméter)	Idő (sec)	Tömörített méret		Lemaradás
		(bájt)	(%)	(%)
ARJ a	151	465 670	63,30	—
LHA a	194	480 291	65,28	3,14
PKZIP -aex	106	502 155	68,26	7,83
PKZIP -aei	107	502 155	68,26	7,83
PAK a	162	504 386	68,56	8,31
PKPAK -a	75	605 344	82,28	29,99
PKZIP -aes	71	609 677	82,70	30,92
ARC a	127	621 501	84,48	33,46
ZOO -a	148	628 298	85,40	34,92

3. *.DBF állományok (5 db, 218 798 bájt) tömörítése:

Program (+ paraméter)	Idő (sec)	Tömörített méret		Lemaradás
		(bájt)	(%)	(%)
ARJ a	30	14 229	6,53	—
LHA a	39	14 802	6,77	3,57
PKZIP -aex	78	17 208	7,86	20,40
PKZIP -aei	78	17 208	7,86	20,40
ARC a	15	20 355	9,30	42,42
PAK a	27	20 998	9,60	46,92
PKPAK -a	10	21 206	9,69	48,38
ZOO -a	18	21 655	9,90	51,52
PKZIP -aes	11	21 699	9,92	51,83

4. *.WK? állomány (1 db, 74118 bájt) tömörítése:

Program (+ paraméter)	Idő (sec)	Tömörített méret		Lemaradás
		(bájt)	(%)	(%)
ARJ a	11,6	48 403	65,30	—
LHA a	10,9	50 337	67,91	4,00
PAK a	9,0	51 372	69,31	6,13
PKZIP -aex	5,6	54 208	73,13	11,99
PKZIP -aei	5,6	54 208	73,13	11,99
ZOO -a	7,4	69 317	93,52	43,21
PKPAK -a	6,7	69 513	93,79	43,61
PKZIP -aes	6,6	70 417	95,01	45,48
ARC a	7,5	71 116	95,95	46,92

**5. Grafikus állományok
(EPS, PCX, TIFF, IMG) tömörítése
(7 db, 2 959 535 bájt):**

Program (+ paraméter)	Idő (sec)	Tömörített méret		Lemaradás
		(bájt)	(%)	(%)
ARJ a	1110	1 240 864	42	—
LHA a	630	1 251 690	42	0,87
PAK a	725	1 302 225	44	4,95
PKPAK -a	207	1 331 652	45	7,32
PKZIP -aes	213	1 346 188	46	8,49
ZOO -a	332	1 370 265	47	10,43
PKZIP -aex	784	1 375 567	47	10,86
ARC a	326	1 454 546	50	17,22

**6. Tömörített állományok továbbtömörítése
(1-1 ARC, PAK, ZIP, ZOO, LZH, ARJ állomány,
209 083 bájt):**

Program (+ paraméter)	(bájt)	Tömörített méret (%)
LHA a	202 801	97
ARJ a	202 802	97
PAK a	207 375	99,18
PKZIP	209 083	100
PKPAK -a	209 277	100,01
ARC a	209 375	100,14
ZOO -a	218 612	104,56

**7. Vegyes állományok tömörítése
(23 file, 748 671 bájt):**

Program (+ paraméter)	Idő (sec)	Tömörített méret		Lemaradás
		(bájt)	(%)	(%)
ARJ a /m1 /jm /jh65000	196	475 036	63,45	—
ARJ a	168	476 994	63,71	0,41
LHA a	150	481 310	64,29	1,32
PAK a	144	497 882	66,50	4,81

Program (+ paraméter)	Idő (sec)	Tömörített méret (bájt)	Tömörített méret (%)	Lemaradás (%)
PKZIP -aex	90	497 074	66,39	4,64
PKZIP -aes	69	564 958	75,46	18,93
PKPAK -a	80	565 538	75,54	19,05
ARC a	138	578 448	77,26	21,77
ZOO -a	168	595 411	79,53	25,34

A teszt eredményéhez néhány megjegyzés:

— A PKZIP-hez hasonlóan az ARJ is rendelkezik olyan lehetőségekkel, hogy a felhasználó írja elő, milyen tömörítési technikát alkalmazzon a program.

— A Huffman puffer méretének beállítása nemcsak gyorsít az ARJ programon, hanem az ARJ állomány még tömörebb is lesz pár százalékkal. Így az LHA-t is megelőzi.

— Már megjelent az ARJ 2.xx verziója is, ami még tömörebb archívokat eredményez, ráadásul tovább gyorsult.

A végeredmény alapján a leginkább ajánlott programok a következők:

1. ARJ 2.10
2. LHA 2.12
3. PAK 2.51 — PKZIP 1.10

A ZOO és az ARC program használata csak akkor ajánlható, ha azok különleges szolgáltatásaira van szükség. Mind tömörítésben, mind sebességben, mind az általános szolgáltatásokban alaposan lemaradnak. A PKARC és PKPAK pedig már elavult, érdemes helyettük áttérni valamelyik hatékony tömörítőre.

A DWC és a HYPER programokat csak a tesztek lezárása után kaptam meg. Az itt nem szereplő póttesztek szerint a DWC a PKPAK/PKARC, a HYPER pedig a PKZIP kategóriájába tartozik, mind a sebéséget, mind a tömörítést tekintve.

A legsikeresebb — a PKZIP

Phil Katz és a PKWare, Inc. neve nem lehet ismeretlen az előtt, aki már egy kicsit is beleszagolt a PC-k világába. A zseniális programozó és csapata hosszú éveken keresztül erős konkurenciát jelentett a tömörítő programok egyik fő fejlesztőjének, a SEA (System Enhancement Associates) cégnek. A PKWare által készített becsomagoló, kipakoló és archívkezelő programok mindig egy kicsivel, néha nagyon sokkal többet tudtak, és gyorsabbak voltak mint az ARC. Végül a SEA megelégtelt, hogy szakmailag nem tud a PKWare fölé kerülni, s egy azóta talán ezerszer megbánt lépésre szánta el magát: beperelte ellenfelét. A per vége ismert, a PKWare cégnek abba kellett hagynia a PKARC csomag forgalmazását.

Az igazi győztes azonban a vesztes lett, mert bár a bíróságon vereséget szenvedett, de a vereségből is erőt merített. A PKWare programozói létrehoztak egy új tömörítő családot, a PKZIP-et. A program és az új archív forma pedig elindult világhódító útjára. Csak négy program van, ami a PKZIP-nél is tömörebb állományokat hoz létre, az LHA (az LHARC és LHICE utóda), az ARJ, a HYPER, valamint a PAK új verziója. Azonban sebességben és az extrákban elmaradnak a PKZIP mögött.

A PKZIP mind sebességét, mind tömörségét, mind pedig extra szolgáltatásait tekintve azonnal az élre ugrott, s azóta is stabilan tartja helyét az első három között. A program szinte mindazt tudja, mint elődei (a PKARC és PKPAK), sőt még többet is, gyorsabban és tömörebben.

A PKARC-hoz és a PKPAK-hoz hasonlóan itt is határozottan szétválasztották a be- és kipakolást, valamint az SFX létrehozását, három külön-külön alkalmazható programba, így egyszerre kisebb programot kell betölteni és az gyorsabban is tud dolgozni, mintha egyetlen állományban lenne.

A programcsomag parancsai megfelelnek a tömörítő programok előzőekben ismertetett egyezményes jelrendszerének, csupán a PKZIP programban van néhány eltérő jelentésű:

e	= Extra compression	= A tömörítési módszer jelölése.
t	= Timestamp	= A fájl keletkezésének figyelése.
c	= Comment	= Megjegyzések hozzáfűzése.
l	= Licence	= Licenc képernyő.
p	= Path	= Path információk tárolása.

Az adatbiztonságról

1. A leggyakoribb hibaforrás az ember. Egy kis odafigyelés sokat segíthet.
2. A Move parancs, illetve kapcsoló használata esetén csak akkor törli a program a forrásállományokat, ha sikeresen és hibaüzenet nélkül be tudta pakolni az állományokat az átneveti ZIP fájlba (.!!! kiterjesztéssel), majd sikeresen létre tudta hozni az új, ZIP kiterjesztésű archívot.
3. A véletlen felülírásokat megfelelő parancsok alkalmazásával kivédhetjük.
4. A PKZIP 32-bites CRC értékkel ellenőrzi az állományok épségét.
5. A PKZIPFIX programmal a sérült ZIP fájlokból ki lehet ollózni az épen maradt részeket.

A.) Bepakolás

A tömörített ZIP állományokat a PKZIP programmal tudjuk létrehozni. Az állományok bepakolására a négy szabványos parancs szolgál, valamint egy új, az i (selective backup), melyek közül az a, u, i és m parancsokkal új archívokat is készíthetünk, nemcsak a már meglévőket lehet bővíteni.

-a (= Add)

Mindenképpen bepakolja a megadott állományokat az ZIP fájlba, függetlenül attól, volt-e már az archívban azonos nevű állomány és mikori keltezéssel. A már meglévő ilyenkor szó nélkül felülírja az újjal. Két azonos nevű állományt csak akkor pakol be a PKZIP program a ZIP fájlba, ha azokat eltérő path-ról, teljes névvel pakoltuk el (lásd még p, P és r parancsokat ugyanitt).

-u (= Update)

Az Add parancshoz hasonlóan működik, de megvizsgálja azt is, hogy van-e már a bepakolásra kijelölttel azonos nevű állomány a ZIP fájlban. Ha igen, csak abban az esetben írja felül a már bent lévő, ha a meglévő fájl régebbi keltezésű, mint a bepakolásra kijelölt állomány. Ez a dátumellenőrzés lekapcsolható, s ilyenkor az Add parancsnak megfelelően dolgozik a PKZIP.

-f (= Freshen)

Csak és kizárólag frissítést végez. Az Update parancshoz hasonlóan csak a már bent lévőnél frissebb keltezésű állományokat pakol be, új állományokat azonban nem épít be az archívba. Értelemszerűen új archív létrehozására sem alkalmas. Használati területe a munkaállományok tömörített biztonsági másolatainak mindennapos felfrissítése a munka végeztével.

-m (= Move)

Megspórolhat vele egy csomó felesleges munkát. Az Add parancshoz hasonlóan minden kijelölt állományt keltezésre való tekintet nélkül bepakol, majd az eredetiket (ha nem történt hiba a tömörítés és a ZIP fájl kiírása közben) kitörli, természetesen a Readonly attribútumú állományok kivételével.

-i (= selective backup)

Egyedülálló a tömörítő programok körében. Csak és kizárólag azokat az állományokat pakolja be az archívba, amelyek még nem szerepelnek abban. Így tökéletesen kiegészíti a Freshen parancsot.

Azt, hogy a program lássa, vagy ne lássa a rejtett és rendszerállományokat, be tudjuk állítani a később ismertetendő konfigurációs állomány, vagy a w és W parancsok (lásd a különleges szolgáltatásoknál) segítségével. A program a DOS-hoz hasonlóan alapértelmezésként NEM látja a rejtett (hidden) és rendszerállományokat (system), így használható vele a NO.COM program is (ismertetése megtalálható a könyv segédprogramokkal foglalkozó fejezetében), de célszerűbb az x (= eXclude) parancsot használni (lásd a program különleges szolgáltatásainál). Azt is befolyásolni tudjuk, hogy a bepakolt állományok attribútumait maszkolja-e a program (lásd később a j és J parancsot, ugyancsak a különleges szolgáltatásoknál).

A fenti alapparancsok közül a három első használható az `m` (Move) kapcsolóval kombinálva is. Az `am` parancs (Add + Move) teljesen egyenértékű az `m` (Move) parancssal. A `Freshen` és az `Update` parancsoknak a `Move` kapcsolóval való bővítése (`mf`, illetve `mu`) azt eredményezi, hogy a bepakolásra kijelölt állományok az új ZIP fájl sikeres létrehozását követően törlődnek, ha csak nem voltak `readonly`-nak jelöltek. Csak azok az állományok törlődnek a `Move` parancs használatakor, melyeknél ténylegesen megtörtént a bepakolás. Figyelem! azok az állományok, amelyeket az `mf`, vagy `mu` parancs nem frissített, mivel régebbiek voltak a már bennlévőknél, nem törlődnek az új ZIP fájl létrejötte után.

-\$ (store volume label)

A `PKZIP` nemcsak az állományokat tudja eltárolni, hanem a lemez nevét is, ha kiadjuk a `-$<meghajtó:>` parancsot is.

-p vagy -P (= Path)

Az egyes bepakolt állományok relatív elérési útjait is eltárolja a filenevekben. A `-p` együtt használatos az alább ismertetett `-r` parancssal. Kinyitáskor a `-d` parancsot használva a `PKUNZIP` mindent a helyére rak, sőt a relatív könyvtárszerkezetet is létrehozza. A `-p` kikapcsolja a path információk tárolását, ha a `PKZIP.CFG` konfigurációs állományban (lásd később részletesen) alapértelmezéssé tettük az elérési utak valamilyen szintű tárolását. Az így megőrzött név relatív, azaz a megadott forráskönyvtárból induló alkönyvtárakat őrzi meg, magát a parancssorban megadott forráskönyvtárat nem. Ha nagybetűvel írjuk be a parancsot, akkor a parancssorban megadott elérési út is a név része lesz, az abból nyíló alkönyvtárakon kívül. Célszerű olyankor használni, ha rögzített könyvtárstruktúrával operáló programcsomag állományait archiváljuk, vagy amikor több azonos nevű, de különböző könyvtárakból származó állományt kell az archívban elhelyeznünk.

-r (= Recurse)

A megadott könyvtár szerkezetét tárolja el az archívban.

Lemezeken végzett archív frissítéskor könnyen ütközhetünk a lemez szűk voltának korlátjába. Ez legtöbbször kivédhető a `-b` parancs alkalmazásával (lásd részletesen a speciális szolgáltatásoknál).

B.) Kibontás

A `PKZIP` párja, a `PKUNZIP` gondoskodik a ZIP állományok kibontásáról, nyomtatásáról. Ha már létrehoztunk vagy megkaptunk egy ZIP fájlt, ki is akarjuk néha csomagolni. A kibontás iránya, célja szerint is több csoportra oszthatjuk a kipakoló parancsokat. Ezek a következők:

Egyszerű kipakolás

-e és az -x (= EXtract)

A parancs két alakja teljesen egyenértékű. Egyszerűen kipakolnak mindent az archívból az adott maszknak megfelelően a parancssorban megadott könyvtár(ak)ba. Természetesen különböző kiegészítő parancsokkal módosíthatjuk működését. Mind az `e`, mind az `x` parancs használatakor rákérdez a `PKUNZIP`, hogy felülírjon-e, ha egy olyan állományt akar kinyitni, ami a célkönyvtárban már szerepel. Ez az üzenet elnyomható, ha biztosak

vagyunk abban, hogy tényleg az archívból kinyitható állományokra van szükségünk. Az EXtract parancs saját módosítói:

- c A CRC értékek sorrendjében bont.
- d Dátum szerinti sorrendben bont.
- e Kiterjesztések szerint, ábécé rendben bont.
- n Nevek ABC sorrendjében bont.
- p Tömörítettség szerint bont.
- s Méret szerint bont.

-o (= Overwrite)

Akkor használatos az EXtract parancs mellett, ha a célkönyvtárban azonos néven már létező állományokat feltétlenül felül akarjuk írni a ZIP fájlból kibontandó archivált állománnyal. Ilyenkor kérdezés nélkül is felülír.

-f (= Freshen)

Megfelel a bepakoláskor is használt Freshen parancsnak, azaz csak azokat az állományokat bontja ki, amelyek mind a célkönyvtárban, mind a ZIP fájlban azonos néven léteznek. Újat nem bont ki.

Átnézés, nyomtatás

-c (= Console)

A PKUNZIP Console parancsa a képernyőre nyitja ki a fájlokat. Természetesen csak szövegfájloknál célszerű használni. Az egy képernyőnél hosszabb szövegek átnézésére rendelkezésünkre áll a m (MORE) alparancs is. A -cm egyszerre egy képernyőnyit jelenít meg a ZIP fájlba bepakolt szövegfájlból. Ilyenkor a következő mozgási lehetőségünk van:

- <SPACE> A következő képernyőnyi anyagot mutatja meg.
- <ENTER> Egy sorral tovább megy.
- <ESC> A következő bemutatandó állományra ugrik.

-p (= Print)

Nyomtatóra küldi a kinyitott fájlokat, nem pedig lemezre. Szöveget és rajzot egyaránt tudunk így nyomtatni közvetlenül a ZIP fájlból. Nem lehet a parancs módosító paramétereit szétválasztani, csak elhagyni a feleslegeseket. A további paraméterek nélkül kiadott -p parancs nem kapcsol a nyomtatón más üzemmódra. A Print parancs a rendelkezésünkre álló kísérő paraméterekkel a következő lehet:

-p[a][b][c][n], ahol

-pa A nyomtatón szöveges üzemmódba kapcsolva nyomtat. Az egyes állományok végén lapot dob.

-pb A nyomtatón bináris (grafikus) módba kapcsolva nyomtat. Nem dob lapot az egyes fájlok után.

-pc Az egyik COM portra küldi a nyomtatnivalót. Ha elhagyjuk, alapértelmezése az LPT1.

Egy soron belül nem keverhető a -pa, -pb és a -pc parancs! Az n lehet 1 és 4 közötti szám. Ez a port száma, ahova a nyomtatnivalót küldi a PKUNZIP.

Egy példa:

-pcb1 A COM1 portra küldi a nyomtatnivalót, miután binárisra (grafikusra) kapcsolta a printer üzemmódját.

Az ZIP fájl épségének ellenőrzése

-t (= Test)

Az archív épségét ellenőrzi azáltal, hogy látszólag kibontja az archívot, de nem írja lemezre a kibontott állományokat, „kipakolás” közben pedig egy CRC ellenőrzést végez. A sérült ZIP fájlok tartalmának még épen maradt részét a legtöbbször meg lehet menteni a PKZIPFIX kiollózó programmal (lásd később a speciális szolgáltatásoknál).

A tesztelést a PKUNZIP programmal kell végeznünk! A PKZIP-nek is van -t parancsa, de az egészen más feladatot lát el (lásd a speciális szolgáltatásoknál).

C.) Fájl törlése az archívból

Az ZIP fájlokból a szabványos -d (= Delete) parancs segítségével törölhetünk ki állományokat. Ilyenkor a PKZIP egy új ZIP fájlt hoz létre, amelyből egyszerűen kihagyja a törlésre kijelölt állományokat. A Delete parancs használatakor nem írhatunk a ZIP fájl nevébe dzsóker karaktereket, és legalább egy törlendő állomány nevét meg kell adnunk (ezekben már engedélyezett a dzsóker karakterek használata).

A Delete parancs a PKZIP csak néhány parancsával használható egyszerre, így a -b és az -x paranccsal.

D.) Megjegyzések, jelszavas védelem

-z (= Zipcomment)

A ZIP fájlhoz maximum 127 karakteres megjegyzés írható közvetlenül a PKZIP-pel. A ZIPcomment megjelenik a ZIP fájl neve után, ha az állományt PKZIP-pel, vagy a PKUNZIP-pel kezeljük. Ékezetes betűk is bevihetők. A régi ZIPcomment felülíródik.

-c (= Comment)

A PKZIP-pel maximum 59 karakteres megjegyzés fűzhető közvetlenül az egyes bepakolt fájlokhoz. A már létező komment felülírható (ha legalább egy szóközt beadunk, akkor a régi törlődik). Ékezetes betűk is bevihetők. A Comment egyike azon négy PKZIP parancsnak, ahol a kis- és nagybetűk között különbséget tesz a program. Ha kisbetűvel használjuk, akkor a már meglévő megjegyzések módosulnak, illetve új megjegyzéseket fűzhetünk hozzá. Ha nagybetűvel írjuk, akkor a már meglévő megjegyzésekkel nem foglalkozik, azokat változatlanul hagyja. Ha azonban a -C az egyetlen kiadott parancs, akkor hatása teljesen megegyezik a -c-vel.

A ZIP kommentről és a fájlkommentekről bővebben lásd később.

-s (= Scramble)

Jelszó megadásával titkosíthatjuk a fájlt. Ha elfelejtjük a jelszót, akár ki is törölhetjük a titkosított állományt a ZIP fájlból, mert csak a jelszóval férhetünk hozzá. A jelszó maximum 32 karakteres lehet és szóközt nem tartalmazhat. A Scramble parancs különbséget tesz a kis- és nagybetűk között. Az -s parancs és a jelszó között sem lehet szóköz. Az 1.10 verziótól kezdve már az önkinyitó állományban is lehet jelszóval védett fájl, mert az SFX is rendelkezik a Scramble parancs alkalmazásának lehetőségével.

E.) Az archív katalógus megnézése

A PKZIP, mint minden fájl-tömörítő program, külön katalógust vezet a tömörített (archív) állományba bepakolt fájlokról. Ezt a listát nevezem archív katalógusnak. Ebben a következő bejegyzések szerepelnek: fájlnev, eredeti hossz, alkalmazott tömörítési módszer, tömörített méret, százalékos méretcsökkenés, keletkezés dátuma, ideje, 32 bites CRC, attribútum, megjegyzés, titkosított-e, mivel készült. Ez az archív katalógus a `-v` paranccsal és annak paramétereivel hívható elő. A PKZIP és a PKUNZIP `-v` parancsai két kivételtől eltekintve megegyeznek, ezért együtt tárgyalom őket. A kivétel a `-vc` parancs értelmezése (lásd a `-v` parancsok ismertetésének a végén) és a `-vt` parancs.

A `-v` (View technical information) parancs

Technikai információk, archív katalógus megjelenítése a parancs feladata. Rovatai:

Length	A fájl eredeti hossza.
Method	A fájl-tömörítésére használt módszer.
Size	A tömörített fájl mérete.
Ratio	A méretcsökkenés az eredeti fileméret százalékában.
Date	A fájl aktuális dátuma.
Time	A fájl aktuális ideje.
CRC-32	A fájl 32 bite CRC értéke.
Attribute	Az eredeti fájlattribútum (h=rejtett, s=rendszer, w=írható, r=csak olvasható, *=titkosított).
Name	Fájlnev a kiterjesztéssel, esetleg a path is.

A felsorolás végén egy aláhúzás után még megjelenik az összesített eredeti méret, a ZIPelt méret, a ZIP fájlban lévő fájlok száma, és az átlagos méretcsökkenés.

A parancs általános formája és megadható paramétereai:

- `-v[b,r,t][c,d,e,n,o,p,s]`
- `-vb` A megjelenített adatok közül kimarad a CRC és az attribútum.
- `-vr` Fordított sorrendben jelennek meg a fájlok.
- `-vt` Részletesebb technikai információk jelennek meg egymás után az egyes állományokról:

Filename	A fájl neve.
File type	A fájl típusa (bináris, vagy text).
Encrypted	Itt jelzi a program, ha az adott fájl titkosított.
Attribute	Fájl attribútum (h=rejtett, s=rendszer, w=írható, r=csak olvasható).
Comments	Fájl megjegyzések.
Date and Time	A ZIP fájl dátuma és ideje.
Compression Method	A fájl-tömörítésre használt módszer (Imploding, Shrinking, Reducing vagy Storing).
Compressed Size	Tömörített fájl mérete.
Uncompressed Size	Az eredeti fájl méret.
CRC-32	A numerikus CRC-32 érték.
Created by	Melyik PKZIP verzióval készült.
Need to extract	Melyik PKUNZIP verzióval lehet kinyitni.

- `-vc` A meglévő fájl kommentek kijelzése (PKZIP).
- `-vd` Listázás a legrégebbtől a legújabbig.
- `-ve` Listázás kiterjesztés alapján, ábécé sorrendben.
- `-vn` Listázás fájlnev alapján, az ábécé sorrendben.

- vo Listázás a bepakolás sorrendjében.
- vp Listázás a tömörítési százalékok szerint, 0—99.
- vs Listázás fájl méret szerint.

A PKUNZIP a -vc parancsot másként értelmezi, mint bepakoló párja a PKZIP. Míg a PKZIP a -vc parancsra a megjegyzéseket is kijelzi, addig a PKUNZIP ugyanerre a parancsra egyszerűen a CRC értékek alapján rendezve jeleníti meg a bepakolt állományokat. A -vt-t nem értelmezi a PKUNZIP.

A ZIP fájl és az archív katalógus fizikai és logikai szerkezetével ebben a kötetben nem foglalkozom. A részletes ismertetés csak a tervezett második kötetbe fér bele, amelyben többek között ismertetem az első kötetből hely és időhiány miatt kimaradt, valamint az újabban beérkezett programokat, továbbá a tömörítés matematikai alapjait és a fontosabb tömörítő algoritmusokat, ahol lehet, forráskódokkal bemutatva.

Az önkinyitó állományok archív katalógusa csak a PKZIP, vagy a PKUNZIP programok, illetve a listázó segédprogramok (MAD, FV, ZIPVIEW, stb.) segítségével nézhető meg, mert az SFX igen színvonalas szolgáltatásai közül a -v parancs kimaradt.

F.) Önkinyitó (SFX) archívok

A PKZIP külön segédprogramot használ az SFX archívok létrehozására. Ez a ZIP2EXE.EXE program. Kezelése meglehetősen egyszerű, csak néhány apróságot kell észben tartani. A korábbi PKZIP verzióknál még a PKSFX.PRG-t is elő kellett állítani a PKZXXX.EXE törzsállományból a MAKESFX programmal, s azt a ZIP2EXE mellé kellett másolni. A ZIP2EXE.EXE a PKZIP 1.10 verziótól kezdve többé már nem igényli a PKSFX.PRG jelenlétét.

Az SFX létrehozásához egy előzőleg elkészített ZIP fájl kell. A ZIP2EXE használatának szintaxisa:

ZIP2.EXE [-parancs] ZIP_fájl

A ZIP fájl változatlanul megmarad, de lesz egy azonos nevű SFX fájl is, EXE kiterjesztéssel.

Ha -j paramétert használunk, akkor csak IMPLODING módszerrel pakolt fájlok lehetnek a ZIP fájlban s a kinyitáskor nem adhatunk parancsot, mindent egyszerűen csak kinyit a céllemez célkönyvtárába. Cserébe viszont kisebb lesz az SFX fájl. Az alapértelmezés a normál SFX fájl készítése.

Az OS/2-t használóknak a -f parancs biztosítja, hogy olyan EXE fájlt hozzon létre a ZIP2EXE, ami mind OS/2, mind MS-DOS alatt fut. Ha ezt nem adjuk meg a parancssorban, akkor a program egy OS/2 protected módban működő EXE fájlt csinál.

Amennyiben elfelejtünk ZIP fájlnevet adni, vagy rossz parancsot adunk meg, a ZIP2EXE Help képernyője jelentkezik be.

A kész ZIP-SFX saját parancsai

A ZIP-SFX állományok normál ZIP fájlként is használhatók, de ilyenkor az archív EXE kiterjesztését is be kell írni. Egyszerűbb azonban az SFX saját parancsainak használata, amelyek megegyeznek a PKUNZIP parancsaival (csak a -v parancs hiányzik az SFX parancskészletéből).

SFX_fájl [-parancsok] [céllkönyvtár] [lista]

Az önkinyitó ZIP fájlok parancsai, ha nem -j opcióval hozták létre:

- x és -e Kibont (alapértelmezés). Ha azonos nevű fájlt talál a célkönyvtárban, akkor rákérdez, felülírja-e.

- c Képernyőre nyit, mint a PKUNZIP esetén.
- cm Képernyőre nyit, mint a PKUNZIP esetén, a MORE paranccsal képernyőnként lapozgatva.
- d A célkönyvtárban létrehozza a ZIP fájlban eltárolt relatív könyvtárszerkezetet.
- n Csak azokat a fájlokat bontja ki, amelyek frissebbek a célkönyvtárban lévőknél.
- o Mindent kibont és kérdezés nélkül felülír.
- t Az SFX fájl épségét ellenőrzi.
- sjelszó Az 1.10 verziótól az SFX is támogatja a jelszavas védelmet.
- l Licenc képernyő
- h Help képernyő. Akkor is jön, ha nem értelmezhető parancsot kap.

A -j opcióval létrehozott SFX fájl csak a -o parancsot értelmezi. Mérete is kisebb, és legfeljebb 500 fájl lehet benne.

Tippek az SFX fájlok használatához

Álljunk a célkönyvtárra és az A:-ról hívjuk meg az SFX-et

```
C:\munka>A:SFXfájl -d
```

Az A:-ról nyissuk ki a fájlt a célkönyvtárba:

```
A:\>SFXfájl -d c:\munka *.doc
```

Az SFX archívok ajánlott használata

1. Ha nincs, hozzuk létre a célkönyvtárat, amennyiben külön könyvtárban akarjuk elhelyezni a kipakolt állományokat. (Pl.: md c:\uj\kipakolt)
2. Lépünk a célkönyvtárba/meghajtóra (Pl. C:> cd \uj\kipakolt)
3. Hívjuk meg az SFX-et, megadva annak elérési útját is. (Pl. C:\UJNKIPA-KOLT>A:zip110)

Természetesen ettől el is térhetünk, de ez a legegyszerűbb és a legbiztonságosabb használata az SFX állományoknak.

G.) Parancsok, kapcsolók

A PKZIP és a PKUNZIP parancsai egybetűsek, némelyiknek további paraméterei lehetnek, amelyeket a parancstól nem lehet elválasztani, legfeljebb elhagyhatók. A parancsok előtt egy bevezető mínusz (-) karaktert kell kitenni. Több parancs is összevonható egyetlen közös bevezető mínuszjel mögé, csak arra kell vigyázni, hogy ne keveredjenek a parancsok és a parancsok kiegészítő paraméterei. A parancsok négy kivétellel (c-C, p-P, w-W, j-J) egyaránt írhatók nagy és kisbetűvel, de célszerű megmaradni a kisbetűs formánál.

A PKZIP 1.10 PARANCSAI:

A parancs után írt - jellel kikapcsolhatjuk az adott parancsot, ha az alapértelmezésként aktív.

Bepakoló parancsok

- a Hozzáadás (Add).
- u Aktualizálás (Update).
- f Frissítés (Freshen).
- i Szelektív backup.

- m Áthelyezés (Move).
- mu Áthelyezés (Move) aktualizálással.
- mf Áthelyezés (Move) frissítéssel.
- \$ Lemeznév bepakolása a ZIP fájlba.
- p/P[-] Elérési út beírása a ZIPkatalógusba (Path).
- r[-] Könyvtárszerkezet tárolása (Recurse).

Megjegyzések, jelszavas védelem

- z Megjegyzés a ZIP fájlhoz (Zipcomment).
- c és -C Megjegyzések a bepakolt fájlokhoz (Comment).
- s<jelszó> Jelszavas védelem (Scramble).

Technikai jellegű parancsok

- x[<filespec>] Megadott állományok kizárása (eXclude).
- t[<datum>] Megadott dátumtól látja a fájlokat.
- q[-] ANSI szekvenciák engedélyezése/tiltása.
- j[hsr] Attributum maszkolás engedélyezése.
- J[hsr] Attributum maszkolás tiltása.
- w[hs] A rejtett és rendszerállományokat is lássa.
- W[hs] A rejtett és rendszerállományokat ne lássa.
- k[-] A ZIP fájl dátum aktualizálódjon, vagy sem.
- o[-] A Zip fájl dátum a legfrissebbhez igazodjon.
- e[xabis] Extra kompressziós módszer kijelölése.
- 3 386-os lehetőségek mellőzése.
- ^ Echózás.
- ! Jogosultság.
- b[<drive>][<path>] Átmeneti munkakönyvtár kijelölése.

Egyéb parancsok

- v[brt][cdenops] Archív katalógus megjelenítése.
- d Törlés a ZIP fájlból.
- h Help képernyőt ad.
- l Licenc képernyőt ad.

A PKUNZIP 1.10 PARANCSAI

Kipakoló parancsok

- e[cdenps] A kibontó parancs.
- x[cdenps] A kibontó parancs.
- n Csak újabb állományt nyit ki.
- o Felülír kinyitáskor.
- f Frissít, új vagy újabb állományokat nyit csak ki.
- d Az elmentett célkönyvtárat létrehozza.
- \$ Elmentett lemeznevet visszaír.
- c[m] Képernyőre nyit.
- t Teszteli a ZIP fájl épségét.
- p[abc][1234] Printerre nyit.

Technikai és egyéb parancsok

-q[-]	ANSI szekvenciák engedélyezése/tiltása.
-s<jelszó>	Jelszavas védelem (Scramble).
-j[hsr]	Attributum maszkolás engedélyezése.
-J[hsr]	Attributum maszkolás tiltása.
-v[bcdenpsr]	Archív katalógus megjelenítése.
-h	Help képernyőt ad.
-l	Licenc képernyőt ad.
-3	386-os lehetőségek mellőzése.
-^	Echózás.

A ZIP-SFX ARCHIVOK PARANCSAI:

Kipakoló parancsok

-e[cdenps]	A kibontó parancs.
-x[cdenps]	A kibontó parancs.
-n	Csak újabb állományt nyit ki.
-o	Felülír kinyitáskor.
-f	Frissít, új, vagy újabb állományokat nyit csak ki.
-d	Az elmentett célkönyvtárat létrehozza.
-\$	Elmentett lemeznevet visszaíre label.
-c[m]	Képernyőre nyit.
-t	Teszteli a ZIP fájl épségét.
-p[abc][1234]	Printerre nyit.

Technikai és egyéb parancsok

-q[-]	Ansi szekvenciák engedélyezése/tiltása.
-s<jelszó>	Jelszavas védelem (Scramble).
-j[hsr]	Attributum maszkolás engedélyezése.
-J[hsr]	Attributum maszkolás tiltása.
-h	Help képernyőt ad.
-l	Licenc képernyőt ad.
-3	386-os lehetőségek mellőzése.
-^	Echózás.

H.) A konfigurációs fájl

A PKZIP a DOS-ból és a saját konfigurációs állományából (PKZIP.CFG) megadott környezetleíró változókat használ alapértelmezései beállításához.

A DOS-ból megadható vagy kötelezően megadandó változók:

SET PKZIP.CFG=C:\ZIP (hol található a PKZIP.CFG fájl)

SET PKTMP=D: (a ramdisk legyen az átmeneti munkakönyvtár)

Ezt a két sort ajánlatos az AUTOEXEC.BAT-ba beszúrni.

A PKZIP és a PKUNZIP alapértelmezéseit megváltoztathatjuk egyszerűen a PKZIP.CFG nevű konfigurációs fájl változóinak módosításával. A konfigurációs fájlt először az aktuális könyvtárban keresi a PKZIP. Ha ott nem találja, akkor a PKZIP.CFG

változóban megadott úton. Ha ott sincs, akkor ahonnan a PKZIP indult (DOS 3.x verzióktól felfelé).

Ahol több érték is rendelhető egy változóhoz, ott az egyes értékeket a per (/) vagy a vessző (,) jellel, illetve szóközzel elválasztva kell beírni.

A PKZIP.CFG konfigurációs állományának változói betűrendben

ANSI=enabled/disabled

A kettő közül az egyik választható. Parancssorbeli megfelelői:

enabled = -q
disabled = -q-

COMPRESS=size/speed/implode

Azt határozza meg a változó tartalma, hogy a tömörítésnél mi a fontosabb: a tömörítés mértéke (size), vagy a sebesség (speed). Választható közülük egy, vagy egyik sem.

Parancssorbeli megfelelője:

size = -ex
speed = -es
implode = -ei

EXCLUDE=hidden/system

A kettő közül választható az egyik, mindkettő, vagy egyik sem. Azt adja meg, hogy a PKZIP ne lássa, ne kezelje a rejtett (hidden) és a rendszerfájlokat (system). Parancssorbeli megfelelőik:

hidden = -Wh
system = -Ws
hidden,system = -Whs

INCLUDE=hidden/system

A kettő közül választható az egyik, mindkettő, vagy egyik sem. Azt adja meg, hogy a PKZIP lássa, kezelje a rejtett (hidden) és rendszer (system) fájlokat. Parancssorbeli megfelelőik:

hidden = -wh
system = -ws
hidden,system = -whs

LISTCHAR=@

Ennek a változónak a megváltoztatásával a listafájlok megadásakor a @ karakter helyett az ide beírt karaktert használhatjuk annak a jelölésére, hogy listafájl következik.

MASK=readonly/hidden/system

Választható közülük egy, kettő, három, vagy egyik sem. A bepakolt fájlok attribútumait maszkoljuk (readonly=csak olvasható, hidden=rejtett, system=rendszer).

A parancssorban a következő felel meg neki:

hidden = -jh
system = -js
readonly = -jr
hidden, system, readonly = -jhsr

OPTIONCHAR= -

A parancsok előtti bevezető karaktert cserélhetjük le másra, ha valamilyen okból nem a mínusz karaktert akarjuk használni bevezető karakterként.

PASS=<readonly>/<hidden>/<system>

Választható közülük egy, kettő, három, vagy egyik sem. A bepakolt fájlok attribútumait ne maszkoljuk (readonly=csak olvasható, hidden=rejtett, system=rendszer). A parancssorban a következő felel meg neki:

hidden =	-Jh
system =	-Js
readonly =	-Jr
hidden, system, readonly =	-Jhsr

PATHS=recurse/none/all

Választható közülük egy vagy egyik sem. A bepakolt állományok neveiben hogyan tárolja az elérési utat. Parancssorbéli megfelelői:

none =	-p-
all =	-P
recurse =	-p

PKNO386=

Ha ennek a változónak értéket adunk, akkor 386-os gépen nem használja ki a 386-os processzor spéci utasításait, miáltal kb. 20%-kal lelassul a program, de így a PKZIP biztosan nem ütközik más, párhuzamosan futtatott programokkal.

RECURSE=on/off

Választható közülük egy, vagy egyik sem. Azt határozza meg a változó tartalma, hogy a könyvtárszerkezetet beírja a ZIP fájlba, vagy ne írja be. Parancssorbéli megfelelői:

on =	-r
off =	-r-

SHRINK > [320]

Az a fájl méret, ami alatt a shrinking módszerrel tömörít a PKZIP. E fölött használható csak az imploding.

VIEW=brief/natural/comments/extension/size/date/name/ratio/long

Azt vezérli az itt megadott érték(sorozat), hogy a ZIP fájl archívkatalógusának kilistázásakor a lista hogyan jelenjen meg. A felsorolt értékek közül több is kombinálható. (Az utolsó három például az összes többivel.)

A parancssorbéli megfelelő parancsok:

comments =	A megjegyzések is megjelennek	-vc
extension =	A kiterjesztések ábécé sorrendben	-ve
size =	Nagyság szerinti sorrendben	-vs
date =	Időrendben	-vd
name =	A nevek ábécé sorrendben	-vn
ratio =	Tömörítettség % szerinti sorrendben	-vp
natural =	Bepakolási sorrendben	-vo

reverse =	Fordított sorrendben	-vr
brief =	Röviden, CRC és attribútum nélkül	-vb
long =	Részletes lista, fájlanként egy screen	-vt

ZIPDATE=current/latest/keep

A három közül egy választható. A ZIP fájl aktualizálásakor a ZIP fájl dátuma a következő lehet:

current =	az éppen aktuális dátum (alapértelmezés)	-k-, -o-
keep =	nem változik meg	-k
latest =	a bent lévő legfrissebb fájl dátuma	-o

I.) Speciális szolgáltatások

Listafájl

Mint a bevezetőben már említettem, az PKZIP, a PKUNZIP és a ZIP-SFX-ek is használhatnak úgynevezett listafájlokat, melyekbe összegyűjthetjük azokat a fájlneveket, maszkokat amelyeket a parancssorba íránk. A listafájl egy közönséges, formázó karaktereket nem tartalmazó ASCII szövegfájl, amely ilyen formátumban dolgozó bármelyik szövegszerkesztővel elkészíthető.

A listafájl kifejezetten a tömörítővel végzett munkák dokumentálására szolgál, valamint megkímélhet a gyakorta ismétlődő hosszú listák sokszori és elkerülhető begépelésétől. Egy bevezető @ karakter jelzi a program számára, hogy listafájl következik. Ha a listafájl nem az aktuális könyvtárban van, akkor a @ jel és a listafájl neve közé be kell írni a listafájl elérési útját is. A listafájl ne legyen 32 K-nál nagyobb. A parancssorban kedvünkre keverhetjük a fájl maszkokat és a listafájlokat.

Az 1.10 verzió két újdonsága a listafájl használatában:

- Az x (eXclude files) parancs mellett is megadhatunk listafájlt a folyamatból kizárandó állományok neveivel.

- A ZIP-SFX archívok is tudják már a listafájlokat használni.

Munkakönyvtár, átmeneti állományok.

-b[<meghajtó:>][<path>]

A PKZIP alapértelmezésként mindig az aktuális lemezegység aktuális könyvtárába írja be átmeneti állományait. Mivel ez, főleg floppykon dolgozva állandó helyproblémákat okozhat és okoz is meglehetősen gyakran, célszerű tudatni a programmal, hova szemeteljen. Ez nemcsak a helyproblémákon segíthet, hanem gyorsítja is a munkát, főleg, ha megfelelő méretű ram drive, vagy gyors winchester áll rendelkezésünkre.

Amennyiben a PKTMP változóba a PKZIP indítása előtt (célszerűen már az AUTOEXEC.BAT fájlba) az alábbi sort beszúrjuk:

SET PKTMP=<drive:\munkakönyvtár>

és nem jelölünk ki meghajtót és könyvtárnevet, akkor a PKZIP a -b paranccsal a PKTMP változóban előzőleg kijelölt lemezre és könyvtárba szemetel. A -b parancs, ha használjuk, mindig a PKZIP utáni első helyen legyen. A többi paranccsal ellentétben ennek ajánlatos rögzített helyen lenni.

A PKZIP arra használja az így megadott könyvtárat, hogy az új ZIP fájlt !!! kiterjesztéssel ezen a gyors meghajtón hozza létre s ha kész, akkor egyszerűen rámásolja a már elavult állományra. Főleg akkor ajánlatos a használata, ha lassú és gyorsan megtelő lemezekre dolgozunk.

Állományok kizárása a ZIP műveletekből

-x (=eXclude files)

A parancs megjelenése azt jelenti, hogy Phil Katz végre integrálta a NO.COM programcskát is — a PKWare megszokott magas színvonalán — a PKZIP programba. Ezentúl, ha csak egy állomány, vagy egy adott csoport kivételével minden állománnyal akarjuk dolgoztatni a PKZIP-et, elég a `-x<fájl.kit>` parancsot kiadnunk a parancssor végén a *.* azonosító után. A PKZIP -a doksik *.* -x*.bak parancs hatására a PKZIP az aktuális könyvtár minden állományát bepakolja, a *.bak állományok kivételével.

Mint már a listafájloknál említettem, az 1.10 változattól kezdve a `-x` parancs mellett listafájl is adható a megszokott bevezető `@` karakter mögött. Így például az EZT.NE lista állományban felsorolt fájlokat nem csomagolja be a PKZIP -a doksik *.* -x@ezt.ne parancs.

A leírás szerint az eXclude fájl parancs a Delete (törlés) paranccsal is társítható.

Technikai parancsok

-e (=Extra compressiva)

A PKZIP bepakolásakor alapértelmezésként egyenként megvizsgálja a tömörítendő állományokat és a szerinte optimális módszerrel tömörít. Néhány esetben hasznos lehet, ha előírhatjuk a PKZIP-nek, milyen módszert használjon. Erre szolgál az e (= Extra compression) parancs a PKZIP mellett. A jelenlegi verzió öt módszer választását teszi lehetővé:

`-ex` = A maximális tömörítést eredményező módszerrel dolgozzon a PKZIP. Ez az alapértelmezés.

`-es` = A maximális sebességű tömörítést eredményező módszerrel dolgozzon a PKZIP (shrinking).

`-ei` = Az implode technikát alkalmazza a PKZIP, hogy a ZIP2EXE a kisebb méretű SFX-et készíthesse belőle el.

`-ea` = Az ASCII (text) fájlokon imploding, a bináris állományokon shrinking.

`-eb` = Az ASCII (text) fájlokon shrinking, a bináris állományokon imploding.

Az `-ea` és `-eb` együttes alkalmazása egyenértékű az `-ex` paranccsal. Bár a PKZIP kompatibilitási okokból számokat is (1-4) elfogad az `-e` parancsokban, ennek nincs hatása a tömörítésre. A shrinking módszerhez valamivel több szabad memória kell.

-t (= Time stamp)

Az 1.10 verzió bővítése ez a parancs, amely lehetővé teszi, hogy egy dátum korlátot adjunk a kiválasztáshoz. Mind a PKZIP, mind a PKUNZIP csak azokkal az állományokkal foglalkozik, amelyeknek keletkezése a megadott dátummal azonos, vagy annál frissebb. Az alábbi módon jelölhetjük ki bepakolásra az 1990.11.17. után készült állományokat: PKZIP -a ZIPFILE *.* -t111890

-^ (= echo)

Az echo parancs mind a PKZIP, mind a PKUNZIP mellett használható az 1.10 verziótól kezdve. A parancs megismétli a parancssort a végrehajtás előtt, így ellenőrizhető, hogy jól gépeltük-e be a parancssort.

-3 (nem használ 386-os technikát)

Olyankor lehet hasznos, ha 386-os gépen multitask rendszerben dolgozunk, és a 386-os lehetőségeket mind kihasználva könnyen összeakadhatna a program egy párhuzamosan futó másik programmal. Ilyenkor célszerű a -3 paranccsal mind a PKZIP, mind a PKUNZIP esetében egyszerűen letiltani a 386-os extra utasítások használatát. Az 1.10 verzióval jelent meg a parancs.

-! (jogosultság!)

Ha a program jogosult és regisztrált felhasználói vagyunk, azaz elküldtük a regisztrálási díjat és a kinyomtatott és kitöltött formanyomtatványt a PKWare cégnek, akkor megküldenek egy kódot, amit a PUTAV.EXE programmal beültethetünk a PKZIP.EXE állományba. Amint ezen is túl vagyunk, máris élvezhetjük a -! parancs használatának előnyeit:

— Az így kezelt PKZIP-pel a -! paranccsal előállított ZIP állományokat kibontáskor a PKUNZIP automatikusan ellenőrzi, és jelzi, ha illetéktelenül belepiszkáltak.

— Saját grafikai jelünket is kiírja az ellenőrzéskor (reklámlehetőség).

— A kilencezer forintért kapható, eredeti PKWare lemezeken sincs rajta semmi, amivel regisztrálttá tehetnénk a PKZIP-et. Ehhez mindenképpen el kell küldeni Amerikába a kitöltött regisztrációs adatlapot és a pénzt. A PKWare megküldi a PUTAV-val beadandó adatokat.

-h (= Help)

Ha nem írunk be archív nevet, a PKZIP, vagy a PKUNZIP nem tudja értelmezni az általunk beírt parancsokat, vagy a -h parancsot használjuk, akkor a program kiírja az adható parancsok és fontosabb paramétereik listáját, valamint a parancssor szintaxisát.

-l (= Licence)

A legtöbb tömörítőtől eltérően az -l parancs nem a szabvány szerinti archív katalógus kiírást biztosítja, hanem a program licenc képernyőjét jeleníti meg.

Errorlevel értékek (hibakódok) és batch alkalmazás

A PKZIP által kijelzett errorlevel értékek

Érték Jelentése

0 Nincs hiba.

1 Rosszul megadott fájlnev, vagy fájlspecifikáció.

2,3 Hiba a ZIP fájlban.

4-11 Túl kevés a memória.

12 Nem talált a PKZIP olyan fájlt amit bepakolhatna, vagy kitörölhetne.

13 Nem találta a megadott ZIP fájlt, vagy listafájlt.

14 Megtelt csurig a lemez, nem tud rá írni a PKZIP.

15 A ZIP fájl csak olvasható, nem módosítható.

16 Rosszul adták meg a paramétert, érvénytelen a paraméter.

17 Túl sok fájlt adtak meg egyszerre.

18 Figyelmeztető hibaüzenet, a PKZIP egy vagy több megadott fájlt nem tudott megnyitni. Egyébként a többi fájlt rendben bepakolta.

A PKUNZIP által kijelzett errorlevel értékek

Érték Jelentése

- 0 Nincs hiba.
- 1 Figyelmeztető, de nem végzetes hiba (Pl. hibás CRC ellenőrző szám).
- 2, 3 Hiba a ZIP fájlban.
- 4-8 Nincs elég szabad memória.
- 9 Nem találja a ZIP fájlt.
- 10 Rossz vagy érvénytelen paramétert adtak meg.
- 11 Nincs a megadott feltételeket teljesítő fájl a ZIP fájlban.
- 50 Tele a lemez, nem tud a PKUNZIP írni rá.
- 51 Váratlan fájlvége jel a ZIP fájlban.

A ZIP-SFX fájlok hibakezelése

Érték Jelentése

- 0 Nincs hiba.
- 1 Figyelmeztető, de nem végzetes hiba (Pl. hibás CRC ellenőrző szám).
- 2 Nem találja az EXE fájlt.
- 3 Hiba az EXE fájlban.
- 4 Hibás a fájltábla.
- 5 Nem tudja létrehozni a fájlt.
- 6 Betelt a céllemez, nem tud rá írni.
- 7 Túl sok fájl.
- 8 A felhasználó megszakította a futást a comment szünetben.
- 12 Ha a Zipcomment-ben CTRL-S volt és nem Y-t választottunk, akkor a program ezt az errorlevel értéket állítja be.

A fenti értékek ismeretében olyan batch-fájlokat írhatunk, melyek az errorlevel értékének vizsgálatával ellenőrzik a programok futását, s jelzik, akár ékes magyar nyelven, hogy mikor, milyen hiba történt.

LAN támogatás

Mind a PKZIP, mind a PKUNZIP úgy nyitja meg az állományokat a hálózatokon, hogy egyidejűleg más taszkok is használhatják azokat. A PKZIP átmeneti állományainak egyedi neveket ad, így nemigen kerül összeütközésbe a párhuzamosan futó egyéb alkalmazásokkal hálózati meghajtókon sem.

J.) A PKZIP és PKUNZIP a gyakorlatban

A PKZIP használatakor a következő szerkezetet használjuk:

PKZIP [-b] [-parancsok] archív [maszkok] [>kimenet]

A PKUNZIP használatakor a következő szerkezetet használjuk:

PKUNZIP [-parancs] archív [célkönyvtár\] [maszk] [>kimenet]

A PKZIP csomag programjait célszerű egy path-on lévő könyvtárba tenni, ahol a DOS könnyen megtalálja és nem kell az elérési utakkal bíbelődni a program minden egyes meghívásánál.

A parancsok helye nem kötött, a PKUNZIP mögött bárhol lehetnek a sorban. Ha nem adunk parancsot, akkor az alapértelmezés a Extract parancs.

Az archív nevét mindig meg kell adni. Az Add, Update, Freshen, Move, selective backup és Delete parancsok használatakor az archív nevében nem szerepelhetnek dzsóker

(* és ?) karakterek, az összes többi parancs használatakor igen. Az archív neve természetesen tartalmazhat elérési utat is.

A PKUNZIP után megadható célkönyvtár végére ne felejtsük el kitenni a \ jelet.

A maszkok lehetnek fájlnevek, melyekben a dzsóker (* és ?) karaktereket szabadon használhatjuk, tartalmazhatnak elérési utat is, mind a be, mind a kikapcsolásokhoz. A maszkok között listafájlt is megadhatunk, ennek ismertetését lásd a kilencedik témakörben részletesen, valamint a gyakorlati példák között.

A kimenetet nem maga az PKZIP, vagy PKUNZIP program, hanem a DOS használja. Segítségével átirányíthatjuk a különben csak a képernyőn megjelenő szövegeket egy megadott fájlba, vagy nyomtatóra. Hasznos ez a lehetőség, amikor például az archív katalógust ki akarjuk nyomtatni, vagy egy listafájllhoz akarjuk felhasználni, esetleg munkánkat és a ZIP programok üzeneteit akarjuk dokumentálni.

Mindennapi feladatok, megoldások

1. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, de nem akarjuk a könyvtár tartalmát elyeszíteni.

```
C:\MUNKA>PKZIP a:munka c:\munka\konyv\*.*
```

2. A C:\MUNKA könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az archív elkészültével ki akarjuk írítani a könyvtárat.

```
C:\MUNKA>PKZIP -m a:munka c:\munka\konyv\*.*
```

3. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, kivéve a *.BAK állományokat.

```
C:\MUNKA>PKZIP -m a:munka c:\munka\konyv\*.* -x*.BAK
```

4. Csak a napi munka során megváltozott állományokat akarjuk az A:-ban lévő lemezen tárolt MUNKA.ZIP fájlban egy kicsit felfrissíteni.

```
C:\MUNKA>PKZIP -f a:munka c:\munka\konyv\*.*
```

5. A C:\MUNKA és a C:\MUNKA\KONYV könyvtárakból minden DOC kiterjesztésű állománnyal aktualizáljuk az A:MUNKA.ZIP állományt. A forrásállományokat nem akarjuk törölni.

```
C:\MUNKA>PKZIP -u a:munka c:\munka\konyv\*.doc c:\munka\*.doc
```

6. A C:\MUNKA\KONYV könyvtárból minden nem COM kiterjesztésű állománnyal bővítjük az A:MUNKA.ZIP állományt. A forrásállományokat nem akarjuk törölni.

```
C:\MUNKA\KONYV>PKZIP -x*.com a:munka *.*
```

7. A MUNKA.ZIP archívba tévedésből bekerült BAK, ARC és ~DO kiterjesztésű állományokat kell eltávolítanunk.

```
C:\MUNKA>PKZIP -d a:munka *.bat *.~do *.arc
```

8. Meg akarjuk nézni, mi van a MUNKA.ZIP állományban.

```
C:\MUNKA>PKZIP -vc a:munka Tömör listát ad megjegyzésekkel.
```

```
C:\MUNKA>PKZIP -vt a:munka Teljes listát ad egyenként sorolva az állományokat.
```


9. Ki akarjuk listáztatni a MUNKA.ZIP tartalmát egy szabadon szerkeszthető ASCII állományba, de csak a nevekre és a kiterjesztésekre vagyunk kíváncsiak. (Pl. egy listafájl készítését tervezgetjük.)

```
C:\MUNKA>PKUNZIP -vbn a:munka >lista.lst
```

10. Gyanús a lemez. Ellenőrizni akarjuk a MUNKA.ZIP épségét.

```
C:\MUNKA>PKUNZIP -t a:munka
```

11. Meg akarjuk nézni az egyik (pl. az ARC.DOC) állományt.

```
C:\MUNKA>PKZIP -cm a:munka arc.doc
```

12. Megtetszett a fenti ARC.DOC. Nosza nyomtassuk csak ki az LPT2 portra kapcsolt nyomtatón, ahol előzőleg grafikát nyomtattunk.

```
C:\MUNKA>PKUNZIP -pa2 a:munka arc.doc
```

13. A régi verziójú PKZIP programmal pakolt KAKUKK.ZIP nevű állományt egy kicsit tömörebbé akarjuk tenni, de azért a régi KAKUKK.ZIP-et is meg akarjuk őrizni.

```
C:\MUNKA>PKUNZIP a:kakukk c:\atmeneti\
```

```
C:\MUNKA>PKUNZIP a:kakukk2 c:\atmeneti\*. * -ex
```

14. A MUNKA.ZIP-ből a TXT kiterjesztésű állományok a C:\TXT könyvtárba, a DOC kiterjesztésűek a C:\DOKSI könyvtárba pakolandók ki.

```
C:\MUNKA>PKUNZIP a:munka c:\txt\*.txt
```

```
C:\MUNKA>PKUNZIP a:munka c:\doksi\*.doc
```

15. Maszek állományainkat titkosítva akarjuk a MUNKA.ZIP archívba áthelyezni, hogy a C:-n nyoma se maradjon.

```
C:\MUNKA>PKZIP -musna_talald_ki_mi_a_jelszo a:munka maszek.*
```

Menüvel bontogatni — PKZMENU

A PKWARE végre elkészítette PKUNZIP egérrel is jól használható, menüvezérelt változatát. A programmal a PKUNZIP összes funkciója használható, de rengeteg új lehetőség is nyílik. Többek között beleépítették a vírusok elleni harcban igen megbízható SCAN program használatát. A programhoz egy részletes (90 oldal, több mint 200 K) és jól használható angol nyelvű leírás is tartozik.

Egér használatakor, amit a program teljes mértékben támogat, az egér bal gombja az <ENTER>, jobb gombja pedig az <ESC> billentyű lenyomásának felel meg.

A PKZMENU futtatásához DOS 2.0 feletti verzió kell, és legalább 192 K szabad RAM. Futtatható floppylemezről is, de sokkal gyorsabb winchesterről indítva. Természetesen path-on lévő könyvtárban ajánlatos tartani, hogy a DOS bármikor megtalálhassa.

A parancssor szintaxisa

pkzmenu [drive:][path] [kapcsolók]

Indításkor a program négy kapcsolója közül kettőt választhatunk egyszerre. A kapcsolók helye nem rögzített, a parancssorban bárhol lehetnek a szokásos bevezető /, vagy – kapcsoló karakter után.

Ezek a következők:

/3

Ne használja ki a 386-os gépek fejlettebb utasításkészletét, akkor sem, ha a program konfigurációs állománya erre lehetőséget ad.

/386

386-os gépen használja ki a 80386-os processzor fejlettebb utasításkészletét.

/e+

A ZIP fájloknak automatikusan jelezze ki a tartalmát.

/e-

Ne írja ki automatikusan a ZIP fájloknak a tartalmát.

A programnak van egy PKZMENU.CFG állománya is, amit maga a program hoz létre, illetve módosít. Mivel ez nem egy normál ASCII szövegfájl, mint a PKZIP konfigurációs állománya, ne próbáljuk szövegszerkesztővel módosítani a tartalmát.

A PKZMENU képernyője

A képernyő tetején található a program főmenüje (Extract, Tag, Sort, Misc, setuP, eXit). Az egyes menüpontokhoz tartozó almenüket többféleképpen is elérhetjük:

- <ALT x> kombinációkkal, ahol x a menüpont kiemelten, nagybetűvel írt betűje.
- Az egér „bal szemével” rákattintva a választott menüpontra.
- A jobbra/balra nyíl megnyomására először az Extract almenüje nyílik ki, az adott billentyű újbóli megnyomására a soronkövetkező menüpont almenüje jön elő.
- Az <ESC> billentyűvel kiléphetünk a menükből, mindig egyet lépve vissza, végül a programból is kiléphetünk vele.
- Az e billentyű az Extract menüt nyitja meg.
- Az s billentyű a Sort menüt nyitja meg.

Az almenükben a fel/le nyilakkal, vagy az egérrel mozoghatunk. Az egyes pontokat az adott menüpont kiemelten jelzett billentyűjével, ráállás után az <ENTER>-rel, vagy az egér bal gombjával rákattintva választhatjuk.

A menük alatt keretben van a lista-terület, melynek bal felső részén látható az aktuális meghajtó és könyvtár neve. A program alapértelmezésként a könyvtárakat és a ZIP fájlokat sorolja fel itt, illetve a setup beállítástól függően a ZIP állományok tartalmát is.

A lista-területen a fel/le, illetve a <PGUP>, <PGDN>, <HOME> és <END> gombokkal, illetve az egérrel mozoghatunk. A listában szereplő könyvtárakba átléphetünk, ha a könyvtár nevére állva megnyomjuk az <ENTER>-t, vagy az egér baloldali gombját.

Kijelölés a listaterületen

A <SPACE> be/kikapcsolja a jelöltséget. Ha egy adott ZIP fájlt már kijelöltünk, akkor a benne lévő tömötített állományokkal ezt nem kell megismételni, a jelölés a ZIP fájl teljes tartalmára érvényes.

Az <ENTER> is kijelöl állományokat, emellett a ZIP fájlokat ki is listázza. Ha már kilistázott ZIP fájlra nyomjuk meg az <ENTER>-t, akkor az becsukódik.

Egyéb billentyűk hatása a lista-területen:

- t — Az adott fájl technikai információit jeleníti meg.
- n — Az aktuális könyvtár tartalmát olvassa újra.
- <F1> — A PKZMENU egyképernyősnyi Help-jét hozza elő.
- <F2> — Kinyitja a célkönyvtárban a kijelölt állományokat. Ha nincsenek kijelölt állományok, a program egy kis ablakban közli ezt velünk.
- <F3> — Ellenőrzi a kijelölt állományok épségét a CRC értékek segítségével. Ha nincsenek kijelölt állományok, a program egy kis ablakban közli ezt velünk.
- <F5> — Megnyit egy ablakot, ahova beírhatjuk azt a path-t, ahová át akarunk lépni.
- : — Megjeleníti a hozzáférhető meghajtók neveit, amelyek közül rámutatással választhatunk.

kijelölt állományaiból egy PKZIP-pel kompatibilis listafájlt készít.

/ — File keresésére szolgál. A / billentyű lenyomására kinyíló ablakban megadhatjuk azokat a karaktereket, amelyekkel a keresendő fájl kezdődik. Ha a program több találatot is jelez, akkor a megtalált állományok között a fel/le nyilakkal mozoghatunk.

<SHIFT F1> — Színes monitorral rendelkezők a <SHIFT F1> lenyomása után kedvükre variálhatják a program színeit. Az ANSI.SYS-nek ajánlatos a CONFIG.SYS sorai között szerepelni az alábbihoz hasonló sorban: DEVICE=c:\DOS\ANSI.SYS.

A menüpontok

Az almenük pontjai alapvetően négyféle típusba sorolhatók:

- Azonnal végrehajtásra kerül, ha kiválasztjuk.
- Ablak nyílik, ahová további adatokat lehet vagy kell beírunk.
- Egy-egy adott kapcsolót be- és kikapcsolunk. A bekapcsolt neve mellett ✓ kipipáló jel van, a kikapcsolt mellett pedig • pontocska.
- Újabb almenü nyílik ki. Előtte egy ■ jel látható.

Extract (kibontás)

Extract tagged files — F2 — Kinyitja a kijelölt állományokat a megadott célkönyvtárban. Kibontás után a lista-terület alatti sorban olvasható a program üzenete a kibontási folyamatról.

Freshen extract — Kinyitáskor a program ellenőrzi a célkönyvtárat is, és csak azokat az állományokat pakolja ki, amelyek már a célkönyvtárban is léteznek.

Print tagged files — A kijelölt állományokat a Printer setup menüben megadható nyomtatóra az ugyancsak ott megadható módon írja ki.

Console More: YES/NO — A kijelölt ASCII text állomány(oka)t megjeleníti a képernyőn. Ha a More mellett a YES-t választjuk, akkor nem folyamatosan gördül felfelé a szöveg, szinte követhetetlen sebességgel, hanem egy-egy képernyő teleírása után megáll.

Test integrity — F3 — Leteszteli a kijelölt állomány(ok) épségét a CRC értékek alapján. A PKUNZIP -t parancsát helyettesíti.

Overwrite — Ha aktivizáljuk ezt a kapcsolót, akkor kinyitáskor kérdés nélkül felülírja a célkönyvtárban esetleg már meglévő azonos nevű állományt. Alapértelmezés szerint minden egyes felülírás előtt rákérdez, hogy csinálhatja-e.

Newer — Csak új, vagy frissebb állományokat bont ki. Ha a célkönyvtárban már meglévő azonos nevű fájl frissebb, mint a ZIP fájl belsejében lévő kijelölt állomány, akkor nem írja felül a már meglévőt.

Directory creation — Az eltárolt könyvtárszerkezetre vonatkozó információkat használja-e a kibontáskor. A PKUNZIP -d parancsa felel meg neki. Ha kikapcsoljuk, akkor csak maguk az állományok kerülnek kibontásra a célkönyvtárban.

Volume label creation — Az eltárolt lemeznevet kiírja-e a céllemezre. A PKUNZIP-nél ezt a -\$ parancs végzi. A program felülírja a már meglévő lemeznevet a ZIP fájlból kiszedett névvel.

Attribute mask — Maszkolja-e kibontáskor a kijelölt állományok rejtett (hidden), rendszer (system) illetve csak olvasható (readonly) attribútumait? Almenüből választhatjuk ki, hogy az adott attribútumok közül mit maszkoljon.

Scramble (password) — A jelszóval titkosított állományokat kibontáskor, teszteléskor, megnézéskor és nyomtatáskor átugorja a program, amennyiben nem adjuk meg a megfelelő jelszót. Ez a menüpont a jelszó megadására szolgál. A jelszóban a program különbséget tesz a kis- és a nagybetűk között!

Where to extract — A célkönyvtár megadására szolgál. Egy ablakban, amely a menüpont választására nyílik meg, beadhatjuk a célkönyvtár adatait. Az itt megadott célkönyvtár teljes neve látható a lista-terület alatti sorban. A beadott könyvtárnevet az <F1> lenyomásával elmenthetjük, illetve az előzőleg használt és elmentett célkönyvtárak közül a fel/le nyíl billentyűvel választhatunk.

Tag (kijelölés)

Tag all files — Az összes fájl ki lesz jelölve.

Clear all tags — <ALT c> — Az összes kijelölt fájl jelöltségét törli.

Mode: tag files/clear tags — Ha itt a tag files van, akkor a program csak a jelölt állományokkal dolgozik. Ha a Clear tags van itt, akkor a munka végeztével a kezelt állományok jelöltségét törli a program.

Name — Itt egy ablak nyílik, ahol megadhatjuk a kijelölendő állományok neveit. A DOS szabványos névmegadási szabályai érvényesek, a dzsóker (*, ?) karakterek használata megengedett. Az <F1> billentyűvel itt meghatározhatjuk, hogy a megadott nevű állományok keresésénél a ZIP fájlban tárolt path információkat figyelembe vegye-e a program.

Size — Megadható méretküszöb is az állományok kijelöléséhez. A fel/le nyilak segítségével az összehasonlító operátorokat (>, <, =, >=, <=, vagy <>) választhatjuk meg.

Date — A fájlok keletkezését is használhatjuk kijelölési kritériumként, a dátumot nn/hh/éé alakban megadva. Az összehasonlító operátorokat ugyancsak a fel/le nyilakkal választjuk ki.

Only today's files — A mai nap készült/módosított állományokat jelöli ki.

In this zip — Az adott ZIP fájl összes bepakolt állományát jelöli.

Archive (changed files) — Az archív attribútumú ZIPfájlokat jelöli ki.

Flip tags — A kijelölést átkapcsolja.

Remove from display — A kijelölt állományokat a képernyőről eltünteti.

Sort (rendez)

A kiírást a menüben választhat módon rendezi. Az aktív módszer neve mellett ✓ jel látható.

Sort Name A-Z/Z-A — Név szerint rendez a-tól z-ig, vagy fordítva.

Sort Extension A-Z/Z-A — Kiterjesztés szerint rendez a-tól z-ig, vagy fordítva.

Sort Date 0-9/9-0 — Dátum szerint rendez időrendi sorrendben, az újabbtól a régebbiekig (9-0), vagy fordítva.

Sort Size 0-9/9-0 — Méret szerint rendez, a legkisebbel (0-9), vagy a legnagyobbval (9-0) kezdve.

Sort Unsorted — Rendezés nélkül jelez ki mindent. Ezt választva minden eddigi kijelölés törlődik, mert a program újraolvassa a lemezt.

Inner Sort — A ZIP fájlban belül is rendez. Választható, hogy név, kiterjesztés, dátum, méret, tömörítettség, CRC szerint rendezzen, vagy ne rendezzen.

Misc (vegyes)

Drive/directory — <F5> — Itt adhatjuk meg, hogy melyik lemezre és/vagy könyvtárra akarunk átkapcsolni. A program megjegyzi az utolsó öt könyvtárat, s ezek közül a fel/le nyilakkal választhatunk. Ha az <F5> billentyűt nyomjuk meg, akkor nem kell a menükkel bajlódni, azonnal a könyvtárváltó ablakba jutunk.

Make a directory — A program egyik leghasznosabb újdonsága ez a lehetőség. Ezzel bármelyik hozzáférhető lemezen létre tudunk hozni új könyvtárat, anélkül, hogy akár átmenetileg is ki kellene lépni a DOS-ba.

List file — @ — A program a PKZIP és PKUNZIP számára használható listafájl generálhat, ha ezt a menüpontot választjuk. A listafájl neve és elérési útja mellett azt is meg kell adnunk, hogy a path-t tartalmazza-e a listafájl. Ha nem adunk meg elérési utat,

akkor az aktuális könyvtárba kerül. Amennyiben a listafájl nevével már létezik egy állomány, akkor ezt jelzi a program, és három lehetőség közül választhatunk:

<ESC> billentyűvel visszaléphetünk a menübe.

<F1> billentyűvel a listafájl felülírja az előző állományt.

<F2> billentyűvel hozzáfűzzük az új adatokat a már meglévő állományhoz.

Saját tapasztalataim szerint a meglévő listafájl bővítése még nem működik megfelelően, talán majd a következő verzióban kijavítják. Az új listafájl létrehozása azonban problémamentes.

Filespec — A kijelzésre kerülő archívokhoz adhatunk meg itt egy szűrő maszkot. Általában *.ZIP van itt, de az SFX-ek miatt be lehet venni az *.EXE állományokat is. A további szűkítés kedvéért a névből is megadhatunk néhány karaktert.

Technical info — T — Itt a ZIP fájlokról és a bepakolt állományokról kaphatunk technikai információkat.

1./ A ZIP fájlról:

Filename	= Fájlnev
File type	= Fájl típus (ZIP)
File count	= Az állományok száma
Date and Time	= A ZIP fájl keletkezési időpontja
Compressed size	= Tömörített méret
Uncompressed size	= Eredeti méret
Ratio	= A méretcsökkenés aránya

2./ Bepakolt állományról:

Filename	= Fájlnev
File type	= Fájl típus (bináris/ASCII)
Attributes	= Attribútumok
Date and Time	= Keletkezési dátum, időpont
Compression method	= Tömörítési módszer
Compressed size	= Tömörített méret
Uncompressed size	= Eredeti méret
Ratio	= A méretcsökkenés aránya
32 bit CRC value	= 32-bites CRC-érték
Created by	= Melyik PKZIP verzióval készült
Needed to extract	= Melyik PKUNZIP verzióval kell kibontani

Quick locate — / — Ha sok állomány van az aktuális könyvtárban, (akár ZIP fájl, akár ZIP fájlba bepakolt tömörített fájl), igen hosszadalmas lenne egy adott fájl megkeresése. Ezen segít a PKZMENU a Quick locate menüponttal, illetve a / billentyűvel. Lehetővé teszi, hogy már a keresett állomány(ok) első karaktereivel megtaláljuk amit keresünk. A találatokat a program számolja, és a fel/le nyilakkal mozoghatunk az egyes megtalált állományok között. A keresendő fájlnev megadásakor a DOS szabványos dzsóker karaktereit is használhatjuk.

View file — <ALT v> — A setup menüben megadott fájlolvasó segédprogrammal megnézhetjük azt az állományt, amin állunk. Az <ALT v> kombinációval a menü megkerülésével ugyanezt elvégezhetjük.

New directory — N — Az aktuális lemez vagy könyvtár katalógusát újraolvassa.

StupenDos — A PKWare által forgalmazott StupenDOS nevű DOS bővítés illesztését végzi ez a menüpont.

Help — <F1> — A PKZMENU egy képernyőnyi Help-jét jeleníti meg. Itt megtalálható a program használatához szükséges majdnem minden információ.

About PKZMENU — Néhány sor arról, hogy kik írták a programot, kié az, illetve hol és mennyiért lehet regisztráltatni.

Setup (alapbeállítások)

A menüpont elemeit választva négy almenüvel találkozhatunk, csak a „Write setup to disk” nem kínál újabb almenüt.

— Zip options — Expand zip files — Bejelentkezéskor a ZIP fájlokat listázza.

— Comments displayed — Kijelzi a ZIPkomment első sorát.

— Pathnames displayed — A kilistázott ZIP fájlban levő állományok elmentett path adatait is kijelzi.

— Volume label creation — Kibontáskor a bepakolt lemeznevet visszaírja a céllemezre.

— More (console) — A program a képernyőre nyitáskor vár egy-egy képernyő, és a PKUNZIP-nél megszokott módon nézhetjük végig a szöveget.

— Overwrite — Ha ez aktív, akkor kipakolásakor a PKZMENU mindent kérdezés nélkül felülír.

— Newer — Kipakolásakor csak új, vagy frissebb fájlokat nyit ki a program.

— Directory creation — Kibontáskor az eredeti könyvtárszerkezetet is visszaállítja a program, ha ez a menüpont aktív.

— Halt on error — Megálljon-e, ha a program működés közben hibát talál.

— Sort inners — Unsorted — Hogyan rendezze a bepakolt állományok listáját.

— Inner display — size, time, date — Itt azt választhatjuk meg, hogy a program milyen adatokat írjon ki egy-egy ZIP fájl kilistázásakor. A választék a következő: tömörítési adatok, CRC, %, fájlnev, méret, időpont, dátum, fájlkommentek.

— Temporary drive — Az átmeneti munkakönyvtár adható meg. Célszerű egy megfelelő méretű ramdrive, vagy egy gyors winchester.

— Where to extract — Célkönyvtár a kibontáshoz.

— Attribute mask — Itt egy újabb almenü jelentkezik, amelyből kiválaszthatjuk, hogy a program kipakolásakor a rejtett, rendszer és readonly állományok attribútumait maszkolja-e.

— Zip file extension — Megadhatók azok a kiterjesztések, amelyekkel ZIP fájlokat tárolunk. Azt is meg kell adni, hogy az adott kiterjesztésű állományokat kilistázza-e bejelentkezéskor.

Video & sound options

Megint egy hosszú menü: Beállíthatjuk, hogy a monitorunk monokróm, színes, EGA fekete-fehér, EGA színes, VGA fekete-fehér vagy VGA színes. A program megnézi a kártyát, így Hercules vagy CGA kártyához nem kínálja fel az EGA és a VGA lehetőségeit. A színválasztással kedvünkre variálhatjuk a program színeit. A Beep menüponttal választhatunk, hogy a PKZMENU a megfelelő esetben adjon-e hangjelzést. Végül az választhatunk, hogy akarjuk-e látni a rendszer óráját a jobb felső sarokban, vagy sem.

Printer setup

Először a printer kimenetet állíthatjuk be. A lehetséges értékek között az <ENTER> nyomogatásával választhatunk (LPT1-4, COM1-4, PRN). A Mode melletti érték azt határozza meg, hogy nyomtatáskor a program milyen üzemmódban (ASCII, Binary,

None) dolgozzon. Az utolsó pontban az egyes állományok közötti lapdobást kérhetjük, illetve tilthatjuk meg.

Miscellaneous

Ez egy vegyesfelvágott almenüt ad. Az első pont (Initial sort) a kezdeti rendezésre vonatkozik. A második pontnál (Return to startup dir) választhatunk, hogy kilépéskor az induló, vagy az aktuális könyvtárba akarunk-e kilépni. Itt kell azt is megadnunk, hogy használunk-e egeret (Use mouse), és akarjuk-e a 386-os lehetőségeket maximálisan kihasználni (386 instructions).

A vírusellenőrzés is ide került. Egyrészt megadhatjuk, hogy egyáltalán akarunk-e élni vele (Check for viruses), illetve hol van víruskereső programunk indító állománya (Virus program).

A fájlolvasó programot is itt adhatjuk meg, a megfelelő paraméterezéssel (List file).

APKWare terméke a StupenDOS nevű DOS bővítés/shell. Ennek elérését a StupenDOS menüpontnál adhatjuk meg.

Az utolsó itt tárgyalt menüpont az egérbeállítás. Ez annyira szemléletesen van megoldva, hogy nincs szükség külön ismertetésére.

Write setup to disk — Az éppen aktuális beállításokat kiírja a PKZMENU.CFG-be.

eXit (kilépés)

Original dir — <ALT x> — Az induló könyvtárba lép ki a program, miután megköszönte, hogy használtuk.

Current dir — <CTRL x> — Az éppen aktuális könyvtárba lép ki a program, miután megköszönte, hogy használtuk.

Kommentezés egy iZIPicit elegánsabban

A PKZIP — hasonlóan a többi felsőosztálybeli tömörítő programhoz — lehetővé teszi egy már használható méretű megjegyzés hozzáfűzését a ZIP állomány egészéhez (= ZIPkomment) és az egyes bepakolt állományokhoz (= komment). Ez a lehetőség — mint ismert — a PKZIP -z és -c parancsai segítségével áll rendelkezésünkre a legegyszerűbben. A kommentező parancsok használata esetén a program a ZIP fájl létrehozásakor, illetve módosításakor bekéri a ZIPkommentet, illetve a kommentet, s a PKZIP 127 karakter hosszúságig hajlandó azt elfogadni.

A kommentezés nagyon hasznos lehetőség, de jó lenne, ha módunkban állna előre megszerkesztett, esetleg ANSI vezérlőkódokat is tartalmazó listaállományból beolvasatni a (ZIP)komment szövegét, és túlléphetnénk a 127 karakteres korlátot.

Több segédprogram is megjelent már, amelyek a fenti problémákra kínálnak valamiféle megoldást. Én két kommentező segédprogramot próbáltam — hiába.

A SolarSoft programkönyvtár ZIPCOMNT.EXE (#343, ZIP Utilities lemez) programja volt az első, amit megpróbáltam használni. Ez a program már saját leírása szerint is csak 117 karakterig fűzhet ZIPkommentet a ZIP fájlhoz, de maga a program szerzője azt javasolja, hogy ne nagyon írjunk 79 karakternél hosszabb szöveget ide. Erről ennyi több mint elég.

Egy előre megszerkesztett állományból is beolvastathatjuk a ZIP kommentet, ha előzőleg a ZIPCOMNT változóban elhelyezzük a kommentet tartalmazó fájl teljes nevét (pl. SET ZIPCOMNT=C:\ZIP\FILEFEJ.DAT). Ilyenkor a ZIPCOMNT a megadott maszknak megfelelő összes ZIP fájlt ellátja az adott állományból kiolvasott ZIP kommenttel. Ha azonos fejléccet akarunk a ZIP fájlokra aggatni, akkor a program remek, ha azonban mindenhova más komment kell, keressünk más megoldást.

A másik program (SOLARSOFT #413, ZIPNOTE) már segédállományokkal dolgozik, melyek közül az egyik feladata az, hogy benne tároljuk, illetve alakítsuk ki kedvünkre a ZIPkomment végső formáját a ZIP fájl kommentezése előtt. Sajnálatos módon többszöri próbálkozásra sem sikerült az általam megtervezett és elkészített logót ZIPkomment formájában a ZIP fájlhoz fűzni. Talán bennem volt a hiba, de mindenesetre furcsa, hogy a leírást betűről betűre követve sem sikerült eredményt elérnem a programmal.

Itt állok tehát egy megszerkesztett logóval, amit szeretnék az összes általam kiadott ZIP fájlra feltenni. Nincs megoldás? ABBS-ekből lehívott állományok jól megszerkesztett, terjedelmes logói azt tanúsítják, hogy van. Ezúton közreadom, mire jutottam én.

A NoGate Consulting cég PAK programja (SolarSoft, #431 lemez) azt nyújtja, amit vártunk tőle, egyéb egyedülálló szolgáltatásairól nem is beszélve.

Miért a PAK?

— Tetszőlegesen konvertálhatunk át a PAK, ARC és ZIP formátumú állományok között oda-vissza, valamint segédprogram nélkül generál PAK-SFX-et.

— A PAK nemcsak a parancssorban fogad el listafájlt, hanem a megjegyzések (fájl-, illetve archív komment) bekérésekor is. A listafájl és így a benne lévő megjegyzés hossza nem korlátozódik egyetlen sorra.

— Viszonylag egyszerűen, a PAKINST segédprogram és kedvenc szövegszerkesztőnk segítségével (feltéve, hogy az formátumozás nélküli tiszta ASCII fájl generál) konfigurálhatjuk a PAK programot arra is, hogy az archív megjegyzést mindig az általunk megadott állományból fűzze hozzá az archívhoz.

— Saját parancs- és kapcsolókészletet állíthatunk össze gyakoribb feladatainkhoz.

— Nemcsak ZIP komment, hanem fájl kommentek bevitelére is alkalmas.

Hogyan lássunk neki?

1. Hozzunk létre egy (lista) fájl, ami a (ZIP) kommentet tartalmazza majd, mondjuk a C:\ZIP könyvtárban FILEFEJ.DAT néven. Ebben készítjük el azt a logót, amelyiket a (ZIP) kommentbe szánunk. A fájl első sora legyen üres vagy legfeljebb 55 karakter hosszú (a PKZIP és a PKUNZIP ugyanis a kommentet nem a sor elejétől kezdi írni).

2. Ha még nem lenne a pathban, tegyük oda a PAK.EXE programot. Ha szükséges, konfigurálhatjuk a PAKINST programmal.

3. PAK h /z ZIPfájl.ZIP

(H a ZIPkomment bekérése, /z a ZIP forma megőrzése. Meg kell adni a kiterjesztést is, mert különben az azonos nevű ARC és PAK állományok is módosulnak.)

4. A program bekéri a ZIPkomment szövegét. Ide egy bevezető @ karakter után be kell írunk a ZIPkommentet tartalmazó állomány teljes nevét (a fenti példánál maradva: @C:\ZIP\FILEFEJ.DAT).

5. A fájlkommenthez a PAK r parancsa vagy az /r kapcsoló való:

PAK r ZIPfile.ZIP

Az eredeti PAK ilyenkor a ZIPkommentet is módosítja, de ez javítható (lásd alább). Használható még a PAK c /z /r ZIPfájl.ZIP fájlnev.kit szerkezet is a fájlkommentek bevitelére. Ilyenkor nemcsak kommentez, de át is csomagol.

Haladóbbak számára ajánlom, hogy a PAK programhoz definiáljanak új parancsokat és kapcsolókat saját feladataikra. Ennek a mikéntjéről bő információ található a PAK program ismertetésénél.

A bővített ZIPkomment

— Nem teljesen szabványos voltuk nagyritkán megzavarhatja az archív kezelő segédprogramok egy részét, így a ZIPVIEW programot is. A kibővített ZIPkommenttel ellátott állományoknak már a tartalomjegyzékét sem írja ki.

— Barátaim tapasztalata szerint a ZIPDMP nevű szeletelőprogram sem kedveli túlságosan a bővített megjegyzéseket.

Ilyen esetekben esetleg segíthet a felesleges ZIPkomment eltávolítása a PKZIP -z parancsával (a ZIPkommentet bekérő üzenetre legalább egy szóközt be kell írunk az <ENTER> megnyomása előtt, amivel felülírja a PKZIP az előző kommentet). Ha még mindig gondot okoz az adott ZIPfájl használata, akkor az azonnal kéznél levő PAK program CONVERT parancsával javíthatunk esetleg rajta, ha ez sem segít, akkor újra kell csomagolnunk a kérdéses ZIP állományt.

SNGLCMNT.EXE

Az átlagos kommentezési feladatokra majdnem olyan jó program, mint a PAK. Igaz, hogy nem szövegszerkesztővel készített állományból veszi a kommentet, viszont a

program szerkesztő ablakában (11 sorban maximum 75-75 karakter) tetszés szerint készíthetünk kommenteket a ZIP fájlhoz.

A programot kétféleképpen indíthatjuk: vagy már a parancssorban megjelölve a ZIP archívot, vagy paraméter nélkül. Az első esetben rögtön szerkeszthetjük is a ZIP fájl kommentet. A következő lehetőségeink vannak:

<ESC> — Quit

Kilépés a programból.

<F1> — Add

A megszerkesztett kommentet a PKZIP segítségével hozzáfűzi a ZIP fájlhoz.

<F2> — View

Meghívja a PKZIP-et a -v paranccsal és kiírja az ablakba a ZIPkommentet, valamint a ZIPkatalógust. Lapozni és visszagörgetni sajnos nem lehet, csak ellenőrzésre szolgál.

<F3> — File

Egyszerű rámutatással választhatunk a lemezen lévő ZIP fájlok közül. A könyvtárakat hasonlóan rámutatással váltja.

<F4> — Drive

Megadhatjuk az <F4> lenyomása után a választott meghajtó nevét.

<F5> — General Help

Egy általános ismertetőt ad a programról. Az <ESC> lenyomásával lehet a szerkesztő ablakba visszajutni.

<F6> — Edit Help

A szerkesztésben használható billentyűket sorolja fel. Igen hasonlít a Wordstarhoz.

<Ins> — Insert On/Off

A beszúró és a felülíró üzemmód váltogatását végzi.

<CTRL O><I> — Auto Indent On/Off

Az sorok automatikus beljebb kezdését be- és kikapcsolja.

<CTRL O><W> — Word Wrap On/Off

Az új sorba lépés funkcióját kapcsolja be és ki.

Az adatsűrítés éLHArcosa

A programot Haruyasu Yoshizaki (Yoshi) japán programozó írta egy másik japán tömörítő program a LARC sikerén felbuzdulva. A program első verziói még az LHARC nevet viselték, és ahogy tökéletesedtek az egymást szaporán követő verziók, úgy terjedtek el szerte a világon. A legismertebb verzió (legalábbis itt Magyarországon) az 1.13C.

A program további sorsa kalandosan alakult, egyszerre több néven is felbukkant ugyanaz a verzió. Az LHARC 1.14, az LHICE 1.14 és az ICE közötti különbség minimális. Például az ICE alapértelmezésként ICE kiterjesztést használ, de gond nélkül kezeli továbbra is az LZH kiterjesztésű archívokat. Ráadásul mindhárom program 1989 nyarán, egy hónapon belül készült.

A nagy ugrást a 2.xx verziók jelentették. A szerző immáron sokadszor újra átírta a programot, egy-két dolgot elhagyott, néhány hibát kijavított és egyes korlátozásokat feloldott, de a parancsokat, a szintaktikát és a kapcsolók többségét is meghagyta. Az új verzióval, amely az LHA nevet viseli, azért is egyszerűbb dolgozni, mert az ismertetések már könnyen emészthető formában írják le a program használatát. A program tulajdonképpen csak néhány lényeges ponton tér el az előző változatoktól:

— Nem lehet többé az s parancs mellett titkosító jelszót adni a /k kapcsolóval.

— Nem AUTOLARK.BAT az automatikusan elinduló csomag indító állománya, hanem !.BAT.

— Sajnos a Shez elterjedt 5.9-es verziójával még nem lehet használni, de az új, Shez 6.1 és 6.2 már ezt is megoldja. A MAD, az FV és az MTV, valamint a Norton Commanderhez írt ZIPVIEW is kezelni tudja. Az LHMENU és a CHECKOUT program csak akkor, ha az LHA.EXE állományt átnevezzük LHARC.EXE-nek.

— A vele készített SFX-et a MAD nem ismeri fel, az FV és a ZIPVIEW azonban igen.

A program mérete, egyszerű kezelhetősége azonban bőségesen kárpótol a veszteséget.

A.) Az LHA.EXE 2.12 használata

A program egyetlen állománya szolgál a bepakolásra, a tömörítésre, a kibontásra, sőt az önkinyitó archívok (SFX) létrehozására is. Szinte hihetetlen, hogyan lehetett mindezt ilyen kis helyen megoldani. (Ha a PKLITE programot ráeresztjük, még ez a 33965 bájtós méret is lecsökkenthető 25885 bájtra, anélkül hogy működésében változást észlelnénk. (Egyedül az LHA.EXE öntesztelése változik, mivel ez már nem az eredeti fájl.)

A többi archívkezelőhöz hasonlóan ennek a programnak is egybetűs parancsai vannak, pontosan egy tucat, melyek közül néhány azért került bele, mert a felhasználók megszoktak egyes parancsokat a többi archiváló programnál. Mivel egyszerre csak egy parancs adható ki, és a program megfelelő működéséhez gyakran van szükség paraméterezésre, 15 különféle kapcsolót használhatunk.

A parancs kiadásakor a következő alapszerkezetet használjuk:

LHA parancs [/opciók[módosítók]] archív [[bázis\] [fájlok]] [@listafájl]

LHA = Az LHARC program 2.1x-es verziója, a csomagban LHA.EXE néven.

parancs = A 12 lehetséges parancs egyike [a,f,d,e,l,t,u,m,p,x,v,s]. Ha egyetlen parancsot sem adunk ki, vagy egyéb, a program számára nem megfelelő karakter kerül a parancs helyére, akkor a Help képernyő jelenik meg a program parancsainak és opcióinak ismertetésével.

opciók = Más néven kapcsolók (switch). A program lehetséges 15 kapcsolója közül [r,w,x,m,p,c,a,z,t,h,o,n,i,l,-] beírható egyszerre több is. Az egyes kapcsolókhoz még további paramétereket adhatunk, ezeket lásd később, a kapcsolók részletes ismertetésénél. A kapcsolókat bevezető /, vagy - karakter után írhatjuk be a parancssorba.

módosítók = A kapcsolók hatását, működését befolyásoló paraméterek lehetnek szám [012], jel [-+], könyvtárnév (a /w opció után), illetve kiterjesztés (a /z kapcsoló után).

archív = Alapértelmezésként a program az LZH kiterjesztésű archívokkal dolgozik, de a belőle képzett SFX fájlt is tudja kezelni. Ekkor azonban meg kell adni a kiterjesztést.

bázis = Az alapkönyvtár (cél- vagy forráskönyvtár) neve, a végén egy \ jellel. Tartalmazhat természetesen meghajtónevet is.

fájlok = A be-, illetve kikapcsolandó állományok felsorolása. Ha nem adunk meg semmit, akkor az alapértelmezés *.*. A filenév lehet listafájl is egy bevezető @ karakter után.

B.) Az LHA 12 parancsa

Bepakoló parancsok

a (Add) — Tömörít és bepakol. Ha már volt az adott néven fájl az archívban, akkor az Add parancs hatására az LHA felülírja a régit, tekintet nélkül arra, hogy a forrás és a cél fájl közül melyik volt a frissebb. Új LZH archívot is létrehozhat.

u (Update) — Tömörít, aktualizál és bepakol. A megadott fájlokat az Add parancshoz hasonlóan tömöríti és bepakolja az LZH archívba, de csak akkor, ha a már bent lévőknél frissebbek. Új állományokat is bepakol, és új archívot is létre tud hozni.

f (Freshen) — Tömörít és frissít. A megadott fájlokat az Add és Update parancsokhoz hasonlóan tömöríti és bepakolja az LZH archívba, de csak akkor, ha a bentlévőknél frissebbek. Új állományokat NEM pakol be és új archívot sem tud létrehozni.

m (Move) — Tömörítve áthelyez. Az adott állományokat az Update parancshoz hasonlóan tömöríti és bepakolja az LZH archívba (csak frissebbeket és újakat visz be), majd végül a bepakolt forrásállományokat törli. Új LZH archívot is létrehozhat.

A fenti parancsok alapesetben csak a nevet írják be az archív katalógusba, az elérési utat a /x, vagy /x1 opció hatására teszik csak hozzá.

Az [ufm] parancsok dátumvizsgálata a /c0 kapcsolóval letiltható.

Az SFX fájlok bővítéséhez is használhatjuk közvetlenül a bepakoló parancsokat, ilyenkor azonban az SFX állomány kiterjesztését is meg kell adni az archív nevében.

Fontos tudni, hogy a rejtett és rendszer (hidden, system) attribútumú állományokat csak akkor látja az LHA, ha erre a /a+ kapcsolóval felkészítjük.

Kibontó parancsok

e (Extract) — A megadott LZH archívból a megadott állományokat bontja ki. Ha azonos nevű fájlt talál a célkönyvtárban, csak akkor bont ki, ha a bentlévő a frissebb. Az 1.1x verzióval készített LZH archívokat is kibontja, ha tömörítési kódjuk -lz5-, -lz4-, vagy -lz5-.

x (eXtract) — A teljes névvel tömörített állományok kibontására szolgál. Ha a névben esetleg szereplő alkönyvtár nem létezik, az LHA létrehozza, hacsak ezt nem tiltjuk a /p0 kapcsolóval. Figyelem! Egy hiba bennmaradt az LHA-ban! Ha az x parancs segítségével

kibontott állomány nem fér el a céllemezre, az LHA nem veszi észre, és nem állítja be az errorlevel értéket. Ha nem ellenőrizzük, komoly adatvesztést is eredményezhet.

Mindkét kibontó parancs használata előtt ajánlatos megnézni, hogy a kibontandó állományok beleférnek-e a célkönyvtárba. Ugyanis ha kibontáskor a céllemez megtelt, az LHA sajnos kiakadhat, mert vég nélkül próbálkozik az állományok kibontásával. Ez főleg a régebbi PC és XT gépeken okozhat problémát, ahol nincs BREAK billentyű.

A kibontó parancsok csak akkor tudják visszaállítani a kibontott állományok attribútumait (hidden, system, readonly), ha ezt a /a+ kapcsoló alkalmazásával kérjük.

p (Print) — Alapesetben a képernyőre bontja ki az LHA az állományokat. A DOS átirányító > parancsával azonban a nyomtatóra is terelhetjük a kibontott állományt. A nyomtatót nem tudja a program a PKUNZIP-hez hasonlóan átkapcsolni, sem a grafikus (bináris), sem a szöveges (text) üzemmódba, és a kimeneti portot (LPT1:, LPT2:, COM1, COM2, stb.) sem lehet a programból kijelölni.

t (Test) — teszteli az LZH archív épségét. A parancs kiadásakor az LHA egy CRC ellenőrzést végez. Az LHA.EXE ilyen ellenőrzésekor (LHA t LHA.EXE) az alábbi üzenetet adja:

„This file seems to be ORIGINAL distributed from H. Yoshi” — azaz:
„Ugy tűnik, ez a fájl eredeti, H. Yoshi-tól származik.”

Felesleges állományok eltávolítása

d (Delete) — A megadott állományokat az LZH egyszerűen törli az archívból.

Az archív katalógus kiírása

l (List) — Az archív katalógusát kiírja. Az archív katalógus az LZH archív állományok tartalomjegyzéke. Ha a fájlnev előtt + jel áll, akkor a fájl teljes névvel lett elpakolva. Olyankor is ez az alapértelmezés, amikor nem adunk ki semmilyen módosító parancsot. A /x kapcsolóval a teljes nevet kiírathatjuk, de az már állományonként két sort ír ki.

v (View) — a teljes nevet tartalmazó archív katalógust írja ki. A List parancs /x opcióval bővített használatával egyező a hatása. A fájlok elérési útját s a neveket az adatokkal fájlanként két-két sorban jeleníti meg.

SFX fájl készítése

s (Self-extract) — Az kész LZH archívból önkibontó (SFX) fájlt csinál. Az alapértelmezés az /x0, ha nem írunk mellé egyéb opciókat. Az így készített fájl ugyan kisebb, mint az /x1 kapcsolóval készült, de kevesebbet is tud (a Small model 1,5 K-val, a Large model 1,9 K-val növeli meg az eredeti LZH archív méretét. Csak a Large model (az /x1 kapcsolóval létrehozott) SFX fájl képes létrehozni a célkönyvtár(ak)at, s elindítani automatikusan az archív belsejéből kipakolt !.BAT fájlt.

Egy megszívlelendő tanács: a létrehozott SFX állományok nevében célszerű valami módon jelezni, hogy nem egy azonnal működtethető állományról van szó. Ajánlom erre például a Solarsoft lemezeknél is alkalmazott módszert, hogy az SFX állománynév utolsó karaktere a # jel.

C.) Az LHA kapcsolói (opciók)

A kapcsolók előtt a „/”, esetleg a megszokott „-” jelet kell kitennünk. Egy parancssorban több kapcsolót is használhatunk egyidejűleg, amelyeket a parancsot követően többnyire tetszés szerint helyezhetünk el. A leggyakrabban használt kombinációt érdemes lehet az

LHA környezeti változóban tárolni (lásd ott). A kapcsolókat írhatjuk külön-külön, egy-egy bevezető / jel után, vagy egy közös bevezető / karakterrel egy csomagba. Az ilyen csomagok végére kell tenni a /w és /z kapcsolókat.

Minden egyes kapcsolót (elméletileg) háromféle számértékkel (0/1/2) finomíthatunk tovább, azonban a kapcsolók egy részében az 1 és 2 érték azonos hatású. A 0 és 1 értékek között a „+” és „-” jelekkel is átkapcsolhatunk, hasonlóan az LHARC113 programhoz. Két kapcsolónak (/w és /z) szöveges kiegészítése is lehet (lásd ott).

/x[+|-|0|1] (eXtend) — A teljes nevek (path is) használatát engedélyezi. A 0 ki, az 1 bekapcsolja. A bepakoló [aufm], archív katalóguskiíró [lv] és kibontó parancsoknál [ex] használatos. Az Extract parancs mellett /x0 az alapértelmezés, az eXtract parancs mellett pedig a /x1.

A Self-extract (s) parancs mellett egy kicsit más az /x szerepe. Itt azt állítja be, hogy a létrejövő SFX fájl a Small model (/x0), vagy Large model (/x1) legyen. Ez az utóbbi valamivel nagyobb, de ennek megfelelően többet is tud (1,9 illetve 1.5 K az LZH archív növekedése).

/p[+|-|0|1|2] (Precise) — Kibontáskor a megfelelő path-ra teszi a teljes névvel elmentett, különböző könyvtárakból származó azonos nevű fájlokat. Olyankor hasznos, ha egy archívon belül van két azonos nevű, de különböző könyvtárakból származó bepakolt állományunk. A /p nélkül a másodiknak kinyíló fájl nagy valószínűséggel felülírná az elsőt, míg a /p-vel minden a helyére kerül. Természetesen letiltható ez is a /p0 kapcsolóval. Az alapértelmezés a /p1.

/c[+|-|0|1|2] (ignore Comparison of time) — Ha ezzel az opcióval használjuk, az [ufex] parancsoknál sem nézi a keletkezési dátumot, hanem egyszerűen felülírja az azonos nevű állományokat. Az alapértelmezés a /c0.

/m[+|-|0|1|2] (no Message) — Üzenet nélkül kibontja a megadott nevű állományt a célkönyvtárba egy 000 és 999 közötti még nem használt kiterjesztéssel, ha az „e” parancsot használtuk előtte.

/a[+|-|0|1] (any Attribute) — Alapértelmezésként az LHA nem archiválja a hidden, system és readonly attribútumu állományokat. Az /a1 kapcsoló engedélyezi ezen állományok archiválását. Kibontáskor az LHA a /a1 kapcsolóval visszaállítja az archivált állományok bármely attribútumát kipakolásakor.

/r[+|-|0|1|2] (Recursive) — A /r kapcsolóval rekurzív pakolást, illetve kibontást végezhetünk. Ez annyit jelent, hogy nemcsak a megadott forráskönyvtárból, hanem az abból eredő alkönyvtárakból is pakolja a megfelelő állományokat. Három üzemmódja is van a kapcsolónak:

/r0 Alapértelmezés. Nem rekurzív, csak a megadott állományokat pakolja.

/r1 Rekurzív keresés, de csak a fileneveket tárolja. A megadott állományokat bepakolja a forráskönyvtárból. Annak alkönyvtáraiból azonban nem menti el a névvel együtt a path-t, a struktúrát.

/r2 Rekurzív keresés, a könyvtárszerkezetet is megjegyzi. A forráskönyvtár szerkezetét eltárolja úgy, ahogy van.

Míg az LHARC 1.13C programnál az /r kapcsoló az /x-et is automatikusan kapcsolta, itt nekünk kell erről gondoskodni. Az kapcsoló egyik leggyakoribb használata egy floppy teljes bepakolása egy LZH archívba:

LHA a /r2x1 archív.LZH a:\

illetve a kipakolás:

LHA x archív.LZH b:\

/t[+|-|0|1] (Time stamp) — A bepakoló és törlő [aufmd] parancsok mellett a /t1 az LZH archív idejét a legfrissebb pakolt állományhoz igazítja. Az alapértelmezés a /t0, amikor ez az idő az LZH archív keletkezési ideje lesz.

/w[+|-|0|1|directorynév\] Ezzel a kapcsolóval megadhatjuk, hogy a program hova írja átmeneti állományait. Az alapértelmezés általában ugyanaz a könyvtár, ahol az új LZH archív felépül. Ha /w+ kapcsolót használunk, akkor az a könyvtár lesz a munkakönyvtár, ahonnan az LHA-t meghívtuk. Ha definiáltunk előzőleg (pl. AUTOEXEC.BAT-ban) egy TMP nevű környezetleíró változót, akkor automatikusan /w1 a kapcsoló alapértelmezése.

Különösen akkor hasznos, ha a céllemezen nincs elég hely, esetleg lehetőségünk nyílik egy Ramdrive, vagy winchester használatára. Ha a /w után könyvtárnevet adunk meg, azt egy \ jellel kell lezárunk.

Arra azonban vigyáznunk kell, hogy bepakolásakor komoly adatvesztés léphet fel, ha a /w kapcsolóval megadott munkakönyvtárba nem fér be az átmeneti állomány. Ezt sajnos nem veszi észre a program és a csonka, hiányos átmeneti állományból készíti el az új archívot.

/n[+|-|0|1|2] (No indicator) — Az LHA működését jelző kiírásokat lehet letiltani ezzel a kapcsolóval:

/n1 A kibontási és bepakolási folyamatot kísérő, bogyózó „oooo...” jeleket tiltja le.

/n2 A fájlnevek és a tömörítés kiírását tiltja le.

/n0 Nem tiltja le egyiket sem.

/z[+|-|0|1|2|[ext]] (Zero compression) — Tömörítés nélküli archiválást tesz lehetővé. Üzem módjai:

/z0 Az alapértelmezés. Bepakolásakor mindent tömörít

/z1 Egyik becsomagolt állományt sem tömöríti.

/z2 Az alábbi kiterjesztésű állományok kivételével mindent megpróbál tömöríteni: ARC, LZH, LZS, PAK, ZIP, ZOO.

/z A kiterjesztés nélküli fájlokat nem tömöríti.

/zXXX A megadott kiterjesztésű állományokat nem tömöríti. (A [/zXXX] kapcsoló többször is szerepelhet egy parancssorban.)

/o[+|-|0|1] (Old compatible compression) — Az LHARC 1.13C-vel kompatibilis régi formátumú LZH archívot csinál. Az LHA még ebben az esetben is tömörebb, mint az LH113C v.1.xx. A header automatikusan beáll /h1-re. (Ne legyünk telhetetlenek! Az LHARC 1.13C nem biztos, hogy ki tudja bontani ezt az állományt, de minek is kellene használnunk ezt az elavult verziót.)

/h[+|-|0|1|2] (Header level) — Header level választása. Az alapértelmezés /h1.

/i[+|-|0|1] (don't Ignore case) — Megkülönbözteti a nagy és kisbetűket. Az LZH archívoknak van egy közös fejlécformátumuk, azokra az esetekre, amikor a kis- és nagybetűk nem egyenértékűek. Ezáltal más operációs rendszerekben készült LZH archívok is kibonthatók DOS alatt. A DOS verziójú LHA archívok nem tesznek különbséget a kis- és a nagybetűk között, de saját maguk a neveket csupa nagybetűvel tárolják.

/l[+|-|0|1|L] (Long display) — A fájlnevek megjelenítési formáját befolyásolja a be- és kipakolásakor:

/l0 Csak a fájlnevek.

/l1 Teljes tárolt, illetve tárolandó nevek, két sorban, a path-szal együtt.

/l2 Az LHA által éppen feldolgozott állományok teljes neve jelenik meg a képernyőn.

/-[01] — Ha a /-1 kapcsolót használjuk, akkor „-” és „@” jeleket a fájlnevek részeként fogja kezelni. Alapértelmezésként az LHA számára egy „@” karakterrel kezdődő név

listafájlt jelent. Ebből az üzemmódból a /-0 vihet ki. Ebben az egy különleges esetben a -- 0 nem használható.

D.) Az LHA környezete

Az alapkönyvtár (base directory)

Az alapkönyvtár nem az aktuális könyvtár, de nem is a gyökér könyvtár, hanem az, ahol az LHA végrehajtódik. Nemcsak egy, hanem több alapkönyvtárat is megadhatunk egyszerre a parancssorban. Természetesen nem lehet egyetlen állományt egyszerre több alapkönyvtárba kinyitni. Az elsőnek megadott könyvtárnak elsőbbsége van.

```
d:\>LHA x archív c:\bin\*.exe *.com c:\temp\*.man *.doc
```

Ez egyenértékű az alábbi parancssorozattal:

```
d:\>c:
```

```
c:\>cd \bin
```

```
c:\bin>LHA x d:\archív *.exe *.com
```

```
c:\>bin>cd \temp
```

```
c:\temp>LHA x d:\Archív *.man *.doc
```

A választás kézenfekvő, hiszen ki szeret feleslegesen írni.

Környezetleíró változók

LHA (Verzió) — Az alapértelmezéseket állíthatjuk be vele. Például, ha a SET LHA=/wc:\temp /x1r2a1l2z0 paranccsal (mondjuk, az AUTOEXEC.BAT-ban) értéket adtunk az LHA változónak, akkor az itt megadott kapcsolások lesznek az alapértelmezések.

TMP (Munkakönyvtár) — A munkakönyvtárat állítja be. Ha pl. az AUTOEXEC.BAT-ban már hozzárendeltünk a változóhoz egy meghajtót és egy könyvtárnevet, akkor az LHA programmal való munkáknál nem szükséges a /w kapcsolót használnunk a munkakönyvtár megadásához, mert az alapértelmezésként a TMP változó tartalmával megegyezik. Többek közt a Microsoft Word is ugyanezre a célra használja a TMP-t.

TZ (Time Zone) — Amennyiben -h2- header level-lel akarunk archiválni, illetve kibontani, akkor a Time Zone értéket is be kell állítanunk (pl. EST, PST, stb.). Amerikában pl. a keleti időzónában a DOS SET TZ=EST+5 paranccsal. A header level alapértelmezése -h1-.

A listafájl és a DOS átirányító parancsal

A DOS parancssora csak korlátozott hosszúságú lehet a DOS közismert korlátai miatt. Ennek kiküszöbölésére a legtöbb profi kivitelű program lehetővé teszi listafájl alkalmazását. Természetesen az LHA is ezek közé tartozik.

Aki már dolgozott PKZIP-pel, vagy fordító programmal, annak semmiképp nem lehet idegen ez a szolgáltatás. Itt is egy „@” jelet kell a listafájl neve elé írni a parancssorban, amiről a program majd felismeri, hogy mire való.

A listafájl szerkezete egyszerű: közönséges ASCII szövegfájl. Bele kell írni azt, amit különben a parancssorba írnánk, ha elférne abban: a kapcsolókat, azok módosítóit, a választott alapkönyvtárat (ha vannak), és a kezelendő fájlok neveit. A listafájl figyelmen kívül hagyja a CR kódot és szóközöket használ elválasztó karakterként.

Az egyetlen megszorítás, hogy a listafájlokat nem lehet egymásba ágyazni, vagy egy másik listafájlt meghívni listafájlból.

Listafájlokat tetszőleges, formázatlan ASCII fájlokat készítő szövegszerkesztővel lehet előállítani (a DOS Edlin programját csak mazochistáknak ajánlom). A másik lehetőség

a DOS COPY CON listafájl parancsa, ha van elég bátorságunk a listafájl egyből való megírásához. Igen jó módszer a listafájl készítésére, hogy a bepakolandó fájlok könyvtárainak katalógusát egy fájlba irányítjuk a DOS „>” vagy „>>” operátoraival és azt szerkesztjük át listafájllá, illetve az LZH archív archív katalógusát is átirányíthatjuk egy később megszerkeszthető állományba. Pl.:

```
dir /w >lista1.lst
dir /w >>lista2.lst
LHA 1 /n1 Archív >lista3.lst
```

Mivel az LHA nem rendezi a fájlokat, ha név szerint sorba rendezett listákat akarunk, akkor pl. a DOS SORT parancsát használhatjuk. Pl.:

```
dir /wlsortl >lista.lst
```

Ezt a lista.lst fájl listafájlként használhatjuk bepakolásakor.

Errorlevel értékek

A program természetesen batch állományokból is futtatható, s a futás ellenőrizhető az errorlevel változó értékének a vizsgálatával:

Errorlevel Jelentése

- 0 Normál futás volt, hiba nélkül.
 - 1 CRC hiba, főleg az [e], [x], [t] parancsoknál lehet.
 - 2 Végzetes hiba volt. A folyamat megszakadt.
 - 3 Nem tudta az átmeneti állományokat archívvá írni.
- Megtalálhatjuk az LHTMP)2(.LZH állományt a munka könyvtárban.
Ezt átnevezhetjük és használhatjuk LZH kiterjesztéssel archívként.

Átmeneti munkaállományok

Az LHA a régi archívot (LZH archív, SFX fájl) LHTMP)1(.LZH névvel látja el munkája közben. Az új állomány az átnevezés előtt az LHTMP)2(.LZH nevet viseli. Az átmeneti állományok átmenetileg rendszerint az alap- vagy célkönyvtárban jönnek létre hacsak a TMP változóval, vagy a /w kapcsolóval ki nem jelölünk egy munkakönyvtárat (lásd /w+ és /wkönyvtár).

D.) Önkibontó (SFX) állományok

Az önkinyitó állományok a már korábban elkészített LZH archívokból az LHA s parancsával készíthetők el. Ha az SFX fájl lehetőségeit teljes mértékben ki akarjuk használni, akkor az s parancs mellett a /x1 kapcsolót is be kell írni a parancssorba.

Az SFX fájl két különleges állományt használ, ezeket a többi belepakolt állományhoz képest eltérően kezeli. A !, azaz a telop fájl az első, a másik az automatikus indítást végző !.BAT fájl.

A telop (!)

A „telop” egy közönséges ASCII fájl, egy képernyőnyi szöveggel, kiterjesztés nélkül, ! névvel. Ebbe az állományba írhatjuk üzeneteinket, azt amit az önkinyitó állomány címzettjeivel akarunk közölni. Általában a bepakolt programokkal, a program készítőjével, esetleg a kinyitással, vagy a csomag terjesztőjével kapcsolatos adatok szoktak benne lenni (rövid programismertetés, címek, telefonszámok, néha személyes üzenetek). Vagyis a telop valamiféle eligazítás, üzenet, „read me” (olvass el), tájékoztató. Akinek van rá találó magyar elnevezése, attól szívesen átvesszük és használjuk. Az üzenetek legvégére

azt a kérdést szokás betenni, hogy kipakolja-e magát az SFX fájl, vagy sem. Ha Y-nal válaszolunk, akkor szépen kipakol, ha N lesz a válasz, akkor kipakolás nélkül visszajutunk a DOS-ba.

Fontos: az LZH archívba pakolásakor legelőször a telop-ot (!) kell bepakolni, ha azt akarjuk, hogy a többi fájl kinyitása előtt megjelenjen, mivel az LHA és az SFX fájl is csak a bepakolás sorrendjében nyit.

Automatikus programindítás

Ez a lehetőség csak /x1 kapcsolóval létrehozott önkinyitó SFX fájlok tulajdonsága. Ha az SFX fájlba egy !.BAT állományt is bepakoltunk, akkor a /! kapcsolóval való indításkor a kibontás után átadódik erre az állományra a vezérlés. A !.BAT fájlba érdemes beépíteni egy hibaellenőrzést, pl.:

```
if errorlevel 3 goto hiba3
if errorlevel 2 goto hiba2
if errorlevel 1 goto hiba1
```

Az SFX fájlok két típusa

A kisebbik (small model)

Az alapértelmezés a small model előállítására az SFX fájlt készítő s parancs használatakor az /x kapcsoló nélkül, vagy /x0 kapcsolóval. Ez a változat méretben a kisebb, de sajnos tudásban is. Nem foglalkozik a könyvtárszerkezettel, még ha az benne is volt a forrás LZH archívban. Csak a pusztán nevet tartalmazza, path nélkül, ezért elindításakor az aktuális könyvtárban nyitja ki a benne lévő állományokat és még csak célkönyvtár sem adható meg. Igaz ugyan, hogy az LHARC 1.13 változatnál kényszerűségből eltűrt méretkorlátozások már megszűntek, de továbbra sem indítható belőle automatikusan a !.BAT állomány. Kiterjesztése .COM vagy .EXE (mintegy 1,5 K az SFX fájl növekedése a forrás LZH fájlhoz képest).

A nagyobbik (large model)

A large model mindig .EXE állomány, amely nagyobb is lehet, mint a rendelkezésre álló RAM, csak a lemezkapacitás szab neki határt. Előállításához az s parancsot az /x1 kapcsolóval kell használnunk amikor az SFX fájlt hozzuk létre (amely mintegy 1,9 K-val nagyobb, mint a forrás LZH archív).

Természetesen lehet benne és működik a telop (! fájl) és az automatikus indítást lehetővé tevő !.BAT állomány is. Archív katalógusában teljes fájlneveket őriz, könyvtárszerkezet visszaállítását is el tudja végezni.

A large model SFX fájl kapcsolói, paraméterei:

Kapcsoló/ Paraméter hatása

/x Nem hoz létre új könyvtárat, alkönyvtárat.

/! Elindítja a kipakolt !.BAT-ot a teljes kibontás után.

/e[cén] Célkönyvtár megadása kapcsolóval.

[cén] Célkönyvtár megadása paraméterként.

Ha a célkönyvtárban már van egy !.BAT fájl, akkor azt az SFX fájl kibontáskor felülírja, a vezérlés mindenképpen a frissen kipakolt indító !.BAT-ra kerül, ha /! kapcsolóval hívtuk meg az SFX fájlt.

A telop ! fájlt egyik típusu SFX fájl sem nyitja ki a célkönyvtárban, az csak az LHA.EXE segítségével pakolható ki szerkeszthető formában (LHA e SFX_fájl !).

A másik megoldás az SFX fájl képernyőkimenetének a DOS >, vagy >> operátoraival történő átirányítása fájlba.

E.) Az LHA a mindennapi gyakorlatban

Aml az LHA-ból kimaradt

Az LHARC 1.13, LHARC 1.14 és LHICE s parancsa mellett használható és használandó kapcsoló a /k[jelszó]. Ha alkalmazzuk, akkor a létrehozott önkinyitó SFX fájl egy újabb paraméterrel bővíthető, mégpedig az AUTOLARK.BAT-ot a kibontás után elindító jelszóval. Ehhez természetesen az AUTOLARK.BAT- nak már bent kell lennie az SFX fájlban.

Az LHA által létrehozott LZH archívokat látja és kezeli az FV, az MTV, a MAD és sok egyéb program. (A SHEZ 5.9 még nem.)

A ZIPVIEW.EXE nevű, a Norton Commanderhez készült bekukkantó program megmutatja az archív katalógust, bele is olvashatunk vele az LHA-val bepakolt szöveges állományokba, sőt ki is pakolhatjuk az archívot, azzal a megszorítással, hogy az önkinyitó állományokból mindent kibont.

Tippek, trükkök

Az SFX fájlokat lehet módosítani ugyan közvetlenül is az LHA-val, de biztonságosabb a kibontott SFX fájlokat külön szerkesztgetni, módosítani, sorbarendegetni, majd visszapakolni, és azután az s paranccsal egy új SFX fájlt csinálni.

Az LZH-SFX-ek, mint általában az SFX állományok, igen hasznosak olyankor, amikor nincs kéznél semmilyen kinyitó program, vagy nem akarjuk az SFX címzettjét egy külön segédprogram rászabadításával terhelni.

Az SFX állományok ajánlott használata:

1. Ha nincs, hozzuk létre a célkönyvtárat, amennyiben külön könyvtárban akarjuk elhelyezni a kipakolt állományokat. (Pl.: md c:\uj\kipakolt)

2. Lépünk a célkönyvtárba/meghajtóra. (Pl. C:\> cd \uj\kipakolt)

3. Hívjuk meg az SFX-et, megadva annak elérési útját is. (Pl. C:\UJKIPAKOLT \>A:ZIPSFX)

Természetesen ettől el is térhetünk, de ez a legegyszerűbb és a legbiztonságosabb használata az SFX állományoknak.

Az LZH-SFX értelmezni tud néhány kapcsolót, ha a nagyobbik változatról (large model) van szó. Ilyenkor az LZH-SFX állományok parancssorának szerkezete:

LZH-SFX [/x] [!/] [/célkönyvtár] [célkönyvtár]

A /x kapcsoló hatására nem hoz létre új alkönyvtárakat az SFX.

A /! kapcsoló alkalmazásakor az SFX átadja kibontás után a vezérlést a frissen kipakolt !.BAT fájlra, így könnyen automatizálhatunk sok mindent.

A célkönyvtár kétféleképpen is megadható, az /e kapcsolóval, illetve kapcsoló nélkül.

Az LHA használatakor a következő szerkezetet alkalmazzuk:

lha parancs [kapcsolók] archív [bázis] [maszkok] [kimenet]

Az LHA.EXE programot célszerű egy path-on lévő könyvtárba tenni, ahol a DOS könnyen megtalálja és nem kell az elérési útvonallal bíbelődni a program minden egyes meghívásánál.

Az előző részekben ismertetett parancsok közül egyet be kell írunk az LHA mögé, attól egy szóközzel elválasztva. Ha az LHA által nem értelmezhető parancsot írunk be,

akkor a Help jelenik meg, ami az LHA parancsait, kapcsolóit és szintaxisát ismerteti. A kapcsolók használata nem kötelező, leírásukat lásd az előző oldalakon.

Az archív nevét mindig meg kell adni. Az Add, Update, Freshen, Move és Delete parancsok használatakor az archív nevében nem szerepelhetnek dzsóker (* és ?) karakterek, az összes többi parancs használatakor igen. Az archív neve természetesen tartalmazhat elérési utat is.

A bázis bepakolásakor a forrás, kipakolásakor a célkönyvtár. Egy parancssorban több báziskönyvtár is megadható a hozzá tartozó fájlmaszkkal. Egy-egy bázis hatása addig érvényes, amíg a parancssorban vagy a listafájlban újat nem adunk

A maszkok lehetnek fájlnevek, melyekben szabadon használhatjuk a dzsóker (* és ?) karaktereket, tartalmazhatnak elérési utat is, mind a be-, mind a kipakoláshoz. A maszkok között listafájlt is megadhatunk, ennek ismertetését lásd a listafájl ismertetésénél részletesen, valamint a gyakorlati példák között.

A kimenetet nem maga az LHA program, hanem a DOS használja. Segítségével átírányíthatjuk a különben a képernyőn megjelenő szövegeket egy megadott fájlba, vagy nyomtatóra (lásd nyomtatás). Hasznos ez a lehetőség, amikor például az archív katalógust ki akarjuk nyomtatni, vagy egy listafájllhoz akarjuk felhasználni, esetleg munkánkat és az LHA üzeneteit akarjuk dokumentálni. Az átírányításhoz a DOS > és >> operátorait használjuk.

Gyakorlati feladatok, megoldások

1. A C:\MUNKA\KONYV könyvtár összes állományából egy gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, de nem akarjuk a könyvtár tartalmát elveszíteni.

```
C:\MUNKA>LHA a a:munka c:\munka\konyv\
```

2. A C:\MUNKA könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az archív elkészültével ki akarjuk üríteni a könyvtárat.

```
C:\MUNKA>lha m a:munka c:\munka\konyv\
```

3. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az egész C:\MUNKA\KONYV alkönyvtárt a pakolás után ki akarjuk törölni.

```
C:\MUNKA>lha m a:munka c:\munka\konyv\
```

```
C:\MUNKA>rd konyv
```

4. Csak a napi munka során megváltozott állományokat akarjuk az A:-ban lévő lemezen tárolt MUNKA.LZH fájlban egy kicsit felfrissíteni.

```
C:\MUNKA>lha f a:munka c:\munka\konyv\
```

5. A C:\MUNKA és a C:\MUNKA\KONYV könyvtárakból minden .DOC kiterjesztésű állománnyal aktualizáljuk az A:MUNKA.LZH állományt. A forrásállományokat nem akarjuk kitörölni.

```
C:\MUNKA>lha u /x1r2 a:munka c:\munka\*.doc
```

6. A C:\MUNKA\KONYV könyvtárból minden nem .COM kiterjesztésű állománnyal bővítjük az A:MUNKA.LZH állományt. A forrásállományokat nem akarjuk törölni.

```
C:\MUNKA\KONYV>no *.com lha a /a- a:munka
```

7. A MUNKA.LZH-ba tévedésből bekerült .BAK, .LZH és .~DO kiterjesztésű állományokat kell eltávolítanunk.

```
C:\MUNKA>lha d a:munka *.bat *.~do *.lzh
```

8. Meg akarjuk nézni, mi van a MUNKA.LZH állományban.

```
C:\MUNKA>lha l a:munka (Csak tömör listát ad)
```

```
C:\MUNKA>lha v a:munka (Teljes listát ad)
```

9. Ki akarjuk listáztatni a MUNKA.LZH tartalmát egy szabadon szerkeszthető ASCII állományba, de csak a nevekre és a kiterjesztésekre vagyunk kíváncsiak. (Pl. egy listafájl készítését tervezetjük.)

```
C:\MUNKA>lha l a:munka lista.lst
```

10. Gyanús a lemez. Ellenőrizni akarjuk a MUNKA.LZH épségét.

```
C:\MUNKA>lha t a:munka
```

11. Egy fura ötlettől vezérelve az A:-n lévő MUNKA.EXE SFX-ből akarjuk futtatni a Norton Editort.

```
C:\MUNKA>a:munka /!
```

(Ha előzőleg elhelyeztünk egy !.BAT állományt az SFX-ben, ami azután meghívja a Norton Editort.)

12. Meg akarjuk nézni az egyik (pl. az LHA.DOC) állományt.

```
C:\MUNKA>lha p a:munka lha.doc
```

13. Megtetszett a fenti lha.DOC. Nosza nyomtassuk csak ki.

```
C:\MUNKA>lha p a:munka lha.doc >prn
```

14. A MUNKA.LZH-ből a .TXT kiterjesztésű állományok a C:\TXT könyvtárba, a .DOC kiterjesztésűek a C:\DOKSI könyvtárba pakolandók ki.

```
C:\MUNKA>lha e a:munka c:\tx\ *.txt c:\doks\ *.doc
```

Az LHARC előző verzióiról

Yoshi mester remek kis programjának legelterjedtebb verziója — legalábbis Magyarországon — az LHARC 1.13C. Ez a programocska szinte mindent ugyanúgy csinál, mint az LHA.EXE, ezért nem érdemes az összes parancs, kapcsoló és egyéb paraméter leírását megismételni, csak a főbb különbségeket mutatom be:

— Az s (Self-extract) parancs az /x kapcsoló nélkül .COM állományt hoz létre, ha az nem lépi túl a 64 K-s határt.

— A large model SFX-ben az elinduló állomány AUTOLARK.BAT nevet viseli.

— Az önindító SFX készítésekor nem elég az indító batch fájl, még egy jelszót is kellett adni /k<jelszó> alakban az s parancs után. Az így elkészített SFX az alábbi szerkezet szerint indítva egyből futtatja is az indító állományt:

```
SFXfájl /<jelszó>.
```

Mivel a 2.1x-es verzió sokat javult és tisztult az 1.xx verziókhöz képest, minden LHARC-ot alkalmazó felhasználónak javaslom, térjen át az LHA 2.1x verzió használatára. A LHA 2.10 csomag leírással együtt szabadsoftverként megtalálható az Alaplap 1991/júliusi számának lemezmellékletén, illetve a SolarSoft #509-es lemezén.

Ahogy én PAKolok...

A NoGate Consulting csúcsterméke, a PAK 2.50 jelen pillanatban a fájl-tömörítők mezőnyében dobogós helyen áll. A program régebbi (1.xx) verziói még ARC kiterjesztésű állományokkal, az újak PAK kiterjesztésűekkel dolgoznak. A szervesen beépített konverziós rutinok segítségével könnyedén kezeli a három legismertebb és legelterjedtebb archív állománytípust, az ARC, a PAK és a ZIP fájlokat. A belőlük képzett önkinyitó (SFX) archívokkal is jól boldogul. A program ezen szolgáltatásával egyedülálló a tömörítő programok körében. Emellett az SDN formátumú állományokat is ki tudja csomagolni, természetesen a megfelelő ellenőrzések elvégzését követően.

A PAK program másik különlegessége, hogy önkinyitó állományokat nemcsak már előre elkészített archívokból képes előállítani, hanem egy lépéses konverzióval a tömörítetlen forrásállományokból közvetlenül is.

A program harmadik különlegessége, hogy a PAKINST.EXE segédprogram (a PAK 2.10 csomag része) és egy formázatlan ASCII szövegállományokat produkáló szöveg-szerkesztő (pl. Norton Editor, Word, SideKick, Kedit stb.) segítségével saját igényeink szerint alakíthatjuk át a program parancskészletét és kapcsolóit, vagy definiálhatunk új, összetett parancsokat is, akár egybetűsre, akár beszélő nevekkal.

A program legfontosabb parancsai megfelelnek a tömörítők egyezményes jeleinek:

a = Add	= Bepakolás, valamiféle tömörítéssel.
f = Freshen	= Frissítés, újat nem visz be.
u = Update	= Aktualizálás, újat is bevisz.
m = Move	= Áthelyezés (a forrásállományok törlésével).
d = Delete	= Fájl törlése az archívból.
e = Extract	= Kibontás, visszaállítás.
x = eXtract	= Kibontás, visszaállítás. (Move kifelé.)
t = Test	= Archív épségét teszteli.
p = Print	= Képernyőn/printeren megjelenítés.
c = Convert	= Átalakítás más archív formátumba.
r = Revise	= Megjegyzések karbantartása.
h = Header	= Archív komment karbantartása.
l = List	= Archív katalógus kiírása.
v = View	= Bővített archív katalógust megjelenítése.

A program támogatja a listafájl alkalmazását is.

Bepakolás

a (= Add)

Alapértelmezésként az aktuális könyvtár, vagy az adott forráskönyvtár minden állományát bepakolja, ha nem adunk meg fájlazonosítót. Társítható szinte valamennyi kapcsolóval.

m (= Move)

Megegyezik az Add paranccsal, kivéve, hogy a sikeres bepakolás után törli a bepakolt forrásállományokat. Társítható szinte valamennyi kapcsolóval.

u (= Update)

Megegyezik az Add paranccsal, azzal az egy eltéréssel, hogy ellenőrzi a bepakolandó fájl keltezését és csak akkor pakolja be, ha az frissebb, mint a már bent lévő azonos nevű állomány, vagy új, és még nem szerepelt az aktualizálandó PAK fájlban. Társítható szinte valamennyi kapcsolóval.

f (= Freshen)

Megegyezik az Update paranccsal, azt kivéve, hogy új állományokat nem pakol be. Társítható szinte valamennyi kapcsolóval.

A fenti parancsok mellett leggyakrabban a következő kapcsolókat alkalmazzuk:

/I (= Include subdirectories)

A megadott forráskönyvtár alkönyvtáraiból is kimazsolázza a fájlmaszknak megfelelő állományokat.

/exe (= make EXE SFX)

Önkinyitó EXE állományt hoz létre a forrásállományokból.

/m (= Move)

Az Update és a Freshen parancsokkal ajánlatos kombinálni, ekkor a sikeres bepakolást követően a PAK kitörli a már bepakolt forrásállományokat.

/g=<jelszó> (= Garble)

A jelszavas titkosítást biztosító kapcsoló. Az így bepakolt állományokat kikapolni, megnézni, nyomtatni is csak a jelszó megadásával lehet. Az egyenlőségjel is része a kapcsolónak.

/p (= Pack archives)

A tömörített archív állományokat ritkán lehet tovább zsugorítani, ezért a PAK ezzel nem is próbálkozik, ami jelentős időmegtakarítással jár. Ha azonban azt akarjuk, hogy a PAK próbálkozzon meg az archívok további tömörítésével is, akkor használjuk a /p kapcsolót.

(A többi kapcsoló hatását lásd külön részletesen.)

Kipakolás, tesztelés, betekintés, nyomtatás, törlés

e (= Extract)

A PAK fájlban lévő, a megadott fájl-specifikációknak megfelelő állományokat kibontja a megadott célkönyvtárba, illetve ha nem adtunk meg célkönyvtárat, akkor az aktuális könyvtárba. Társítható kapcsolói: g=<jelszó>, path, wp, wo, wn.

x (= eXtract and remove)

A program az x parancs hatására a kibontás után törli az archív formátumú állományokat. (A PAK ezen a ponton eltér az általános szabálytól, mert az x és az e parancs nem teljesen azonosan működik.)

A fenti két parancs leggyakoribb kísérő kapcsolója a /path. Ez arról gondoskodik, hogy a /i kapcsolóval becsomagolt állományok az eredeti helyüknek megfelelő (al)könyvtárba kerüljenek. A jelszóval védett állományok természetesen csak a helyes jelszó megadásával (/g=<jelszó> kapcsolóval) bonthatók ki.

A fenti mellett ajánlott kapcsoló még a /d (= Duplicate). Ez a Freshen kifelé dolgozó változata. Amikor használjuk, kizárólag azokat a bepakolt fájlokat nyitja ki, amelyek a célkönyvtárban is megvannak.

p (= Print)

A bepakolt állományok közül a kijelölteket alapesetben a képernyőre írja ki. Ezáltal lehetővé teszi az archivált szövegállományok kipakolás nélküli megtekintését.

Ha /more kapcsolóval használjuk, akkor egy-egy képernyőnyi anyag kifrése után megáll a program és egy billentyű lenyomására vár.

A parancssor végén a DOS átirányító > operátorával a kimenet a képernyő helyett a nyomtató lehet (>prn, vagy >lpt1). A fentiekből következik, hogy a PAK nyomtatási lehetőségei messze elmaradnak az e téren vezető PKZIP-étől, sőt a PKPAK-étől is. Mindazonáltal nem okoz semmi gondot a tömörített állományokból a PAK programmal való nyomtatás sem.

A Print parancsot a /hex kapcsolóval alkalmazva bináris (program) állományok is megjeleníthetők a képernyőn, illetve nyomtathatók hexadecimális formában. Programozóknak megfizethetetlen szolgáltatás!

A jelszóval titkosított állományokat kinyomtatni is csak a jelszó megadásával lehet.

t (= Test)

A CRC értékek felhasználásával a PAK egyenként megvizsgálja a megadott fájlazonosítóknak megfelelő becsomagolt állományokat, hogy nem sérültek-e meg. Ezáltal nem kell egy hosszú archívot feleslegesen végigtesztelni, csak a keresett állományokat. Ha nem adunk a Test parancs használatakor fájlazonosító neveket, akkor a teljes archívot teszti.

A jelszóval védett fájl teszteléséhez meg kell adni a jelszót is a /g=<jelszó> kapcsolóval.

d (= Delete)

A Delete parancs használatakor az archív fájl nevét egyértelműen, * és ? dzsóker karakterek alkalmazása nélkül kell megadnunk, az archívon belül törlendő állományok megadásakor azonban már szabadon használhatjuk a dzsóker karaktereket. A parancs mellett alkalmazott /wp kapcsoló hatására a program minden egyes fájl törlése előtt rákérdez, törölheti-e.

Megjegyzések, jelszavas védelem

Megjegyzéseket a PAK nemcsak az egyes bepakolt fájlokhoz (fájl komment), hanem az archív állomány egészéhez is fűzhet (archív komment).

r (= Revise)

A meglévő archívban tárolt állományokhoz fűzött megjegyzések frissítésére szolgáló parancs.

h (= Header)

A Revise parancshoz hasonló feladatot lát el, de ez csak egyetlen megjegyzést módosít, amely az archív egészéhez tartozik.

A program lehetőséget ad a bepakolással egyidőben megjegyzések hozzáfűzésére is. Az alábbi kapcsolók a bepakoló és konvertáló parancsok mellett alkalmazhatók.

/r (= Remark)

Az egyes bepakolt állományokhoz kéri a megjegyzéseket.

/h (= Header remark)

Az archív kommentet kéri bepakolásakor.

A megjegyzések nem korlátozódnak egyetlen sorra. Ha egy bevezető @ karakter után megadjuk az előre elkészített megjegyzés szövegét tartalmazó fájl nevét, akkor a PAK onnan olvas be akár többsoros szöveget. Ez utóbbi esetben a megjegyzés tartalmazhat ANSI grafikát is.

Ha a Revise, vagy a Header parancsot alkalmazzuk, akkor a PAK kiírja először az egyes bepakolt állományokhoz előzőleg fűzött megjegyzés szövegét (ha volt ilyen), utána pedig az adott fájl nevét, végül bekéri az új megjegyzést. Ha ilyenkor csak az <ENTER>-t nyomjuk le, akkor a régi megjegyzés szövege marad változatlanul.

A konfigurációs szövegállományok segítségével többek között egy standard archív kommentet is definiálhatunk, amit a /h kapcsoló alkalmazásakor a PAK mindig beír az archívba.

Ide is tartozik, hogy a program remekül használható többek között arra is, hogy ZIP állományainkat terjesztés előtt saját logóval lássuk el. Nem kell hozzá más, mint a logó elkészítése normál ASCII állományban, pl. LOGO.DAT néven, amit a PAK könyvtárában helyezünk el. Utána a PAK gyorsan hozzáfűzi a fájl tartalmát a ZIP fájlhoz:

PAK h ZIP-fájl.ZIP (a ZIP kiterjesztést adjuk meg !!!) majd a komment bekérésekor egy bevezető @ jel után írjuk be a logót tartalmazó fájl nevét a teljes elérési úttal (pl. @c:\archiva\Nogo.dat).

/g=<jelszó> (= Garble)

A bepakoló parancsok mellett a /g=<jelszó> kapcsolót használva titkosíthatjuk a bepakolt állományokat. Az = jel a kapcsoló része! A program először nagybetűssé konvertálja a jelszót, így nem tesz különbséget a kis- és nagybetűs jelszóbeírás között. A titkosított állományok csak a pontos jelszó megadásával bonthatók ki, nézhetőek meg, nyomtathatók és tesztelhetőek.

Az archív katalógus megnézése

A PAK két katalóguslistázó parancsa teljesen egyenértékű, ezért együtt is tárgyalható.

l (= List)

v (= View)

Mind a List, mind a View parancs a DOS DIR parancsának megfelelően kilistázza az archív állomány tartalomjegyzékét. A fájlnevek, méretek és dátum adatok mellett közlik még az alkalmazott tömörítő eljárást, a tömörítési %-ot, az egyes állományok tömörített méretét, és a CRC-t. Ha a parancs mellett a /more kapcsolót is alkalmazzuk, akkor egy-egy képernyőnyi anyag kiírása után mindig megáll és bármelyik billentyű lenyomásával megy tovább.

Amennyiben az /r kapcsolót is alkalmazzuk, akkor a fájl kommentek is megjelennek a listán, a /h kapcsolóra pedig az az archív komment is. A PAK 12 féle tömörítési eljárást ismer és tart nyilván. Ezek a következők:

Kijelzés	Jelentése (A módszert alkalmazó programok)
--	Tömörítés nélkül (SEA ARC, PAK, PKARC).
Packed	Ismétlődő bájt értékek kódokkal helyettesítve (SEA ARC).
Squeezed	Huffman kódolás (SEA ARC 5.20-ig).
crunched	Lempel-Zev tömörítés (SEA ARC 4.5-ig).
Crunched	Lempel-Zev tömörítés (SEA ARC 5.0-tól).
Squashed	Lempel-Zev tömörítés (PKARC).
Crushed	Lempel-Zev tömörítés (PAK).
Distill	Huffman/Sliding Window (PAK 2.50).
Shrunk	Lempel-Zev tömörítés (PKZIP).
Reduced	Sliding Window (PKZIP 0.9).
imploded	Huffman/Sliding Window (PKZIP 1.0-tól).
Imploded	Huffman/Sliding Window (PKZIP 1.1-től).

Ha rendezetten akarjuk az archív katalógust kifratni, akkor az /s kapcsolók (= Sort) egyikét kell használnunk:

/sn	Név és kiterjesztés szerint rendez sorba.
/se	Kiterjesztés és név szerint rendez sorba.
/st	Az állományok dátum és idő bejegyzései szerint rendez sorba.
/ss	Nagyság szerint rendez.
/ne	Nem rendez, megtartja az eredeti sorrendet.

Konverziók

c (= Convert)

A PAK egyedülálló tulajdonsága a programba szervesen beépített formátum-konverzió. Már meglévő ARC, PAK és ZIP archív állományok egyetlen paranccsal közös formátumra hozhatók. A különböző kapcsolók alkalmazásával a célarchívba tetszőlegesen megválasztott tömörítő módszerrel csomagolhatjuk át a meglévő állományokat. Természetesen ugyanezek a lehetőségek egy új archív létrehozásakor is a rendelkezésünkre állnak. Itt kell megtárgyalnunk az SFX állományok készítését, illetve visszaalakítását normál archívokká.

Ha a Convert parancsot magában használjuk, kapcsolók nélkül, akkor a PAK a megadott archív névnek megfelelő archívokat PAK formátumúra konvertálja, de megtartja az eredeti kiterjesztést. Így, ha volt egy BONZO.ARC, egy BONZO.PAK és egy BONZO.ZIP állományunk az aktuális könyvtárban, akkor a „PAK c bonzo” parancssor egymás után mind a három archívot átalakítja PAK formátumúvá, de kiterjesztésüket változatlanul

hagyja. Amennyiben csak az egyiket akarjuk átkonvertálni, meg kell adni a kiterjesztést is.

A /ren kapcsoló (= Rename) alkalmazásakor a kiterjesztést is PAK-ra változtatja a program.

A /c (= Crunched) kapcsoló a SEA ARC, PKARC és a PAK 1.x számára olvasható módon tömörít.

A /s (= Squashed) kapcsoló a PKARC és a PAK 1.x programok módszereivel tömöríti az átkonvertált állományokat.

A /cr (= Crushed) kapcsoló PAK 1.x kompatibilis állományt eredményez.

Mivel az állományok formátuma teljesen eltérő, a PAK automatikusan átnevezi az átalakított archívot, ha ZIP és PAK formátumok között történt a konverzió.

A /z (= imploded) kapcsolóval a PKZIP 1.10 formátumra konvertál át a PAK.

A /bugs kapcsoló a PKZIP 1.01 formátumára alakítja át a forrásállományt. Ez a kapcsoló magába foglalja a /z kapcsolót is.

A /zs (= shrinking) kapcsolóval pedig a PKZIP 0.92-nek megfelelően csak shrinking technikával konvertál át a PAK.

A fenti kapcsolók a Convert parancson kívül a bepakoló parancsok mellett is használhatóak, s így közvetlenül a megfelelő formátumú archívot tudjuk a PAK-kal előállítani.

Az önkinyitő PAK-SFX archívok

Az SFX EXE állományok létrehozását a /exe kapcsolóval kérhetjük. A kapcsoló és a PAK program különlegessége, hogy nemcsak már kész archívokból lehet SFX állományokat csinálni a Convert parancs alkalmazásával, hanem a program képes az új archívot közvetlenül a „nyers” forrásállományokból létrehozni, bepakoló parancsokkal kombinálva (a, u, m). Ha a meglévő PAK állományt a Convert paranccsal és az /exe kapcsolóval alakítjuk át SFX állománnyá, akkor a PAK a forrás archívot a sikeres konvertálás után törli.

Ha meguntuk az SFX formát és vissza akarunk állni a „hagyományos” archív szerkezetre, csupán a Convert parancsot kell kiadnunk, azzal a kapcsolóval kombinálva, amelynek a formátumára szükségünk van.

Az előző részben ismertetett SFX előállítás eredményeként létrejött EXE kiterjesztésű önkinyitő archív ugyanúgy kezelhető, mint a többi archív állomány, csupán a kiterjesztését kell rendszeresen melléírni, hogy a PAK felismerje, mivel áll szemben.

A PAK-SFX állományoknak van néhány saját kapcsolója is, amivel érdemes megismerkednünk:

/w (= Write-over)

Az önkibontó PAK-SFX állományok kibontáskor mindig megkérdezik, hogy felülírhat-e, amennyiben van már a kibontandóval azonos néven egy fájl a célkönyvtárban. Ez a Write-over kapcsolóval elnyomható, és az SFX felülírja kérdés nélkül az azonos nevű állományokat.

/n (No pause)

A PAK-SFX megáll a kibontásban egy archív megjegyzés kiírását követően és megkérdezi, akarjuk-e folytatni. Ez letiltható az /n kapcsoló alkalmazásával.

/p (Path)

Ha az archívban tárolt könyvtár szerkezetet is vissza akarjuk állítani a célkönyvtárban, akkor az SFX Path kapcsolóját kell alkalmaznunk. Amennyiben valamelyik könyvtár nem létezne, az SFX létrehozza azt, és a frissen elkészített könyvtárba is kirakja az oda való állományokat.

A PAK-SFX indításakor nemcsak kapcsolókkal operálhatunk, hanem célkönyvtárat is megadhatunk. Ilyenkor a megadott könyvtárba kerülnek az SFX-ből kibontott állományok. Ha nem létezik a megadott célkönyvtár, akkor az SFX megpróbálja létrehozni.

A PAK parancsai ábécérendben

a (= Add) — Az archív bővítése a megadott állományokkal. Az azonos nevű, már bent lévő állományokat felülírja.

c (= Convert) — A meglévő különböző típusú (ARC, PAK, ZIP, SDN) archívok közötti átkonvertálást végzi, illetve a PAK-SFX fájlokról leszedi az SFX fejet. Kapcsolót lásd az adatkonverzióval foglalkozó résznél.

d (= Delete) — A feleslegessé vált állományokat eltávolítja az archívból.

e (= Extract) — A szabványos kibontást végzi. A /d (= Duplicate) kapcsolóval kombinálva kifelé frissít.

f (= Freshen) — Állományfrissítést végez, új fájlokat nem visz be, és a bent lévők közül is csak az kerül felülírásra, amelyik régebbi keltezésű, mint az archívon kívüli forrásállomány.

h (= Header) — Az archív kommentek karbantartására szolgál. Egy új archív létrehozásakor a /h kapcsolóval (Header remark) adhatunk archív megjegyzést az állomány egészéhez.

l (= List) — Az archív katalógus kilistázását végzi.

m (= Move) — A szabványos áthelyezést biztosítja a fájlok keltezésének vizsgálata nélkül. Ha fontos, hogy a frissebb verzió kerüljön be az archívba, használjuk inkább az Update, vagy Freshen parancsot a /m (= Move) kapcsolóval.

p (= Print) — A megadott archív kijelölt állományait a képernyőre bontja (ajánlatos a /more kapcsolóval használni). Átírányítható a megjelenítés a nyomtatóra is.

r (= Revise) — Az egyes bepakolt állományokhoz fűzött megjegyzések (fájl comments) karbantartására szolgál. Új állományok bepakolásakor megjegyzést az /r (Remark) kapcsolóval fűzhetünk az egyes fájlhoz.

t (= Test) — Az archív épségét ellenőrzi a CRC értékek segítségével.

u (= Update) — A Freshen parancshoz hasonlóan működik, de új állományokat is bevisz az archívba.

x (= eXtract and remove) — Kibontás után kitörli a kibontás előtti állományokat az archívból.

v (= View) — Az archív katalógus kilistázását végzi.

A PAK kapcsolói ábécérendben

/bugs (= make ZIP 1.01 compatible files) — A Convert parancs és a bepakoló parancsok mellett használatos ez a kapcsoló. Hatására a PAK program ZIP kiterjesztésű, a PKZIP 1.01-nek megfelelő módszerrel tömörített állományt hoz létre.

/c (= make ARC compatible files, Crunched) — A kapcsoló hatására a Convert parancs és a bepakoló parancsok kiadásakor a Crunched technikát alkalmazza az állományok tömörítésére, s így a SEA ARC programja számára emészthető formátumban ír.

/cr (= make PAK 1.x compatible files, Crushed) — Az 1.x verzióju PAK formátumban készíti el az archívot a PAK, amennyiben a Convert parancs, vagy a bepakoló parancsok mellett a /cr kapcsolót alkalmazzuk.

/d (= use only duplicate files) — Hatására a PAK csak azokkal az állományokkal foglalkozik, amelyek mind az archívon belül, mind pedig a forrás-, vagy célkönyvtárban egyaránt megvannak. Bár legtöbbször az a, e és x parancsokkal használjuk, a többi parancs mellett (pl. List) is alkalmazható. A Freshen parancs eleve tartalmazza ezt a funkciót.

/exe (= create SFX archives) — A Convert parancssal alkalmazva a már meglévő archívokból készít SFX állományokat, ha pedig a bepakoló parancsoknál alkalmazzuk, akkor közvetlenül a bepakolandó forrásállományokból készít PAK-SFX archívot.

/g=<jelszó> (= Garble) — Az állományok jelszavas titkosításáról gondoskodik. A jelszó nem tartalmazhat szóközt, és a kis- és nagybetűk egyenértékűek. Az egyenlőségjel része a kapcsolónak. A jelszót meg kell adni, valahányszor a titkosított állományt kibontani, megnézni, kinyomtatni, illetve ellenőrizni akarjuk.

/h (= use Header remark) — A bepakoló parancsok mellett alkalmazva archív kommentet adhatunk az archív állomány egészéhez.

/hex (= display using HEXadecimal format) — A print parancs mellett alkalmazva a megadott állományokat hexadecimális formában írja ki a képernyőre, illetve a nyomtatóra.

/i (= Include subdirectories) — A bepakoló parancsok (a, u, f, m) mellett alkalmazva arról gondoskodik, hogy a bepakolt állományok elérési útja is bekerüljön az archívba.

/l (= use Last date in archive to stamp archive) — Az archívot bármiképp módosító parancsok mellett alkalmazható az /l kapcsoló. Hatására az archív fájl keltezése a benne lévő legfrissebb bepakolt állomány keltezéséhez lesz igazítva.

/m (= Move files) — A normál bepakoló, illetve kipakoló parancsokat úgy hajtja végre a PAK az /m kapcsoló alkalmazásakor, hogy a be-, illetve kipakolt forrásállományokat törli. A Move és eXtract parancsok (m és x) mellett felesleges, azok magukban hordozzák már ezt a funkciót.

/more (= pause after every page) — Képernyőre bontáskor a képernyő beteltekor egy billentyű lenyomására vár.

/ns (= no order) — Az eredeti sorrendben, rendezés nélkül foglalkozik az állományokkal. Alkalmazható mind a bepakoló, mind a kipakoló, mind pedig az archív katalógust kiíró parancsok mellett.

/o (= use Only Distilling and Imploding) — A PAK általában az 1500 bájtól kisebb állományokra a „crushing” módszert alkalmazza, illetve ZIP formátumban az 500 bájtól kisebbekre a „shrinking” módszert. Az /o kapcsolóval arra utasíthatjuk a programot, hogy kisméretű állományok esetén is a „distilling”, illetve az „imploding” eljárást alkalmazza.

/p (= Pack archives) — A PAK az archív állományokat (PAK, ARC, ZIP) nem próbálja tovább tömöríteni, hacsak a /p parancssal erre nem utasítjuk. Ez legtöbbször sok időbe kerül és az archívok méretét alig tudja csökkenteni.

/path (= use PATH to compress, extract or display) — A PAK a meghajtónév kivételével a teljes Path információt megőrzi az archívban. Ezt azonban csak akkor használja fel működése során, ha a /path kapcsolót is alkalmazzuk.

/r (= use Remarks) — A bepakoló parancsok mellett alkalmazva az újonnan bevitt állományokhoz fájl-kommenteket kér be. A List, View és a kibontó parancsok mellett használva kiírja az állományokhoz fűzött megjegyzéseket.

/ren (= REName extension to PAK when converting) — A Convert parancs mellett alkalmazva az átkonvertált archív kiterjesztését PAK-ra írja át.

/s (= make PKARC compatible files, Squashed) — A Convert parancs, illetve a bepakoló parancsok mellett PKARC kompatibilis tömörítési módszert alkalmaz (squash).

/se (= Sort by Extension + filename order) — A kiterjesztés és a fájlnev szerint ábécé sorrendben foglalkozik az állományokkal a PAK a Convert, EXtract, List és bepakoló parancsok mellett.

/sec (= check SECcurity envelope if present) — Hatására a PAK minden parancsa bármi más művelet elvégzése előtt ellenőrzi a biztonsági csomagolást.

/sn (= Sort by fileName + extension order) — A fájlnev a kiterjesztés szerinti ábécé sorrendben foglalkozik az állományokkal a PAK a Convert, EXtract, List és bepakoló parancsok mellett.

/ss (= Sort by Size order) — Fájlméret szerinti rendezést biztosít a kapcsoló a különböző parancsok mellett.

/st (= Sort by Time order) — A fájl keletkezése szerinti rendezést biztosít a kapcsoló a különböző parancsok mellett.

/t (= use Temporary path) — A program működéséhez átmeneti munkakönyvtárat adhatunk meg a /t kapcsolóval. Különösen akkor előnyös, amikor floppyn dolgozunk, és/vagy a munkalemezen meglehetősen kevés a szabad lemezkapacitás. Ilyenkor a /t kapcsolóval tehermentesíthetjük az aktuális könyvtárat. A PAK a Print és a Convert parancs alkalmazásakor is átmeneti állományokat hoz létre, ha azokat nem tudja fejben tartani (64 K méret fölött). Az átmeneti munkakönyvtárt a PAK konfigurálása során is megadhatjuk. Ha a megadott átmeneti munkakönyvtár nem elérhető, akkor a PAK az archívot tartalmazó könyvtárba szemetel. Az alapértelmezés átállítását lásd a PAKINST program ismertetésénél.

/wa (= Write over Always existing files) — A kapcsoló hatására a PAK mindig felülírja, kérdés nélkül az azonos nevű állományokat.

/wn (= Write over no existing files) — A kapcsoló hatására a PAK sosem írja felül az azonos nevű állományokat.

/wo (= Write over Older files) — A kapcsoló hatására a PAK kérdés nélkül mindig felülírja az azonos nevű és régebbi keletkezésű állományokat.

/wp (= Wait for Permission, normal) — A kapcsoló hatására a PAK mindig engedélyt kér az adott művelet, legtöbbször felülírás, vagy törlés végrehajtására.

/z (= make ZIP 1.10 compatible files, Imploded) — A PKZIP 1.10 formátumában (az imploding módszert alkalmazva) tömörít. Hatására a PAK .ZIP kiterjesztésű állományokat hoz létre.

/zs (= make ZIP 0.91 compatible archives, Shrunk) — A PKZIP 0.91 formátumában (a shrunk módszert alkalmazva) tömörít. Hatására a PAK .ZIP kiterjesztésű állományokat hoz létre.

A PAKINST program

A PAK használatát barátságosabbá tehetjük, ha a PAKINST program segítségével komfortosítjuk a programot. Ehhez a következőket kell tennünk:

— Bemásoljuk a PAK és a PAKINST programot egy path-on lévő könyvtárba (C:\PAK), célszerűen a többi tömörítő mellé, majd átkapcsolunk erre a könyvtárra (cd \pak).

— A DOS promptról kiadjuk a PAKINST parancsot, majd a programmal beállítjuk a megfelelő adatokat.

— Gyakorlottsabb PAK alkalmazók kedvenc szövegszerkesztőjük segítségével átírhatják a PAK parancs- és kapcsolókészletét, s a PAKINST segítségével ezt beültethetik a PAK programba.

A PAKINST menüje

A Directory kijelölést csak a 2.xx DOS verziót használóknak kell igénybe venniük. A DOS 3.0 verzió felett ez szükségtelen. A PAK konfigurálása szövegállományból a P lenyomására indul, ekkor meg kell adni az előzőleg megszerkesztett, új parancsokat és kapcsolókat tartalmazó szövegállomány nevét. A T lenyomásával hozhatunk létre egy, a PAK jelenlegi parancs- és kapcsolókészletét tartalmazó szövegállományt. A Quit parancs befejezi a PAKINST ténykedését.

Konfigurációs szövegállomány a PAKINST-hez

A konfigurációs szövegállományokat kétféleképpen hozhatjuk létre:

— Megírjuk magunk a programmal együtt terjesztett két TXT fájl mintájára.

— A PAKINST programból a T billentyűt lenyomva hozzuk létre a PAK aktuális beállításából.

A konfigurációs állományt kedvenc szövegszerkesztőnkkel tetszésünk szerint átírhatjuk, feltéve ha az utána is megmarad normál ASCII szövegállománynak, és a szövegszerkesztő nem rak bele formázó és vezérlő karaktereket. (Ha mégis, akkor „tisza ASCII” formátumú szöveg előállítására képes másik kedvencet kell választanunk és megtanulnunk.)

A PAK konfigurációs állománya négy fő területre osztható:

1. Header (fejléc)

Ha megnézzük a mintaállományokat, vagy a PAKINT programmal készült szöveget, láthatjuk, hogy az elején egy szépen bekeretezett fejléc van. A sor elején és végén pedig { illetve } jelek fogják közre a szöveget.

A PAK számára a { és } jelek, illetve a (* és *) jelek közötti szöveg megjegyzésnek minősül, aminek semmi hatása nincs a program működésére, csak a fájl és az adott funkciók azonosítására, szemléletesebb leírására szolgál. A megjegyzés szövegek egy-egy soron belül bárhol lehetnek a nyitó ({, vagy (*)) és záró (*), vagy }) jelek között.

Hasonló megjegyzéseket nemcsak a fejlécben, hanem bárhol a szövegben elhelyezhetünk.

2. Parancsdefiníciós terület

A parancsokat az alábbi szerkezettel adjuk meg:

<parancs> = <PAK parancs> [+ <PAK kapcsolók>], „Help szöveg”

<parancs> — Azt a parancsot kell ide beírni, amit a PAK mellett akarunk használni. Betűvel kell kezdődnie, tartalmazhat betűket, aláhúzás jelet és számokat, nem lehet benne szóköz, aritmetikai jel, vagy írásjel. Hossza nem kötött, de senki sem szeret túl sokat írni feleslegesen.

<PAK parancs> — A PAK saját parancsai közül az, amit hozzá akarunk rendelni a fent ismertetett <parancs>-hoz. Az ide írható parancsok: ADD, EXTRACT, DELETE, LIST, VIEW, TEST, CONVERT, REVISE. (A PAK parancsnevek magukért beszélnek.)

<PAK kapcsolók> — Az általunk definiálandó parancsot természetesen a legkülönbélebb PAK kapcsolókkal kombinálhatók össze. Ezeket + jellel írhatjuk be (lásd a mintaállományokat). Egy adott PAK belső parancsához tetszőleges számban rendelhetünk PAK kapcsolókat, közte + jelekkel. A PAK belső kapcsolói a következők: crunch, squash, crush, implode, shrink, early_zip, single, move, duplicate, pack_archives, password, remark, archive_remark, self_extract, test_envelope, use_path, include_dirs, name_sort, time_sort, size_sort, ext_sort, no_sort, always_replace, prompt_replace, replace_older, never_replace, last_date, temp_path, hex, pause, rename, distill, duplicate, ascii. A PAK belső kapcsolóinak megadásakor a kis és nagybetűk egyenértékűek, a program nem tesz különbséget közöttük.

A parancs definíciós része után egy vesszőt kell tennünk, majd idézőjelek között beírhatjuk azt a szöveget, amit a PAK Help képernyőjén akarunk az adott parancs neve mellett megjeleníteni. Sajnos ékezetes betűket nem fogad el a PAKINST, azokat más módon kell (pl. utólag a PCTools program segítségével) bevinni a PAK programba.

Nem feltétlenül kell egy sornak a képernyő szélénél véget érnie, hosszabb is lehet, sőt egy logikai sor két sorban is elhelyezhető, ha a vessző utáni részt a következő sorba visszük át. A Help szöveg hosszát azonban tartsuk 35 karakter alatt.

3. Kapcsolódefiníciós terület

A saját használatra készített kapcsolók megadása rendkívül hasonlít a parancs definíciós szerkezetéhez:

`/<kapcsoló> = <PAK kapcsolók>, „Help szöveg”`

Az eltérés csupán annyi, hogy a sor elejére egy / jel kerül, és PAK parancsot nem rendelhetünk a kapcsolókhoz, csak a PAK kapcsolói közül az általunk választottakat, ugyanúgy, mint a fent említett parancsoknál. (Nem kötelező, de nem is hátrányos az összes PAK kapcsolót felhasználni a saját kapcsolók meghatározásakor.)

A kapcsoló definíciója után egy vesszőt kell tennünk, majd idézőjelek között beírhatjuk azt a szöveget, amit a PAK Help képernyőjén akarunk az adott kapcsoló neve mellett megjeleníteni.

4. Saját változók területe

A konfigurációs állomány utolsó területét nem kötelező felhasználni, de csak nyerhetünk, ha a három itt megadható változóhoz hozzárendelünk valamilyen értéket. Az értékadáshoz az idézőjelet is be kell írunk.

`$TPATH = "<meghajtó>:[\<Path>]"`

Az átmeneti munkakönyvtár adható meg a programnak ezen a módon. Ha a változónak megadjuk a ramdrive, vagy a gyors winchester azonosítóját, akkor a /t kapcsolót nem kell használnunk. (Pl. `$TPATH = "C:\TEMP"`.)

`$TDRIVES = "<meghajtó>: [<meghajtó>:[<meghajtó>: ...]"` illetve

`$TDRIVES = "<meghajtó>[,<meghajtó>,...]"`

Olyan esetekben, amikor el akarjuk kerülni, hogy a PAK bizonyos meghajtókra írjon, s legfeljebb az archív végleges alakja kerülhet a megadott meghajtókra (hálózati

meghajtók, WORM drive-ok, stb.), érdemes a változóba elhelyezni a védendő meghajtók neveit. Pl.:

```
$TDRIVES = "A: B: K: L:" illetve $TDRIVES = "A,B,K,L"
```

A sorrendben harmadiknak tárolt PAK változó tartalmazhatja magát az archív kommentet, vagy annak az állománynak a nevét egy bevezető @ jel után, amelyben előzőleg felépítettük azt a szöveget, amelyet az archív kommentjének szántunk. Pl.:

```
$STD_TITLE = "Archív komment" illetve
```

```
$STD_TITLE = "@<meghajtó>:\<Path>\<fejlec.lst>"
```

```
$STD_TITLE = "TallSoft Lop-Ware Stúdió & Library" illetve
```

```
$STD_TITLE = "@C:\DOKSNTS_FEJ.DOC"
```

Amennyiben a \$STD_TITLE változónak adtunk valami értéket, és a PAKINST segítségével beépítettük a PAK programba, valahányszor /h kapcsolóval hozunk létre új archívot, annak az archív kommentjébe a \$STD_TITLE tartalma, illetve a benne megadott állomány tartalma kerül. Nem kötelező a közvetítő fájl használata esetén annak teljes nevét megadni, de célszerű, hogy a PAK bárholon indítva megtalálja.

A PAK a gyakorlatban

PAK parancs [/kapcsolók] archív [maszkok] [@listafájl] vagy

PAK @listafájl

PAK

Természetesen a DOS promptról kell kezdeni a parancssor beírását. Célszerű a PAK.EXE-t egy path-on lévő könyvtárban elhelyezni, hogy a DOS mindig megtalálja. Ha nincs path-on, elé kell írni az elérési útvonalat, vagy abból a könyvtárból kell meghívni, ahol található.

Parancsok

A parancs a rendelkezésre álló 14 parancs egyike. A parancskészletet egy formázatlan ASCII szövegeket készítő szövegszerkesztővel (Norton Editor, Personal Editor, Word, Kedit, SideKick, stb.) és a PAK 2.50 részét képező PAKINST.EXE segédprogram segítségével könnyedén átírhatjuk, hogy jobban megfeleljen egyéni igényeinknek. Mintának lásd a csomagban megtalálható STDCFG.TXT és ALTCFG.TXT állományokat.

A PAK használatakor a következő szerkezetet használjuk:

PAK parancs [kapcsolók] archív [maszkok] [>kimenet]

A PAK.EXE programot célszerű egy path-on lévő könyvtárba tenni, ahol a DOS könnyen megtalálja és nem kell az elérési utakkal bíbelődni a program minden egyes meghívásánál.

Az előző részekben ismertett parancsok közül egyet be kell írunk a PAK mögé, attól egy szóközzel elválasztva. Ha nem írunk be a PAK által értelmezhető parancsot, akkor a Help jelenik meg.

Kapcsolók

A parancsok működését befolyásoló 30 kapcsoló (options) az alapértelmezés szerint a parancs után bárhol állhat a parancssorban. Egyszerre több kapcsolót is megadhatunk, ha azok hatása egymással nem ütközik. A többi tömörítőtől eltérően azonban nem adhatjuk meg egyetlen közös bevezető jel után, hanem egyenként szóközzel elkülönítve és /, vagy

— jelet közbeiktatva kell beírni. Azért kellett szétválasztani a kapcsolókat, hogy a program akkor se keverhesse őket össze ha átdefiniáltuk a parancs- és kapcsolókészletet. A kapcsolók használata nem kötelező, leírásukat lásd az előző oldalakon.

Archív

A létrehozandó, bővítendő, frissítendő, aktualizálandó, kibontandó, tesztelendő, megnézendő, vagy átkonvertálandó tömörített archív állomány neve. Alapértelmezésként, ha nem adunk meg kiterjesztést, a PAK kiterjesztésű állományokat keresi, de ha ilyen nincs, akkor az ARC vagy ZIP állományok kerülnek sorra. Az SFX állományokkal is igen jól elboldogul, ilyenkor azonban meg kell adnunk az EXE kiterjesztést is az archív nevében. Konvertáláskor is gyakran elhagyható a kiterjesztés, de biztonságosabb beírni.

Az archív nevét mindig meg kell adni. Az Add, Update, Freshen, Move és Delete parancsok használatakor az archív nevében nem szerepelhetnek dzsóker (* és ?) karakterek, az összes többi parancs használatakor igen. Az archív neve természetesen tartalmazhat elérési utat is.

Maszkok

Ha megadunk állományneveket (* és ? megengedett), akkor a PAK a megadott neveknek megfelelő állományokkal dolgozik. Ha nem adunk meg ilyeneket, akkor az alapértelmezés *.* , azaz minden. Amennyiben csak egy könyvtárnevet adunk meg (nem kell \ jel a könyvtárnév végére!), bepakoláskor abból mindent feldolgoz, illetve kipakoláskor az összes állományt lerakja oda.

Kipakoláskor a fájlnevek elé írt útvonalon éri el az adott fájlhoz tartozó célkönyvtárt. Egy állományból egyszerre több különböző célkönyvtárat is feltölthetünk, de egy adott fájl egy menetben nem kerülhet egyszerre két célkönyvtárba.

Listafájl

A DOS parancssorába csak korlátozott hosszúságú szöveget írhatunk be egyszerre. A nap mint nap ismételt parancsok azonos, vagy hasonló paraméterekkel való beírása sok hibalehetőséget rejt magában. Jó volna valamiképp dokumentálni, amit csináltunk az archív állománnyal. E három megoldandó problémára elegáns és közös megoldás a listafájl alkalmazása.

A listafájl tartalmazhat parancsokat, kapcsolókat és állományneveket, amiket különben a PAK után íránk be a parancssorba. A listafájl egy közönséges formázatlan ASCII szövegállomány, amit kedvünkre változtathatunk. A listafájl neve elé a parancssorban egy bevezető @ karaktert kell tenni, erről ismeri fel a PAK, hogy most listafájl az, amivel találkozott.

A kimenetet nem maga a PAK program, hanem a DOS használja. Segítségével átírányíthatjuk a különben csak a képernyőn megjelenő szövegeket egy megadott fájlba, vagy nyomtatóra. Hasznos ez a lehetőség, amikor az archív katalógust például ki akarjuk nyomtatni, vagy egy listafájlhoz akarjuk felhasználni, esetleg munkánkat és a PAK üzeneteit akarjuk dokumentálni.

Az SFX állományok ajánlott használata

1. Amennyiben még nincs, hozzuk létre a célkönyvtárat, ha egy külön könyvtárban akarjuk elhelyezni a kipakolt állományokat. (Pl. md c:\uj\kipakolt)

2. Lépünk a célkönyvtárba/meghajtóra (Pl. cd \uj\kipakolt)

3. Hívjuk meg az SFX-et, megadva annak elérési útvonalát is. (Pl. c:\uj\kipakolt>a:pak250)

Természetesen ettől el is térhetünk, de ez a legegyszerűbb és a legbiztonságosabb használata az SFX állományoknak.

Hibakódok

A PAK, amint egy jól nevelt tömörítő programhoz illik, munkája befejezésekor az errorlevel változó értékének beállításával is hagy üzenetet arról, hogy történt-e, és miféle hiba munkája közben. Az alábbi kis táblázat összefoglalja a hibakódokat és jelentésüket. Ezek ismeretében a PAK batch alkalmazását is elkezdhetjük, mivel csak az errorlevel értékét kell figyeltetnünk, ami nem túl nagy probléma.

Errorlevel Jelentése

- 0 Semmi hibát nem észlelt a PAK.
- 1 CRC hiba, vagy hiányzott a megadott fájl.
- 2 Nem volt elég a szabad RAM terület.
- 7 Lemezhiba.

Gyakorlati feladatok, megoldások

1. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, de nem akarjuk a könyvtár tartalmát elveszíteni.

C:\MUNKA>PAK a a:munka c:\munka\konyv*.*

2. A C:\MUNKA könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az archív elkészültével ki akarjuk írítani a könyvtárat.

C:\MUNKA>PAK m a:munka c:\munka\konyv*.*

3. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az egész C:\MUNKA\KONYV alkönyvtárt a pakolás után ki akarjuk törölni.

C:\MUNKA>PAK m a:munka c:\munka\konyv*.*

4. Csak a napi munka során megváltozott állományokat akarjuk az A:-ban lévő lemezen tárolt MUNKA.ARC fájlban felfrissíteni.

C:\MUNKA>PAK f a:munka c:\munka\konyv*.*

5. A C:\MUNKA és a C:\MUNKA\KONYV könyvtárakból minden DOC kiterjesztésű állománnyal aktualizáljuk az A:MUNKA.ARC állományt. A forrásállományokat nem akarjuk törölni.

C:\MUNKA>PAK u a:munka c:\munka\konyv*.doc c:\munka*.doc

6. A C:\MUNKA\KONYV könyvtárból minden nem COM kiterjesztésű állománnyal bővítjük az A:MUNKA.ARC állományt. A forrásállományokat nem akarjuk törölni.

C:\MUNKA\KONYV>no *.com PAK a a:munka *.*

7. A MUNKA.ARC-ba tévedésből bekerült BAK, ARC és ~DO kiterjesztésű állományokat kell eltávolítanunk.

C:\MUNKA>PAK d a:munka *.bak *.~do *.arc

8. Meg akarjuk nézni, mi van a MUNKA.ARC állományban.

C:\MUNKA>PAK l a:munka — csak tömör listát ad

C:\MUNKA>PAK v a:munka /more — teljes listát ad

9. Ki akarjuk listáztatni a MUNKA.ARC tartalmát egy szabadon szerkeszthető ASCII állományba, de csak a nevekre és a kiterjesztésekre vagyunk kíváncsiak. (Pl. egy listafájl készítését tervezgetjük.)

C:\MUNKA>PAK l a:munka >lista.lst

10. Gyanús a lemez. Ellenőrizni akarjuk a MUNKA.ARC épségét.

C:\MUNKA>PAK t a:munka

11. Meg akarjuk nézni az egyik (pl. az ARC.DOC) állományt.

C:\MUNKA>PAK p a:munka arc.doc /more

12. Megtetszett a fenti ARC.DOC. Nyomtassuk ki.

C:\MUNKA>PAK p a:munka arc.doc >prn

13. A régi verzióju ARC programmal pakolt KAKUKK.ARC nevű állományt tömörebbé akarjuk tenni, de azért a régi KAKUKK.ARC-ot is meg akarjuk őrizni.

C:\MUNKA>copy kakukk.arc kakukk_2.ARC

C:\MUNKA>PAK c a:kakukk_2

(A régi formátumú KAKUKK.ARC megmarad)

14. A MUNKA.ARC-ból a TXT kiterjesztésű állományok a C:\TXT könyvtárba, a DOC kiterjesztésűek a C:\DOKSI könyvtárba pakolandók ki.

C:\MUNKA>PAK e a:munka c:\txt*.txt c:\doksi*.doc

15. Maszek állományainkat titkosítva akarjuk a MUNKA.ARC-ba áthelyezni, hogy a C:-n nyoma se maradjon.

C:\MUNKA>PAK m /g=na_találd_ki_mi_a_jelszó a:munka maszek *.*

Új trónkövetelő: az ARJ

Robert K. Jung ARJ nevű programjának váratlan felbukkanása egyik napról a másikra átrendezte a tömörítők élmezőnyét. Az Alaplap 1991. júliusi számában a tömörítőkről szóló részletes összeállítás tesztjének elkészítésekor a programról még csak hírből hallottunk, két hónappal később, az ARJ bevonásával megismételt összehasonlító próbánkban pedig az ARJ azonnal az élre került.

A program, amelynek 2.10-es verziója 1991. május 28-i keltezésű, az LHA és a PAK programhoz hasonlóan egyetlen állománnyal oldja meg a bepakolással, kipakolással és az SFX készítésével kapcsolatos feladatokat. Kísérő leírásokat is tartalmaz bőségesen, ezek segítségével, valamint a help képernyők és saját tapasztalataim alapján ismertetem a programot.

Mitől vált érdekessé ez a program?

- A szabványos parancsokat természetesen mind beleépítették.
- Az alkalmazott tömörítési módszer kapcsolóval választható.
- Több lemezzel is dolgozhat.
- A műveletekből kizárhatók kiválasztott állományok.
- A program ellenőrizheti saját épségét is. (!)
- A bepakolt állományok átnevezhetők.
- A bepakolt állományok átrendezhetők.
- Szövegrész megkereshető vele az archívban is.
- Rejtett és rendszerállományokat is kezelhet.
- Backup funkciója van (+ archív bitet is hozzákapcsolhat).
- Megnézheti a szabad helyet a céllemezen.
- Kommenteket listafájlokból is bepakolhat.
- Jelszavas védelem alkalmazható.
- Időkorlátok adhatók meg számára.
- A Huffman féle pufferméret megadható (2048-65535).
- Az ARJ az SFX-et is maga állítja elő, a PAK-hoz hasonlóan, akár közvetlenül, egyetlen lépésben a bepakoláskor.
- Az LHA 2.12-nél tömörebb archívokat készít.
- Egzotikus ARJ kiterjesztést alkalmaz, a vírusok jó része nem támadja meg az ARJ archívokat.
- Paraméterek nélkül egy átlagos tömörítő helpet ad, a -? kapcsolóval pedig egy négy képernyős részletes helpet.
- Aki már megszokta az LHA-t, az örülhet az ARJ-nek is, mert a két programban majdnem teljesen azonos a parancssor szerkezete.
- Az ARJ-SFX saját parancsokkal és kapcsolókkal is rendelkezik, melyek megegyeznek az ARJ kipakoláskor használható paramétereivel.
- ARJ-SFX-JR is készíthető.
- A PKZIP-hez hasonlóan 32 bites CRC-t használ.

— A sérült archívból segédprogram nélkül is ki tudja ollózni a még épen maradt állományokat.

— A csomag részletes leírást és sok hasznos segédprogramot tartalmaz.

Negatívumai

— Lassú, bár megfelelően megválasztott kapcsolók alkalmazásával gyorsítható.

— Az ARJ-SFX mintegy 14 K-val növeli meg az archív méretét.

— A „buta” ARJ-SFXJR csak kibont. Megadható paramétere nincs.

Mit tartalmaz a program?

Az ARJ 2.10 egy csaknem 200 K méretű ARJ210.EXE nevű SFX állományban terjed. A csomag az ARJ program és leírásai mellett több igen hasznos és érdekes segédprogramot is tartalmaz. Kibontásához mintegy 340 K szabad lemezterület kell. Kibontás után az alábbi állományokat találjuk az ARJ könyvtárban:

ARJ210.EXE	A csomagot tartalmazó SFX állomány.
UNARJ.ARJ	Az UNARJ.EXE program forráskóddal, dokumentációval.
ARJ.DOC	Az ARJ felhasználói kézikönyve (angolul).
ARJ.EXE	Maga az ARJ program.
ARJ_BBS.DOC	Az ARJ programot támogató BBS-ek.
ARJSORT.BAT	Az archív rendező segédprogram behívója.
ARJSORT.COM	A rendező program.
ARJSORT.DOC	A rendező program leírása (angolul).
LICENSE.DOC	Jogi tudnivalók.
ORDERFRM.DOC	Megrendelő formanyomtatvány az ARJ-hoz.
README.DOC	Eligazító.
REARJ.CFG	A REARJ program konfigurációs állománya.
REARJ.DOC	A REARJ leírása (angolul).
REARJ.EXE	Archív átalakító segédprogram.
REGISTER.EXE	Az ARJ regisztrálásához szükséges program.
TECHNOTE.DOC	Technikai adatok az ARJ-ről. ARJ szerkezet.
UPDATE.DOC	Ami a kézikönyvből kimaradt.
WHATSNEW.DOC	Változások az 1.00-tól kezdve.
WHY_ARJ.DOC	Összehasonlító adatok.

A program szintaxisa

ARJ <comm> [/<sw>[<options>]] <archív> [<base>\] [fájllista]

vagy

ARJ <comm> [-<sw>[<options>]] <archív> [<base>\] [fájllista]

A parancsok: <comm>

Egybetűsek, egyszerre egy adható közülük, nincs előttük bevezető karakter. Rögzített helyük a parancssorban közvetlenül az ARJ után, attól egy szóközzel elválasztva.

Kapcsolók: <sw>

Bevezető /, vagy – karakter után állhatnak a parancs után a parancssorban bárhol. Hatásukat kiegészítő módosítók lehetnek (részletesen lásd a konkrét kapcsolók ismertetésénél). A kapcsoló után írt + jel bekapcsolja, a – jel pedig kikapcsolja az adott kapcsolót. Az üres kapcsoló csak átkapcsol. A kapcsolók többször is szerepelhetnek egy parancssorban.

Ha le akarjuk tiltani a parancssorban a hátralévő kapcsolókat, akkor a /– kapcsolót kell beírunk a megfelelő helyre. Bár kétféle kapcsoló bevezető jel is használható, egy parancssoron belül csak egyfajta alkalmazunk! Ez vonatkozik az ARJ_SW változóra is.

Módosítók: <options>

A kapcsolók hatását módosítják. A – és + az adott kapcsoló ki- és bekapcsolására szolgál, a többi lehetséges módosítót lásd részletesen a konkrét kapcsolók ismertetésénél).

Archív

Az archív neve tartalmazhat path-információt is. Az archív kiterjesztése elhagyható, alapértelmezése ARJ. Több részes archívok létrehozásakor az első rész .ARJ, a többi sorra .A01, ..., .A99 kiterjesztést kap.

Báziskönyvtár: <base>

A forrás, vagy célkönyvtár, attól függően, hogy bepakolunk, vagy kipakolunk. Ha elhagyjuk, akkor az éppen aktuális könyvtár lesz a báziskönyvtár. A megadott báziskönyvtár nevének a végén \, vagy : karakter legyen.

Fájllista: [!<listafájl>], [<teljes név>], [<fájlmaszk>]

Megadhatók konkrét állományok egyértelmű nevekkel, teljes elérési úttal (<teljes név>), vagy dzsóker karakterekkel (<fájlmaszk>), valamint listafájl is egy bevezető ! jel után (!<listafájl>).

Listafájl

Egy bevezető ! karakter után adhatjuk meg a listafájl nevet az ARJ számára. Amennyiben a listafájl segítségével nem kommenteket, hanem fájlspecifikációkat akarunk megadni, akkor a listafájl szerkezetére érvényesek az alábbi megkötések:

- Soronként egy fájlnevet adhatunk meg.
- Nem szabad üres sorokat beszúrni.
- Megjegyzéseknek nincs helye a listafájlban.
- Ne tegyünk sem a nevek elé, sem a nevek után szóközöket.
- A neveket akár teljes path-szal is megadhatjuk a DOS szabályai szerint.

Ha ! jellel kezdődő fájlnevek is szerepelnek a kezelendő állományok között, akkor a !/ kapcsolóval megadhatunk egy új listafájl-azonosító karaktert (pl. !/@ @listfile.lst).

A program megfelelően igazodik a tömörítő programok íratlan szabványaihoz.

Az ARJ szabványos parancsai

a	= Add	= Sima bővítés.
d	= Delete	= File törlése az archívból.
e	= Extract	= Egyszerű kibontás.
x	= eXtract with path	= Kibontás path-ra.
f	= Freshen	= Frissítés.

l	= List	= Archív katalógus kiírása.
u	= Update	= Az archív aktualizálása.
v	= Verbose list	= Részletes archív katalógus kiírása.
m	= Move	= Fájlok áthelyezése az archívba.
t	= Test	= Az ARJ-fájl épségének ellenőrzése.
c	= Comment	= Megjegyzések hozzáfűzése.
p	= Print fájls	= Kiírás a standard kimenetre.

A nem szabványos parancsok

s = print to Screen with pause =	Képernyőre írás.
w = Where are text strings in archive =	Szövegkeresés az archívban.
i = check Integrity of ARJ =	Az ARJ.EXE ellenőrzése.
j = Join archives =	Archívok összedolgozása.
n = reName fájls in archive =	Bepakolt fájl átnevezése.
o = Order fájls in archive =	Az archív átrendezése.
r = Remove path =	Path info eltávolítása.

Az ARJ bepakoló parancsai

a (= Add)

Egyszerűen bepakolja a kijelölt állományokat az archívba. Új archív létrehozására is alkalmas. Ha maximális tömörítést akarunk elérni, akkor a /jm és /m1 kapcsolókkal együtt használjuk. A /r kapcsolóval a báziskönyvtár alkönyvtáraiban is keres bepakolandó állományt. Amennyiben az archívot a forráskönyvtárban hozzuk létre és (főleg a /je, illetve a /je1 kapcsolók mellett) már létezik egy állomány az archívval azonos néven, akkor az ARJ nem törli az eredeti fájlt, és a frissen elkészült archívot nem nevezi át a megadott névre, hanem megmarad ARJTEMP.\$00 néven. Ezt később átmásolhatjuk, vagy átnevezhetjük.

u (= Update)

Az a parancshoz hasonlóan bepakol, de csak azokat az állományokat viszi be az archívba, amelyek még nincsenek ott bent, vagy frissebbek azoknál. Új archívot is létre tud hozni.

f (= Freshen)

Az a parancshoz hasonlóan bepakol, de csak azokat az állományokat viszi be az archívba, amelyek régebbi verziója már bent van. Nem tud új állományt bevinni, és új archívot létrehozni.

m (= Move)

Az a parancshoz hasonlóan bepakolja az archívba a kijelölt állományokat, majd a sikeres bepakolás után törli a forrásfájlokat. A törléshez nem kér engedélyt. (Ha azt akarjuk, hogy rékérdezzen, akkor a /q kapcsolót, vagy az a parancsot a /d kapcsolóval együtt kell alkalmaznunk.)

A fenti parancsokkal leggyakrabban kombinált kapcsolók: /g<jelszó>, /w<könyvtár>, /x<* .kit>, /r, /v, /ja, /je, /jf.

A bepakoló parancsok használatakor az archív nevét nem adhatjuk meg dzsóker karakterekkel. Ilyenkor egyértelmű nevet kell megadni. A báziskönyvtár alapértelmezése a bepakoló parancsok esetén is az aktuális könyvtár. A bepakolandó állományok

felsorolásában már alkalmazhatunk a DOS szabályainak megfelelően dzsóker karaktereket, valamint listafájlt is, amit az ARJ esetén közvetlenül a fájlnev elé tett ! jelez a programnak.

Kipakoló parancsok

e (= Extract)

A szabványnak megfelelő kibontó parancs. A megadott állományokat a báziskönyvtárban, illetve annak hiányában az aktuális könyvtárban bontja ki. Több részből álló archívoknál a /v kapcsoló is kell a rendes kibontáshoz.

x (= eXtract)

A bepakolt állományokat a forráskönyvtár szerkezetének megfelelően bontja ki, ha a path információt is rögzítettük bepakolásakor a /p, vagy /jf kapcsolókkal. Több részből álló archívoknál a /v kapcsoló is kell a rendes kibontáshoz. Ha az archív létrehozásakor a /jf kapcsolót is használtuk, akkor az abszolút könyvtárszerkezetet építi újra, különben a célkönyvtárból indulva relatív szerkezetet épít újjá.

p (= Print)

A standard kimenetre nyitja ki a megadott archív megadott állományait. Ez alapesetben a képernyő. Az így előhívott szöveg meglehetősen gyorsan gördül át a monitoron, nehéz elolvasni. Beleolvasás céljára jobb az alább ismertetendő s parancs. A DOS > átirányító operátorával a nyomtatóra is küldhető. Ha ki akarunk lépni a parancsból, akkor <CTRL BREAK> kell neki.

Fontos tudni, hogy a p paranccsal létrehozott kimenő állományba nem kerül extra header adat, azaz a bepakolt fájl kerül oda tisztán, kibontva. Az extra adatok mindenképpen a képernyőre kerülnek.

s (= Screen)

A p parancshoz hasonlóan a képernyőre nyitja ki a megadott archív megadott állományait, de minden egyes megtelt képernyő alján megkérdezi, hogy elég volt-e már a bepillantás. A válasz alapértelmezése yes, és ha elfogadjuk az <ENTER>-rel, akkor a következő fájl kiírásába kezd. Ha még a következő oldalt is látni akarjuk, akkor n, majd <ENTER> a válasz. Ha ki akarunk lépni a parancsból, akkor <CTRL BREAK> kell neki.

Az s parancs úgy tördeli kiírásakor a szöveget, hogy soronként maximum 79 karaktert ír ki, valamint a vezérlő kódok helyére ? jelet tesz.

Mind az s, mind a p parancsra érvényes, hogy az ékezetes karaktereket nem tudja megjeleníteni, ezért az ilyen szövegek megnézéséhez vegyük igénybe inkább a SHEZ 6.1 és a LIST programok segítségét.

t (= Test)

A parancs hatására a program először kiírja a megadott archív kommentjét, majd a CRC értékek segítségével a máshol is megszokott módon épségellenőrzést végez. Hasonló funkciót lát el a /jt kapcsoló, amely a bepakoló parancsok mellett használható, s aminek a hatására az ARJ program a frissített archív véglegesítése előtt ellenőrzi az átmeneti állományként jelen lévő új archívot.

A kibontó parancsok mindegyikénél használhatók dzsóker karakterek az archív megadásakor is. A tényleges kibontást végző e és x parancsok az éppen aktuális könyvtárban bontják ki az archív tartalmát, ha nem adunk meg nekik célkönyvtárat.

Megjegyzések, jelszók

c (= Comment)

Az ARJ kommenteket csak már meglévő archívokhoz tud hozzáfűzni, újonnan létrehozott ARJ állományokhoz nem. Kétféleképpen adhatunk meg kommenteket:

— Közvetlenül, a c parancs kapcsolók nélküli alkalmazásával mind az archívhoz, mind az egyes bepakolt állományokhoz hozzáfűzhetünk megjegyzéseket. Egy megjegyzésre maximum 2048 bájt juthat. Ez megfelel 25 darab 80 karakteres sornak. A sor végét <ENTER>-rel jelezzük, az üres sor pedig a megjegyzés végét jelenti az ARJ-nek, amire a soron következő fájlhoz tartozó megjegyzést kéri be a program. Az ARJ először az archív kommentet kéri be, majd sorban a fájl kommenteket.

— A/z kapcsolóval is megadható a kommentet tartalmazó listafájl, ha a kommentszöveg legelső karakterétől egy bevezető ! karakterrel indítva megadjuk a listafájl teljes nevét útvonallal, kiterjesztéssel.

— Ha a kommentet tartalmazó listafájl hosszabb, mint 2048 bájt, akkor csak a fájl első 2048 karaktere kerül be kommentként az archívba, a többi elvész.

— A /z[<arckommfájl.kit>] és a /jz[<fájlkommfájl.kit>] kapcsolók csak a c parancs mellett használhatók, illetve a /z kapcsoló az e és az x mellett is (lásd a részletesen a kapcsolóknál). Hatásukra a megadott állomány(ok)ból veszi át a megjegyzés szöveget az ARJ.

— Ha a „c /z”, illetve a „c /z<kommentfájl>” kapcsoló mellett nem használjuk a /jz kapcsolót, akkor csak archív kommentet viszünk be. A /z és /jz nélkül a c parancs mind az archív, mind a fájlkommenteket bekéri.

A bevitt megjegyzéseket a v parancs segítségével olvashatjuk ki.

Az ARJ a jelszavas védelmet a /g<jelszó> kapcsolóval biztosítja. A kapcsoló az összes paranccsal együtt alkalmazható. A jelszóban nem lehet szóköz. A jelszóval titkosított állományokat csak a jelszó megadásával lehet kibontani, megnézni, nyomtatni, tesztelni.

Archív karbantartó parancsok

d (= Delete)

A szabványnak megfelelően törli a megadott állományt az archívból. Amennyiben a /k kapcsolóval használjuk, akkor a régi ARJ is megmarad .BAK kiterjesztéssel.

A parancs nem kérdez rá a törlés előtt, hogy törölhetünk-e. Ha az /x kapcsolóval nem sikerült rendesen kizárni a folyamatból a kizárandó állományokat, akkor használjuk a /q kapcsolót, így egyenként válogathatunk, hogy az egyes állományokat akarjuk-e törölni vagy sem.

l (= List)

A parancs hatására az ARJ kilistázza az archív katalógust. Ha van archív komment, az is megjelenik a lista elején. Az alábbi információk kaphatók meg az l paranccsal:

— Az archív neve, valamint dátum és időpont adatai. Ha a dátumban 80-nál kisebb értéket adunk, azt a program XXI. századi dátumnak veszi. Ezt itt a dátum előtti + jel mutatja.

— Az archív komment.

— A bepakolt állományok neve.

— Az eredeti és a tömörített méret, valamint a tömörítési arányszám.

— A bepakolt állományok dátum és idő adatai.

— 32 bites CRC.

— A bepakolt fájlok eredeti attribútumai (a, h, s, w).

— TPMGVX.

Az utolsó oszlop (TPMGVX) jelentése:

T = Text/binary/directory/volume (TBDV).

P = Van-e path információ a névben (P).

M = A használt tömörítési módszer (0-4).

G = Titkosított-e a fájl (G).

V = Többrészes archív, a fájl folytatása következik (V).

X = Többrészes archív, a fájl egy nagy fájl folytatása (X).

A parancs használatakor az archív nevét dzsóker karakterekkel is megadhatjuk. Ekkor az összes megfelelő archívot kilistázza a program.

v (= Verbose list)

A fenti l parancshoz hasonlóan működik, de jóval több információt jeleníthet meg. Plusz szolgáltatásai:

— Kiírja az archív path-t is.

— Kiírja a fájlkommenteket is.

— Kiírja, milyen operációs rendszer alatt készült az archív.

— A /jv kapcsolóval egy listafájl szerkesztéséhez felhasználható kilistázást ad, ami átirányítható a DOS >, vagy >> átirányító parancsával egy ASCII fájlba.

— A /jp kapcsolóval a képernyő megtelte után vár egy billentyű lenyomásra.

A parancs használatakor az archív nevét dzsóker karakterekkel is megadhatjuk. Ekkor a program az összes megfelelő archívot kilistázza.

n (= reName)

Az archívban lévő állományok átnevezésére szolgál. A parancs kiadása után az ARJ egyenként felkínálja a bepakolt állományok nevét az átnevezéshez. Az <ENTER> billentyűvel elfogadhatjuk a régi nevet vagy adhatunk újat is, akár path-szal együtt. Ha több nevet akarunk megváltoztatni, akkor az archív neve után az átkeresztelendő fájlok nevét is meg kell adnunk a parancssorban, vagy pedig az <ENTER>-t nyomogathatjuk szorgalmasan. Ha meggondoltuk közben magunkat, akkor <CTRL BREAK>-kel megszakítható a program futása.

o (= Order)

A parancs használatával rendezhetjük az archívot. De ne várjon senki sokat ettől a lehetőségtől. Mindössze azt biztosítja, hogy a megadott fájlok az archív elejére kerülnek, a parancssorban megadott sorrendben. A parancs használatakor sem az archív, sem a fájlok megadásakor nem alkalmazhatunk dzsóker karaktereket. Többrészes archívokat nem lehet rendezni.

Listafájl alapján is rendezhetünk, ekkor a listafájl előtt egy bevezető ! karaktert kell írunk. Listafájl könnyen készíthető rendezéshez. Csak az ARJ v /jv <archív> >ARJ-SORT.\$\$1 parancsot kell kiadnunk. Az így előállított fájl egy jobb szövegszerkesztővel, vagy a DOS SORT parancsát használva átrendezhető. A rendezett listaállományt az ARJ ARJSORT.\$\$\$ néven várja, azzal remekül tud dolgozni tovább: ARJ o <archív> !ARJSORT.\$\$\$ Ennek automatizálására született az ARJSORT.BAT és az ARJ-SORT.COM program. Batch-fájlokban az utóbbi gyorsabb és jobb.

r (= Remove path info)

Az r parancs eltávolítja a megadott archív megadott állományainak nevééről a path bejegyzést. Dzsóker karakterek alkalmazhatók.

Önkicsomagolók (SFX) létrehozása és használata

Az ARJ nemcsak kész archívból, hanem a PAK-hoz hasonlóan becsomagolatlan forrásállományokból közvetlenül is tud ARJ-SFX-et készíteni. Az SFX készítést a /je és a /jel kapcsolók biztosítják.

Célszerű egy olyan archív kommentet tenni az SFX elé, amiben e jövődő felhasználóval közöljük, mi van a csomagban, mekkora helyet foglal el kibontva, és azt hogyan érdemes kibontani. Az sem árt, ha az SFX nevében jelezzük, például egy # karakterrel a név végén, hogy ez SFX állomány.

Az ARJ-SFX (/je kapcsoló) nem támogatja az /m4 kapcsolóval megadott tömörítési technikát.

A /jel kapcsolóval készített SFX mindössze 6 K-val növeli meg az archív méretét, de csak kinyitni tud és nem lehet paraméterezni, mert ez egy „buta” junior változat. A /jel nem támogatja az /m4, a /t1 és a /g kapcsolók használatát sem.

A /je kapcsolóval készített SFX 14 K-val növeli meg az archív méretét, ami nem kevés, de az így készített SFX több parancsot, kapcsolót és egyéb paramétert is használhat:

ARJ-SFX [-<command>] [-<sw>] [<célkönyvtár>\] [maszk]

Az ARJ-től eltérően az ARJ-vel készített SFX állományok parancsai előtt ki kell egy bevezető – karaktert tenni!

Az ARJ-SFX parancsal: <command>

e (= Extract)

Egyszerűen kibontja az SFX megadott állományait a megadott célkönyvtárba. Ha nem adunk meg célkönyvtárat, akkor az aktuális könyvtárba.

x (= eXtract)

Az e parancshoz hasonlóan az x parancs is kibontja az SFX-et, de ezen felül a tárolt könyvtárszerkezetet is visszaállítja és a path-szal elmentett állományok mind a helyükre kerülnek, ha a /p kapcsolót is használjuk. A 2.10 verziótól ez a parancs az alapértelmezés, amennyiben az SFX-et paraméterek nélkül hívjuk meg.

t (= Test)

A t parancs az ARJ t parancsához hasonlóan a tárolt CRC értékek segítségével teszteli az SFX épségét.

l (= List)

Az ARJ l parancsával megegyezően (részletesebben lásd ott) kilistázza az SFX archív katalógusát.

v (= Verbose list)

Az ARJ v parancsával megegyezően (részletesebben lásd ott) kilistázza az SFX archív katalógusát.

? (= Help)

Ha nem adunk parancsot az ARJ-SFX indításakor, akkor az SFX kiírja az archív kommentet, az SFX teljes nevét és kelezését, közli, hogy a -? parancsra Help képernyőt adhat, majd visszakérdez, hogy folytassa-e a kibontást. Yes, No, vagy Quit lehet a válasz, ezt közli is, ha csupán az <ENTER>-t nyomjuk le. A Help képernyő felsorolja az ARJ-SFX parancsait, kapcsolóit és szintaktikáját.

Az ARJ-SFX kapcsolói: <sw>

/a

Az ANSI kommenteket is megmutatja,

/c

A célkönyvtárban lévő azonos nevű állományokat szó nélkül felülírja.

/f

Nem nyit ki új állományokat, csak a már kint is meglévőket frissíti, ha az archívban lévő frissebb kelezésű.

/g<jelszó>

A jelszóval titkosított állományok kibontásához és teszteléséhez is meg kell adni a jelszót a /g<jelszó> kapcsolóval. Ha nem használjuk, akkor a titkosított állományokat az SFX nem bontja ki, nem ellenőrzi.

/n

Csak olyan állományokat bont ki az /n kapcsolóval, amilyen a célkönyvtárban még nincs.

/p

A helyükre kerülnek a path-szal elmentett állományok.

/u

Csak olyan állományokat bont ki az /u kapcsolóval, amilyen a célkönyvtárban még nincs, vagy régebbi kelezésű, mint az SFX-ben lévő.

/y

Minden igen/nem kérdésre igent válaszol.

Az ARJ kapcsolóiról részletesen

A kapcsolók helye a parancsokkal ellentétben nem rögzített, a parancs után bárhol állhatnak. Egy parancssorban több kapcsolót is használhatunk, de külön-külön bevezető /, vagy – karakterek után.

Az ARJ szép számú kapcsolói különböző módosítókat kaphatnak, amelyek jelentősen befolyásolják a program működését. Részletesen lásd a konkrét kapcsolóknál.

/-

Nem engedélyez további kapcsolókat a parancssorban. Olyankor használjuk, ha a ezt a karaktert egyéb célokra kívánjuk csak használni, vagy az egyik bepakolandó fájl neve – jellel kezdődik.

/?

A listafájl bevezető karakterét adhatjuk meg vele. Pl. átállhatunk a más programoknál megszokott @ karakterre. Ha magában áll, akkor lehetővé teszi ! karakterrel kezdődő állományoknak, mint LHA önindító állományának a bepakolását.

/\$

A PKZIP-hez hasonlóan az ARJ is megőrizheti, illetve visszaírhatja a lemezek nevét. Ha nem adunk meghajtó azonosítót, akkor az aktuális lemezzel foglalkozik. A meghajtó név után nem kötelező a kettőspontot kitennünk. A /je parancsok nem támogatják. A lemeznevet nagybetűsként őrzi. Ha a lemeznév grafikus karaktereket illetve kisbetűket is tartalmaz, akkor a program megzavarodik. Az ilyen lemezeket tömörítés előtt nevezzük át a DOS LABEL parancsával.

/a

A kapcsoló a bepakoló parancsok mellett használható. Az /a kapcsoló használata engedélyezi a DOS által nem látott rejtett és rendszer attribútumú állományok bepakolását. Ha módosítóként az 1 szám kerül mögé (/a1), akkor lehetővé teszi az aktuális könyvtár összes állományának és a belőle közvetlenül nyíló alkönyvtáraknak az elrakását. Ilyenkor az üres könyvtárakat is bepakolja. Az /a1 kapcsoló akkor fontos, ha a tárolt program kinyitása után az elindításkor az addig üres könyvtáraknak is a helyükön kell lenniük.

/b

Bepakoláskor használatos kapcsoló, amely egyfajta szűrőként szolgál. Használatakor csak a megváltozott állományokról készít backup-ot, melyek archív bitje be van kapcsolva. Ha módosítónak az 1 számot írjuk mögé (/b1), akkor a Backup készítése mellett az adott fájl archív bitjét is beállítja. Ha a módosító a 2 szám (/b2), akkor nem szűri meg a bepakolandó állományokat, csak az archív bitet kapcsolja le a bepakolt állományokon.

/c

Minden parancs mellett használható. Hatására az ARJ sem bepakoláskor, sem kipakoláskor nem ellenőrzi az állományok keltezését.

/d

Bepakoló parancsok mellett használható. Hatására a már sikeresen bepakolt forrásállományokat törli az ARJ. Az „a” parancssal együtt használva pontosan ugyanaz a hatása, mint az m (= Move) parancsnak. A tényleges törlés előtt engedélyt kér a törléshez.

/e

Bepakoláskor használatos, a nevekről leveszi az elérési útvonalat. Amennyiben módosítója az 1 szám (/e1), akkor a báziskönyvtár nevét is leszedi a nevekről bepakolás előtt.

/f

Mind bepakoló, mind kipakoló parancsok mellett használható. Hatására az ARJ csak azokkal az állományokkal foglalkozik, amelyek mind az archívban, mind a báziskönyvtárban megvannak és csak a frissebbeket mozgatja.

/g<jelszó>

Jelszavas titkosítást végez. A jelszó nem tartalmazhat szóközt. A titkosított állományokat csak a jelszó megadásával lehet kibontani, nyomtatni, megnézni és tesztelni. Az l és v parancsokkal megjelenített listákon a jelszavas védelemet a G jelzi a jobb szélső mező negyedik pozíciójában.

/i

Az /i kapcsoló minden parancs mellett alkalmazható, hatására az ARJ nem írja ki a tömörítési/kibontási folyamatot követő kijelzéseket. Az /il kapcsolóra az ARJ egy alternatív grafikus indikátort használ a folyamatok követésére.

/k

A kapcsoló hatására az ARJ a bepakoló parancsok alkalmazásakor megőrzi az archív backup-ját. Az esetleg ott maradt régi mentés törlődik.

/l<fájl.kit>

Minden parancssal használható. Hatására a fájlmaszknak megfelelő állományok neveit elhelyezi a megadott nevű állományokban. Jól használható listafájlok előkészítésére.

/m

A bepakoló parancsok mellett használatos kapcsoló, az alkalmazandó tömörítés típusát választhatjuk meg vele.

Kiegészítő módosítói: 0, 1, 2, 3, 4.

Az /m0 hatására az ARJ tömörítés nélkül egyszerűen csak bepakol. Alkalmazásának legfőbb értelme az, hogy az így bepakolt állományokat a vírusok nem látják.

Az /m1 a maximális tömörítést biztosítja (ez az alapértelmezés).

Az /m2 kapcsoló kevesebb memóriát igényel, de gyengül a tömörítés határfoka.

Az /m3 még gyorsabb, de tovább romlik a tömörítés.

A leggyorsabb, de legrosszabb határfokú tömörítést a /m4 kapcsoló eredményezi.

Az /m1 és az /m2 memóriai igénye 282 kilobájt. Az /m3 249, az /m4 pedig csak 235 kilobájtot kér.

A /jm kapcsolóval a /m1 tömörítése mintegy 1-8%-kkal növekedhet.

/n

Csak új fájlokat pakol be vagy ki, tehát amelyek az archívban, illetve a célkönyvtárban még nincsenek meg.

/o<yymmddhhmmss>

Ez a kapcsoló technikai jellegű. Mindegyik paranccsal kombinálható. Hatására az ARJ csak azokkal az állományokkal foglalkozik, amelyek a megadott dátumnál nem régebbiek (pl. az 1991 június 29-nél nem régebbi keltezésű állományok kiválasztására: /o910629). A számokat kettesével adjuk meg: év-hónap-nap-óra-perc-másodperc sorrendben.

Nem kötelező végig mindent megadni. Az 1991-nél nem régebbi állományokat a /o91 kapcsoló választja ki. A 80-nál kisebb évszámot az ARJ program XXI. századnak tekinti.

/ob<yymmddhhmmss>

A fenti kapcsolóhoz hasonló ez is, a különbség a kettő között annyi, hogy az /obxxxx a megadott időpont előtt készült állományokkal foglalkozik. A kettő kombinációjával egy adott időintervallumban keletkezett állományokat gyűjthetünk ki (pl. az 1990. május 7. és 1991. április 1. közöttiek az /o900507 /ob910401 kapcsolópárossal).

/p

A kapcsoló minden paranccsal használható. Hatására az ARJ a teljes, a path-t is tartalmazó nevet használja, csak azokkal az állományokkal foglalkozik, amelyeknél a bepakolt állomány és a forrásfájl path-adatai azonosak. Ha módosítónak az 1 számot írjuk mögé (/p1), akkor az ARJ a könyvtárak alkönyvtárait is megnézi és használja.

/q

A j, l, t, v és w parancsok mellett nem, máshol azonban mindenütt használható. Hatására az ARJ először minden egyes állománynál megkérdezi, hogy elvégezze-e a megadott parancsot.

/r

Minden parancs mellett használható kapcsoló. A báziskönyvtár alkönyvtárait is figyeli.

/s

A bepakoló parancsokkal használatos a kapcsoló, de a nem aktualizáló parancsokkal is együttműködik. Az archív dátumbejegyzését módosítja, konkrétan a legfrissebb bepakolt állományhoz igazítja. Az /s1 kapcsoló megőrzi a módosított archív eredeti keltezési bejegyzéseit. Az /s2 csak a módosító parancsokkal működik, hasonló eredménnyel, mint a /s kapcsoló.

/t

A kapcsoló a bepakoló parancsokkal használható. A fájl típus kézi beállítását teszi lehetővé. A /t0 (alapértelmezés) a bináris típust jelöli ki, a /t1 a szöveges (text) típust. Ha a /t1 kapcsolót használjuk, nem lehetnek grafikus és nyomtató vezérlő karakterek a szövegben, különben adatvesztés is történhet.

Bináris állományokra ne használjuk a /t1 kapcsolót, mert akkor nagy a valószínűsége az adatvesztésnek. A /t egyenértékű a /t0 beállítással.

/u

Mind a bepakoló, mind a kipakoló parancsokkal használható kapcsoló. Hatására csak az új és a frissebb állományok kerülnek be vagy ki.

/v

A /v kapcsolók engedélyezik a több részes archívval való munkát. Minden parancs mellett használhatók. Egy-egy archívrésszel végezve az ARJ megáll, lehetőséget adva a lemezcserére. A kapcsolónak rengeteg módosítója lehet, akár egyszerre is. Kibontáskor nem kell méretparamétert megadni a /v után.

A többrészes archívok létrehozásakor ajánlatos a /v kapcsolók mellett a /jt kapcsolót is alkalmazni, ez jelentősen megnöveli az adatbiztonságot.

A /vv hangjelzést is ad az új archívra váltáskor. Ha használjuk, akkor ez legyen a /v kapcsolók között az első.

A /va automatikusan érzékeli a szabad területet.

A /vas kapcsoló automatikusan érzékeli a szabad területet és lehetővé teszi, hogy külső parancsokat, programokat hívjunk meg (visszatérés EXIT parancssal).

A /vas<parancs> a -va kapcsolóval azonosan dolgozik, csak a végrehajtás előtt meghívja a megadott programot (a megadott <parancs> lehet DOS belső parancs, végrehajtható EXE, vagy COM állomány, illetve batch-fájl. Amennyiben a meghívott programnak paramétereit is át akarunk adni, akkor az egész kapcsolót " " jelek közé kell tenni ("/vaslha m e:vacak c:\vacak*. *"). Ha az utolsó átadott paraméter könyvtárnév és \ karakterrel fejeződik be, akkor a \ és a " jel közé egy szóközt kell tenni ("/vaslha a e:vacak c:\vacak\ ").

A bepakoló parancsokkal együtt alkalmazott /v<xxxx> kapcsoló az archív méretét maximalizálja. Akkor célszerű, amikor adott szabad lemezterületet akarunk kitölteni. Az ARJ úgy végzi az archív feltöltését, hogy a negadottnál legfeljebb 200—3000 bájtal lesz kisebb az archív. A kommentek növelik az archív méretét, ezért inkább kisebb értéket adjunk meg a /v mellett, ha kommenteket is adunk.

Néhány kitüntetett értéket rövidíteni is lehet: /v360 = /v36200, /v720 = /v73000, /v1200 = /v121300, és végül /v1400 = /v1457000.

Az ezres értékeket K-val is rövidíthetünk. Itt a K 1000-et jelent, nem pedig a megszokott 1024-et. Így például a /v360 nagyobb, mint a /v360K.

/w<directory>

A bepakoló parancsokkal használható. Munkakönyvtár kijelölésére szolgál. Hatására az átmeneti állományokat nem az aktuális könyvtárban, hanem pl. a winchester megadott könyvtárában, vagy a ram drive-on hozza létre. A program működésének gyorsítására szolgál, valamint elkerülhetjük vele a disk full (megtelt a lemez) üzenetet, illetve nem szemetelünk feleslegesen az aktuális könyvtárba.

Arra azonban vigyáznunk kell, hogy a /w kapcsoló használatakor az eredeti archív felülírása előtt az ARJ nem ellenőrzi a szabad tárterület nagyságát.

Az átmeneti állományokat az új archív építések ARJTEMP.\$?? néven hozza létre az ARJ. A kiterjesztés utolsó két karaktere egy szám. Ha már van egy ARJTEMP fájl a

munkakönyvtárban, akkor az új átmeneti fájl kiterjesztésének a száma eggyel nagyobb lesz.

/x<maszk>

Ez a kapcsoló hasonló szolgáltatást nyújt, mint a PKZIP-ben: fájlmaszkkal megadhatjuk a bepakolásból, kipakolásból, törlésből és listázásból kizárandó állományok nevét. A dzsóker karakterek használata a fájlmaszkban megengedett. Szintén a PKZIP-hez hasonlóan itt is alkalmazható listafájl, de nem @, hanem ! bevezető karakter után (/x!<fájl.kit>). Ha az /x kapcsoló után csak egy könyvtárnevet adunk meg (pl. /xc:\kakuk\), akkor az abban lévő összes állományt kizárjuk az ARJ műveleteiből.

/y

Minden igen/nem jellegű visszakerdezésnél helyettünk igennel válaszol. Minden parancs mellett alkalmazható, célszerűen elsősorban batch-állományokban.

/z[<fájl.kit>]

A c, e és x parancsok mellett használható kapcsoló. Az archívhoz fűzendő archív komment szövegét tartalmazó fájl adható meg ezzel a kapcsolóval. Ha SFX archívot akarunk készíteni, akkor ajánlatos egy olyan feliratot (logót) készíteni, amely az SFX kibontásához szükséges minden információt megad. Ha a c parancs mellett használjuk, akkor az ARJ csak az archív kommentet írja be. A /z hatására a promptról adhatjuk be a maximum 25 soros kommentet. Ha a komment legelső sorába egy bevezető ! karakter után beírjuk a listafájlt, az ARJ onnan veszi a kommentet. Ha a /z<fájl.kit> alakot használjuk, akkor az ARJ közvetlenül onnan tölti át a megjegyzés szövegét. Mindhárom esetben a komment maximális hossza 2048 bájt lehet. Ha a listafájl hosszabb, akkor is csak az első 2048 bájt épül be, a többi elvész.

Ha az „e” parancs mellett használjuk a /z<fájl.kit> kapcsolót, akkor az ARJ a megadott állományba írja ki az archív kommentet.

Ha az x mellett használjuk, akkor az ARJ kibontja az állományokat s az archív kommentet a megadott állományba küldi.

/ja

Minden paranccsal használható. Engedélyezi az ANSI vezérlő kódokat tartalmazó kommentek használatát. Az ARJ alapértelmezésként nem írja ki az Escape karakterét.

/jc1

Minden parancs mellett alkalmazható.

/jd

A kibontó és kilistázó parancsokkal alkalmazható együtt. Az e /jd50K kombináció csak akkor nyit ki, ha az adott fájl kibontása után 50 K-nál nagyobb szabad lemezterület marad. Az l /jd1000 kombináció pedig az errorlevel értéket módosítja, ha 1000 bájtnál kevesebb a hely. A parancs alapértelmezése /jd0, azaz csurig megtölti a lemezt.

Az l és v parancsok mellett használva a kapcsolót, az ARJ megnézi, hogy a választott állományok kibontás után ráférnek-e céllemezre a megadott méretű hely üresen hagyásával.

/je

A bepakoló parancsokkal használható együtt. SFX állományok készítését biztosítja. A /je1 kisméretű, de „butább” SFXJR-t állít elő (lásd részletesebben az ARJ-SFX-eket tárgyaló részben). Ha már létező ARJ fájlból akarunk SFX-et csinálni, akkor az f parancsot és egy még nem létező „...” fájlnevet kell a /je kapcsoló mellett használnunk (ARJ f archiv ... /je).

A kapcsoló nem támogatja a többrészes archívokat és az /m4 tömörítő opciót. Maga az SFX fej az LZEXE segítségével lett tömörítve. (Az ARJ-SFX és az ARJ-SFXJR állományokról részletesen lásd az ezzel foglalkozó fejezetet.)

A /je1 nem támogatja az /m4, /t1 és /g<jelszó> kapcsolókat. A vele előállított SFX csak kibont, mintha /x kapcsolót használna, path-ra tesz, könyvtárt visszaállít. Ha relatív elérési útvonalat adtunk meg bepakolásakor, akkor a célkönyvtárból kiindulva építi fel a szerkezetet.

/jf

A bepakoló parancsok mellett használható kapcsoló. A teljes megadott elérési útvonalat beírja az archívba.

/jh<xxxxxx> (2048—65535)

A bepakoló parancsok mellett használható kapcsoló. A tömörítéshez használt Huffman puffer méretét lehet beállítani 2048 és 65535 bájt közötti értékre (az alapértelmezés /jh16384, ami a bináris állományokra van optimalizálva). Minél nagyobb a puffer, annál tömörebb lehet az archív, sőt egy kicsit gyorsul is a program.

/ji<indexfájl.kit>

Hatására az ARJ egy indexfájlt csinál, amely tartalmazza a bepakolt állományok neveit. A létrehozott indexfájl kétféle sort tartalmaz. Az egyik a technikai adatokat tartalmazza (dátum, időpont, a fájl kezdőpozíciója, archív név), a másik csak a fájlnevet tartalmazza. Az indexfájlt ne floppy lemezen hozzuk létre, hanem winchesteren.

/jk

A bepakoló parancsok mellett használható kapcsoló. Ha hiba történik a tömörítés közben, akkor az ARJ az átmeneti archívot olyan állapotba hozza, hogy a sérült részt eltávolítva az használható legyen.

/jm

Bepakoló parancsok mellett használjuk a maximális tömörítés biztosítására. Az /m1 és /m2 kapcsolókból kihozza a maximumot. (Az /m1 /jm páros átlagosan 1-8%-kal tömörebb archívot eredményez.)

A /jm1 valamivel kisebb tömörséget ér el a /jm kapcsolónál eredményez, de jelentősen gyorsabb.

/jn<fájlnév.kit>

A többrészes archívokkal használjuk. Ha egy félbeszakított bepakolást folytatni akarunk, akkor a /jx kapcsolót adjuk meg, és a program a beírt névig minden állományt átugrik.

/jo

Az a és az m parancsok mellett használjuk. Ha bepakoláskor az archívban már van azonos nevű állomány, akkor a /jo kapcsoló hatására a program megkérdezi, felülírhatja-e.

/jp

Az archív katalógust kiíró l és v parancsok mellett alkalmazható kapcsoló. Azt biztosítja, hogy a képernyő megteltekor a program vár az <ENTER> lenyomására és csak utána írja ki a következő képernyőnyi részletet.

/jr

Minden paranccsal alkalmazható. Sérült archívok kijavítására szolgál. Ajánlatos a /q kapcsolóval kombinálni.

/js<.xxx>

A bepakoló parancsok mellett használható. A különböző típusú archív állományokat a megadott kapcsolóval az ARJ tömörítés nélkül csomagolja és nem bajlódik velük. Ez időnyereséget jelent.

Az alapértelmezés szerint az ARJ nem próbálkozik feleslegesen az .ARJ, .ARC, .LZH, .ZIP, .PAK állományok további zsugorításával. Ha újabb kiterjesztést is ki akarunk zárni a tömörítésből, akkor a pontot is oda kell írni a kapcsolóba. Pl. a ZOO és LZH állományok kizárására a /js.zoo.lzh kapcsolót kell alkalmaznunk.

Ha az ARJ_SW változó tartalma = /js.arj.zip.lzh /js- , akkor a /js paranccsal a felsorolt három archív típust az ARJ nem tömöríti, csak bepakolja.

/jt

A bepakoló parancsok mellett használható kapcsoló. Az átmeneti állományok teszteléséről gondoskodik. Ha a /w kapcsolóval együtt használjuk, akkor az ARJ először ellenőrzi az archívot és csak akkor törli a forrásállományokat, ha nem történt hiba.

/ju

Általánosan használható kapcsoló. Az Unix stílusú Path lefordítását végzi.

/jv

A v paranccsal együtt alkalmazva csak a bepakolt állományok neveit listázza ki. Ha /jv1 formában alkalmazzuk a v parancs mellett, akkor néhány plusz oszloppal bővül a kijelzés, ami igen előnyös a rendezéshez. Az l, t, e és x parancsokkal használva több információt jelenít meg a program.

/jx<xxxx>

A bepakoló parancsok mellett használható kapcsoló. Hatására adott pozíciótól indul a program.

Akkor használjuk, ha egy többrészes archív létrehozása valami hiba miatt megszakadt, és kézi vezérléssel akarjuk befejezni a pakolást. A megadott <xxxx> bájttól építi tovább az archívot.

/jy[a/c/d/k/n/o/s/v]

Arra használható ez a kapcsoló, hogy elnyomjuk az ARJ yes-t, vagy no-t kérő egyes promptjait. A módosítókkal együtt (egyszerre több módosító is adható a /jy kapcsolóhoz) a következő feladatokat látja el a kapcsoló:

/jya Többrészes állományoknál az append kérdésre helyesel.

/jyc Kérdés nélkül hozhat létre alkönyvtárat.

/jyd Kérdés nélkül töröl.

/jyk A szabad lemezterület bekérése elmarad.

/jyn Az új fájlnevet bekérő prompt elmarad.

/jyo Felülír.

/jys Továbbmegy, nem vár.

/jyv Továbbmegy a következő részre, nem vár.

/jz<fájl.kit>

A c (= Comment) parancs mellett alkalmazható kapcsoló. A fájlkommentet tartalmazó állományt adhatjuk meg vele. Az ARJ onnan írja majd be a fájlkommentet.

Az ARJ extra szolgáltatásai

i (= Integrity of ARJ.EXE)

Csak az LHA rendelkezik hasonló szolgáltatással: nemcsak az archív, hanem önmaga épségét is ellenőrizni tudja. Az i paranccsal az ARJ önmagát teszteli le. A 3.0 feletti verziójú DOS-ok esetén az i parancs után nem kell beírni az ARJ.EXE nevét.

w (= Where is)

E paranccsal megnézhetjük az ARJ programmal, hogy az archív mely bepakolt állományaiban található a megadott szövegrészek (maximum 78 karakterig). A program ilyenkor felsorolja, miben mit talált és hányszor. A 2.10 verzió azzal is bővült, hogy a keresés előtt még két dolgot megkérdez: először is, hogy a kis és nagybetűket azonosnak vegye-e, másodsor, hogy a talált szöveget a környezetével együtt jelezze-e ki. A talált szöveg kiírt környezetének méretét határozhatjuk meg a /jv kapcsolókkal: /jv = 6 sor, /jv1 = 12 sor, /jv2 = 18 sor, /jv3 = 24 sor.

A kereső parancs nem találja meg azokat szövegeket, amelyek a többrészes archívokban kettészakítva tárolódnak.

Egyszerre 20 különböző karaktersorozat adható meg. Akkor kezd keresni a program, ha az utolsó beadott szöveg után egy üres sort is beadunk. Egy-egy szövegrész (sor) lezárása <ENTER>-rel történik.

j (= Join archives)

Ez a parancs archívok összefűzésére szolgál. Használata előtt mindössze egy üres (al)könyvtárban kell elhelyezni az összedolgozandó ARJ archívokat. Más fájl nem lehet az adott könyvtárban, mert hibát okozna. Ezután egyszerűen kiadjuk az ARJ j <főarchív> parancsot és az ARJ szépen beépíti a megadott archívba a könyvtárban található összes archívot. Ha az /r kapcsolót is alkalmazzuk, akkor az adott könyvtár alkönyvtáraiban lévő ARJ archívokat is beépíti.

A j parancs nem támogatja az alábbi kapcsolókat: /o, /ob, /f, /u, /c, /d.

A felsorolt parancsokon kívül extra szolgáltatás még a sérült archívokból való adatmentés (/jr kapcsoló).

Az ARJ-SFX állományokat egyszerűen visszaalakíthatjuk normál ARJ archívvá az alábbi séma szerint: ARJ j <új név> <SFX név.EXE>

Az ARJ kijelzett errorlevel értékel

Hibakód	Jelentése
0	Nem történt hiba.
1	Csak figyelmeztető, nem végzetes hiba: az ARJ nem találta a megadott állományt, nem találta az archívot.
2	Végzetes hiba történt.
3	CRC hiba (header, vagy fájl CRC).
4	Hiba az ARJ-SECURITY blokkban, vagy egy ilyen blokkban lévő állományt próbáltunk módosítani.
5	Teli a lemez, vagy egyéb okból nem tudott írni a lemezre.
6	Az archívot nem tudta megnyitni.
7	Hibásan adtunk meg egy parancsot vagy kapcsolót.
8	Nem elég a RAM.

A fenti adatok felhasználásával könnyen írhatunk olyan batch alkalmazásokat, melyekben az errorlevel érték figyelésével ágaztatjuk el a programot a megfelelő irányba.

Környezeti változó használata

Az eredeti leírások szerint az ARJ egyetlen környezeti leíró változót (environmental variable) használ. Ez pedig az ARJ_SW. Benne elhelyezhetjük azokat a kapcsolókat, amelyeknek az alapértelmezését módosítani kívánjuk. Például az AUTOEXEC.BAT-ba beszúrt SET ARJ_SW=/wc:\temp -k -e -jm hatására az ARJ a C:\TEMP könyvtárba szemetel, megőrzi az archívok Backup-ját BAK kiterjesztéssel, bepakolásakor nem tárolja a path adatokat és maximális tömörítést használ.

Tippek, trükkök jó tanácsok

- Érdemes a backup lehetőségeket kihasználni.
- Az SFX-eket jelöljük meg a fájlnevében (pl. a fájlnev utolsó karaktere legyen mindig egy # jel).
- Használjunk listafájlokat.
- A sok jelszót könnyű elfelejteni, ezért jelszóval titkosított archívot minél kevesebbet és minél ritkábban csináljunk.
- A /jh65535 kapcsolóval gyakran tovább javíthatjuk a tömörítést.
- A p és s parancsokat csak olyan szövegekre használjuk, amelyek biztosan nem tartalmaznak ékezetes betűket és 127 feletti kódú karaktereket. (Azok megtekintésére használjuk a SHEZ 6.2 és a LIST programokat.)
- Az ARJ.EXE és a REARJ.EXE összenyomható a PKLITE programmal.
- A REARJ program a path-on lévő tömörítők és az ARJ.EXE segítségével átkonvertálhatja a winchesteren található összes archívot.

- A parancssorban maximum 64 fájlnevet adhatunk meg.
- Dzsóker karakterekkel legfeljebb 16 000 fájlt választhatunk ki egyszerre.
- Az archív rendezésére használjuk az ARJSORT programot, amely a csomag szerves része.
- Az /x kapcsolóval egyszerre legfeljebb 8 000 állomány kezelését tilthatjuk le.
- Legfeljebb 8 000 ARJ állományt adhatunk meg dzsóker karakterekkel.
- Ne feledkezzünk meg arról, hogy a kommentek hossza nem haladhatja meg az ARJ programnál a 2048 bájt hosszúságot. Ez megfelel 25 darab 80 karakteres sornak. Ha a listafájl ennél hosszabb szöveget tartalmaz, a többi elvész.
- Ismert probléma, hogy a Super PCKwik 3.30 cache programmal kiakad az ARJ, amennyiben a cache-t /D+ paraméterrel használjuk. A többi hasonló programok, pl. a PCTools csomag PC-CACHE 5.5 és 6.0 verziói ilyen problémát nem okoztak.
- Az ARJ210.EXE csomag tartalmazza a REARJ programot is. Ez a segédprogram a főbb archiváló programokat ismeri és segítségével tetszőleges formátumra konvertálhatunk bármely ismert archív formátumból. Természetesen a célarchív alapértelmezése ARJ.
- A SHEZ 6.2 verziója már teljes mértékben támogatja az ARJ archívokat és ARJ-SFX-eket is.

A jó öreg PKARC és PKPAK

A PKPAK & PKUNPAK 3.61 csomag a Phil Katz féle PKARC csomag egyenesági leszármazottja. Parancsaik teljesen megegyeznek, csak néhány apróbb, alig észrevehető hibát javítottak ki a programokban.

A PK361.EXE nevű SFX állományban van összegyűjtve a csomag minden szükséges állománya:

- PKPAK.EXE — Archívot létrehozó program.
- PKUNPAK.EXE — Archívot kinyitó program.
- PKSFX.PGM — Segédállomány SFX archív létrehozásához.
- APPNOTE.TXT — Az archív fájl szerkezetének leírása.
- MANUAL.DOC — A PK361.EXE-ben levő programok kézikönyve.
- PKPAKJR.EXE — Archívot létrehozó program kis memóriához.
- PKUNPAKJ.EXE — Archívot kibontó program kis memóriához.
- README.DOC — Eligazító.
- ORDER.DOC — Megrendelő szöveg.
- MAKESFX.COM — Segédprogram, ami a PKSFX.PGM-et hozza létre.
- SOFTDEV.DOC — Tájékoztató.

A PKARC-hoz és a PKZIP-hez hasonlóan itt is határozottan szétválasztották a be- és kipakolást, valamint az SFX létrehozását, három külön-külön alkalmazható programba, így egyszerre kisebb programot kell betölteni és az gyorsabban is tud dolgozni, mintha mindhárom egyetlen állományban lenne. A kis szabad RAM-mal rendelkező rendszerek számára fejlesztették ki a PKPAKJR és PKUNPAKJ programokat. Ezek parancsai megegyeznek a PKPAK, illetve PKUNPAK parancsaival, csupán az alkalmazható tömörítő módszerek száma valamint az egy menetben kezelhető állományok száma csökkent.

A programcsomag parancsai megfelelnek a tömörítő programoknál általánosan használtaknak:

- | | |
|-----------------------|---|
| a = Add | = Bepakolás, valamiféle tömörítéssel. |
| f = Freshen | = Frissítés, újat nem visz be. |
| u = Update | = Aktualizálás, újat is bevisz. |
| m = Move | = Áthelyezés (a forrásállományok törlésével). |
| d = Delete | = Fájl törlése az archívból. |
| e = Extract | = A PKUNPAK kibont, visszaállít. |
| x = eXtract | = A PKUNPAK kibont, visszaállít. |
| t = Test | = Az archív épségét teszteli (PKUNPAK). |
| p = Print | = Nyomtatás. |
| c = Console | = Képernyőre kibontás (PKUNPAK). |
| c = Comment | = Megjegyzések hozzáfűzése (PKPAK). |
| g = Garbage | = Jelszavas titkosítás. |
| v = View/Verbose list | = Bővített archívkatalógust ad. |
| h = Help | = Segédinformációk. |

Az adatbiztonságról

1. A leggyakoribb hibaforrás az ember. Egy kis odafigyelés sokat segíthet.
2. A Move parancs illetve kapcsoló használata esetén csak akkor törli a program a forrásállományokat, ha sikeresen és hibaüzenet nélkül be tudta pakolni az állományokat az átneveti (\$\$\$ kiterjesztésű) ARC fájlba, majd sikeresen létre tudta hozni az új, ARC kiterjesztésű archívot.
3. A véletlen felülírásokat megfelelő parancsok alkalmazásával kivédhetjük.
4. A PKPAK 16-bites CRC értékkel ellenőrzi az állományok épségét.

A bepakolás

Az ARC állományokba új fájlokat a PKPAK programmal tudunk berakni. Az állományok bepakolására a négy szabványos parancs szolgál, valamint egy új, az i (selective backup), melyek közül az a, u, i és m parancsokkal új archívokat is létre lehet hozni, nemcsak a már meglévőt bővíteni.

-a (= Add)

Mindenképpen bepakolja a megadott állományokat az ARC fájlba, függetlenül attól, hogy volt-e már az archívban azonos nevű állomány és mikori keltezéssel. Könyvtárszerkezetet és path információt a PKPAK nem tárol.

-u (= Update)

Az Add parancshoz hasonlóan működik, de megvizsgálja azt is, hogy van-e már a bepakolásra kijelölttel azonos nevű állomány a ZIP fájlban. Ha igen, csak abban az esetben írja felül a már bent lévő, ha az régebbi keltezésű. Ez a dátumellenőrzés kikapcsolható, s ilyenkor az Add parancsnak megfelelően dolgozik a PKPAK.

-f (= Freshen)

Kizárólag frissítést végez. Az Update parancshoz hasonlóan a már bent lévőknél frissebb keltezésű állományokat pakolja be, az új állományokat nem. Értelemszerűen új archív létrehozására sem alkalmas. Használati területe a munkaállományok tömörített biztonsági másolatainak mindennapos frissítése a munka végeztével.

-m (= Move)

Megspórolhat vele sok felesleges munkát. Az Add parancshoz hasonlóan minden kijelölt állományt keltezésre való tekintet nélkül bepakol, majd a forrásállományokat (ha nem történt hiba a tömörítés és az archív kiírása közben) kitörli, természetesen a readonly attribútumú állományok kivételével.

-i (selective backup)

Egyedülálló a tömörítő programok körében. Csak és kizárólag azokat az állományokat pakolja be az archívba, amelyek még nem szerepelnek abban. Így tökéletesen kiegészíti a Freshen parancsot.

A program a DOS-hoz hasonlóan nem látja és nem is kezeli a hidden (rejtett) és a system (rendszer) állományokat, így igen jól használható vele a NO.COM program is. (Ismertetése megtalálható a könyv segédprogramokkal foglalkozó fejezetében.)

A fenti alapparancsok közül a három első használható az m (Move) kapcsolóval kombinálva is. Az am parancs (Add + Move) teljesen egyenértékű az m (Move) parancssal.

A Freshen és az Update parancsoknak a Move kapcsolóval való bővítése (mf, illetve mu) azt eredményezi, hogy a bepakolásra kijelölt állományok az új ARC fájl sikeres létrehozását követően törlődnek, ha csak nem voltak readonly-nak jelöltek. Azok az állományok törlődnek a Move parancs használatakor, melyeknél ténylegesen megtörtént a bepakolás, viszont azok, amelyeket az ma, mf, vagy mu parancs nem frissített (mivel régebbiek voltak a már bennlévőknél), nem törlődnek az új ARC fájl létrejötte után.

Lemezeken végzett archív frissítéskor könnyen beleütközhetünk a lemezkapacitás korlátaiba. Ez legtöbbször kivédhető a -b parancs alkalmazásával. (Lásd részletesen a speciális szolgáltatásoknál.)

Kipakolás

A PKPAK párja, a PKUNPAK gondoskodik az ARC állományok kibontásáról, nyomtatásáról. Ha létrehoztunk, vagy kaptunk egy ZIP fájlt, valamikor ki is akarjuk csomagolni. A kipakolás iránya, célja szerint is több csoportra oszthatjuk a kipakoló parancsokat. Ezek a következők:

Egyszerű kipakolás

-e és az -x (= EXtract)

A parancs két alakja teljesen egyenértékű. Egyszerűen kipakolnak mindent az archívból az adott maszknak megfelelően a parancssorban megadott könyvtárba. Különböző kiegészítő parancsokkal természetesen módosíthatjuk működését. A PKUNPAK mind az e, mind az x parancs használatakor rákérdez, hogy felülírjon-e, ha egy olyan állományt akar kinyitni, amely a célkönyvtárban már szerepel. Ez az üzenet elnyomható, ha eleve biztosak vagyunk abban, hogy csak az archívból kinyitható állományokra van szükségünk.

-n (= Newer)

Felülírás elleni védelemmel nyílnak ki a fájlok az archívból. Ez a parancs nem engedi, hogy az archívban lévő régebbi verzió felülírjon a célkönyvtárban már bent lévő újabb verziójú fájlt.

-r (= Replace)

Akkor használatos az EXtract parancs mellett, ha a célkönyvtárban azonos néven már létező állományokat feltétlenül felül akarjuk írni az archívból kibontandó állománnyal.

Átnézés, nyomtatás

-c (= Console)

A PKUNPAK Console parancsa a képernyőre nyitja ki a fájlokat. Természetesen csak szövegfájloknál célszerű használni. Egy képernyőnél hosszabb szövegek átnézésére rendelkezésünkre áll a m (More) alparancs is. A -cm egyszerre egy képernyőnyit jelenít meg az ARC fájlba bepakolt szövegfájlból. Ilyenkor a következő mozgási lehetőségünk van:

<SPACE> — A következő képernyőnyi anyagot mutatja meg.

<ENTER> — Egy sorral tovább megy.

<ESC> — A következő bemutatandó állományra ugrik.

<CTRL C>, vagy <CTRL BREAK> Megszakítja a programot.

-p (= Print)

Nyomtatóra küldi a kinyitott fájlokat, nem pedig lemezre. Szöveget és rajzokat is lehet így nyomtatni közvetlenül az ARC fájlból. A parancs módosító paramétereit nem tudjuk szétválasztani, csak elhagyhatjuk a feleslegeseket. A további paraméterek nélkül kiadott -p parancs nem kapcsol a nyomtatón más üzemmódra. A Print parancs a rendelkezésünkre álló kísérő paraméterekkel a következő lehet: -p[a/b], ahol

-pa — A nyomtatón szöveges üzemmódba kapcsolva nyomtat. Az egyes állományok végén lapot dob.

-pb — A nyomtatón bináris (grafikus) módba kapcsolva nyomtat. Nem dob lapot az egyes fájlok után.

Az ARC fájl épségének ellenőrzése

-t (= Test)

Az archív épségét ellenőrzi azáltal, hogy látszólag kibontja az archívot, de nem írja lemezre a kibontott állományokat, „kipakolás” közben pedig egy CRC ellenőrzést végez.

Fájl törlése az archívból

Az ARC fájlokból a szabványos -d (= Delete) parancs segítségével törölhetünk ki állományokat. Ilyenkor a PKPAK egy új ARC fájlt hoz létre, melyből egyszerűen kihagyja a törlésre kijelölt állományokat. A Delete parancs használatakor nem írhatunk az ARC fájl nevébe dzsóker karaktereket, és legalább egy törlendő állomány nevét meg kell adnunk (ezekben már engedélyezett a dzsóker karakterek használata).

A Delete parancs csak a PKPAK néhány parancsával használható egyszerre, így a -b paranccsal.

Megjegyzések, jelszavas védelem

A -x (= archiv comment)

Az ARC fájlhoz maximum 32 karakternyi megjegyzés írható. Az archív komment megjelenik az ARC fájl neve után, ha az állományt PKPAK-kal vagy a PKUNPAK-kal kezeljük. Ékezetes betűk is bevihetők. A régi archív komment felülíródik.

-c (= Comment)

Maximum 32 karakternyi megjegyzés fűzhető az egyes bepakolt állományokhoz. A már létező komment felülírható (ha akár egyetlen szóközt beírunk, a régi komment már törlődik). Ékezetes betűk is bevihetők.

A -g (= Garbage)

Jelszó megadásával titkosíthatjuk a bepakolt fájlokat. A jelszó maximum 63 karakter hosszú lehet, és nem tehetünk bele szóközt. A -g parancs és a jelszó között sem lehet szóköz. A Garbage parancs különbséget tesz a kis- és nagybetűk között. A titkosított fájl csak akkor nyitható ki a PKUNPAK-kal, ha megadjuk a jelszót is. Ha elfelejtjük a jelszót, akár ki is törölhetjük a titkosított állományt az ARC fájlból.

Archív katalógus megnézése

A PKPAK, mint minden fájl-tömörítő program, külön katalógust vezet a tömörített (archív) állományba bepakolt fájlokról. Ezt a listát nevezem archív katalógusnak. Ebben a következő bejegyzések szerepelnek: fájlnev, eredeti hossz, alkalmazott tömörítési módszer, tömörített méret, százalékos méretcsökkenés, keltezés dátuma, időpontja, 16-bites CRC, attribútum, megjegyzés, titkosított-e, mivel készült. Ez az archív katalógus a `-v` parancsokkal hozható elő. A PKPAK és a PKUNPAK `-v` parancsa megegyezik, az eltérés mindössze annyi, hogy a PKPAK a `-vm` paranccsal lapozhatóan adja ki a listát, illetve a `-vc` paranccsal a megjegyzéseket is kiírja, a PKUNPAK `-vv` parancsával pedig arról kaphatunk információt, hogy milyen nevű és verziószámú programmal hozták létre az adott tömörített állományt. A parancs általános formája, és adható paraméterei: `-v[cm]`.

`-v` (View technical information)

File name	Fájlnev a kiterjesztéssel.
Length	A fájl eredeti hossza.
Method	A fájl-tömörítésére használt módszer.
Size	A tömörített fájl mérete.
Ratio	Méretcsökkenés az eredeti fájl méret százalékában.
Date	A fájl dátuma.
Time	A fájl időpontja.
CRC	A fájl 16-bites CRC értéke.
Total Files	Az archívban lévő fájlok száma.
Total Length	Az eredeti fájl méretek összesen.
Total Size	A tömörített fájlok mérete összesen.
Total Ratio	A zsugorítás aránya %-ban.

`-vc`

A meglévő fájl-kommentek kijelzésére használatos. Az alábbi adatokat írja ki a bepakolt állományokról:

File name	Fájlnev a kiterjesztéssel együtt.
CRC	A fájl CRC értéke (16-bites).
Comment	Megjegyzések (max. 32 karakter).
Total Files	Az archívban lévő fájlok száma.
Total Length	Az archivált fájlok eredeti mérete összesen.
Total Size	A tömörített fájlok mérete összesen.
Total Ratio	A zsugorítás aránya %-ban.

`-vv`

A PKUNPAK kiírja, hogy az egyes állományok milyen verziószámú tömörítővel készültek. (Pl. ha PK 1.1 a kiírás, akkor PKARC 1.1 verzióval, ha PK ≥ 2.0 jelenik meg, akkor a PKARC 2.0 verziónál újabbal. Három kérdőjel (???) esetén a PKUNPAK nem ismerte fel a tömörítőt.)

Az ARC fájl és az archív katalógus fizikai és logikai szerkezetéről röviden szó esik ennek a kötetnek a végén is, de részletes ismertetésük csak a készülőben lévő második könyvbe fér bele. Abban többek között szerepel majd az itt hely- és időhiány miatt kimaradt vagy újabban megjelent számos program, a tömörítés matematikai és számí-

tástechnikai alapjainak és a fontosabb tömörítő algoritmusoknak a bemutatása, továbbá sok példa, forráskódokkal illusztrálva.

Az önkinyitó állományok archív katalógusa megnézhető a PKPAK, vagy a PKUNPAK segítségével, de az SFX igen színvonalas szolgáltatásai között is szerepel a -v parancs is.

Önkinyitó (SFX) archívok

A PKPAK két külön állományt használ az SFX archívok létrehozására. Ezek közül a MAKESFX program másolja ki a PK361.EXE állomány SFX fejrészét PKSFX.PGM néven, amit aztán egyszerűen össze kell fűznünk a már elkészített archív állománnyal a következő módon:

```
COPY/B PKSFX.PGM + archív.ARC SFXfájl.EXE
```

Az ARC fájl változatlanul megmarad, de lesz egy azonos nevű SFX fájl is, EXE kiterjesztéssel. A Copy parancs /B paramétere feltétlenül szükséges, nem elhagyható.

A kész ZIP-SFX saját parancsai

Az ARC-SFX állományok normál ARC fájlként is használhatók, de ilyenkor az archív EXE kiterjesztését is be kell írni. Egyszerűbb azonban az SFX saját parancsainak használata, amelyek megegyeznek a PKUNPAK parancsaival (csak a -vv parancs hiányzik az SFX parancskészletéből).

SFX_fájl [-parancsok] [célkönyvtár] [lista]

- x Kibont (alapértelmezés). Ha azonos nevű fájlt talál a célkönyvtárban, akkor rákérdez, felülírja-e. (Az -e ugyanezt teszi.)
- c Képernyőre bont ki, mint a PKUNPAK esetén.
- cm Képernyőre bont ki, mint a PKUNPAK esetén, a More parancssal képernyőnként lapozgatva.
- n Csak azokat a fájlokat bontja ki, amelyek frissebbek a célkönyvtárban lévőknél.
- r Mindent kibont és kérdezés nélkül felülír.
- t Az SFX fájl épségét ellenőrzi.
- g<jelszó> Az SFX fájl is támogatja a jelszavas védelmet.
- l Licenc képernyő.
- h Help képernyő. Akkor is bejön, ha nem értelmezhető parancsot kap.

Tippek az SFX fájlok használatához

Álljunk a célkönyvtárba és az A:-ról hívjuk meg az SFX-et

```
C:\munka>A:SFXfájl -d
```

Az A:-ról nyissuk ki a fájlt a célkönyvtárba:

```
A:\>SFXfájl -d c:\munka *.doc
```

Az SFX archívok ajánlott használata

1. Ha nincs, hozzuk létre a célkönyvtárat, amennyiben külön könyvtárban akarjuk elhelyezni a kipakolt állományokat. (Pl.: md c:\uj\kipakolt)
2. Lépünk a célkönyvtárba/meghajtóra (Pl. c:, majd cd \uj\kipakolt)
3. Hívjuk meg az SFX-et, megadva annak elérési útját is. (Pl. c:\uj\kipakolt>a:pk361)

Természetesen ettől el is térhetünk, de ez a legegyszerűbb és a legbiztonságosabb használata az SFX állományoknak.

Parancsok, kapcsolók

A PKPAK és a PKUNPAK parancsai egybetűsek, némelyiknek további paraméterei lehetnek, amelyeket a parancstól nem lehet elválasztani, legfeljebb elhagyni. A parancsok előtt egy bevezető mínusz (-) karaktert kell kitenni. Több parancs is összevonható egyetlen közös bevezető - jel mögé, csak arra kell vigyázni, hogy ne keveredjenek a parancsok és a parancsok kiegészítő paraméterei. A parancsok egyaránt írhatók nagy- és kisbetűvel, de célszerű megmaradni a kisbetűs formánál.

A PKPAK 3.61 parancsai

Bepakoló parancsok

-a	Hozzáadás (Add).
-u	Aktualizálás (Update).
-f	Frissítés (Freshen).
-i	Szelective backup.
-m	Áthelyezés (Move).
-ma	Áthelyezés (Move).
-mu	Áthelyezés (Move) aktualizálással.
-mf	Áthelyezés (Move) frissítéssel.

Megjegyzések, jelszavas védelem

-x	Megjegyzés az ARC fájlhoz (archiv comment).
-c[m]	Megjegyzések a bepakolt fájllokhoz (Comment).
-g<jelszó>	Jelszavas védelem (Garbage).

Technikai jellegű parancsok

-b [<drive>][<path>]	Átmeneti munkakönyvtár kijelölése.
-o[tc]	ARC kompatibilis állományokat készít.
-n[tc]	PKPAK kompatibilis állományokat készít.

Egyéb parancsok

-v[cm]	Archív katalógus megjelenítése.
-d	Törlés az ARC fájlból.
-h	Help képernyőt ad.
-l	Licenc képernyőt ad.

A PKUNPAK 3.61 parancsai

Kipakoló parancsok

-e	A kibontó parancs.
-x	A kibontó parancs.
-n	Csak újabb állományt nyit ki.
-r	Felülír kinyitáskor.

- c[m] Képernyőre nyit.
- t Teszteli az ARC fájl épségét.
- p[ab] Printerre nyit.

Technikai és egyéb parancsok

- g<jelszó> Jelszavas védelem (Garbage).
- v[v] Archív katalógus megjelenítése.
- h Help képernyőt ad.
- l Licenc képernyőt ad.

Az ARC-SFX archívok parancsal

Kipakoló parancsok

- e A kibontó parancs.
- x A kibontó parancs.
- n Csak újabb állományt nyit ki.
- r Felülír kinyitáskor.
- c[m] Képernyőre nyit.
- t Teszteli az ARC fájl épségét.
- p[ab] Printerre nyit.

Technikai és egyéb parancsok

- g<jelszó> Jelszavas védelem (Garbage).
- h Help képernyőt ad.
- l Licenc képernyőt ad.

A konfigurációs fájl

A PKPAK a DOS-ból és a saját konfigurációs állományából (PKPAK.CFG) megadott környezetleíró változókat használ alapértelmezései beállításához. A DOS-ból megadható/megadandó változók:

SET PKPAK.CFG=C:\ZIP (Hol található a PKPAK.CFG fájl.)

SET PKPAKTMP=D: (A ramdrive legyen az átmeneti munkakönyvtár.)

Ezt a két sort ajánlatos az AUTOEXEC.BAT-ba beszúrni.

A PKPAK és a PKUNPAK alapértelmezéseit megváltoztathatjuk egyszerűen a PKPAK.CFG nevű konfigurációs fájl változóinak módosításával. A konfigurációs fájlt először az aktuális könyvtárban keresi a PKPAK. Ha ott nem találja, akkor a PKPAK.CFG változóban megadott úton. Ha ott sincs, akkor ahonnan a PKPAK indult (DOS 3.x verzióktól kezdve).

Ahol több érték is rendelhető egy változóhoz, ott az egyes értékeket törtvonallal (/), vesszővel (,) vagy szóközzel elválasztva kell beírni.

A PKPAK.CFG konfigurációs állományának változói:

FILECHECK=normal/relaxed

A Normalt választva a program megáll, ha nem talál egy kijelölt állományt. A Relaxed választásakor ezt csak egy üzenettel jelzi és dolgozik tovább.

ARCCOPY=C:\

A munkakönyvtár megadására szolgál.

TEMPNAMES=fixed

Rögzített átmeneti fájlneveket használ, hogy a hálózati alkalmazásoknál ne kerüljön összeütközésbe más programokkal.

ARCTIME=latest

Az archív frissítésekor, módosításakor az archív dátumát a legfrissebb bepakolt állományhoz igazítja.

SQUASH=disable/enable

A PKPAK a SEA ARC program által is használt módszerekkel dolgozik, ha a disable írjuk be ide.

Speciális szolgáltatások

Listafájl

A PKPAK és a PKUNPAK is használhat úgynevezett listafájlokat, amelyekbe beírhatjuk azokat a fájlneveket és maszkokat amelyeket egyébként a parancssorba íránk. A listafájl egy közönséges, semmilyen formázó és vezérlő karaktert nem tartalmazó ASCII fájl, amit az ilyen szövegállományokat előállító szövegszerkesztőkkel szerkeszthetünk meg. A listafájl a tömörítőkkel végzett munkák dokumentálására szolgál, illetve megkímélhetjük vele magunkat a gyakorta ismétlődő hosszú listák újbóli begépelésétől.

Egy bevezető @ karakter jelzi a program számára, hogy listafájl következik. Ha a listafájl nem az aktuális könyvtárban van, akkor a @ jel és a listafájl neve közé be kell írni a listafájl elérési útját is. A listafájl ne legyen 32 K-nál nagyobb. A parancssorban kedvünkre keverhetjük a fájlmaszkokat és a listafájlokat. A listafájl belsejében @ karakterrel jelezzük a könyvtár váltást és a rákövetkező sorokban felsorolt állományokat a program a @ karakterrel jelzett könyvtárban keresi.

Munkakönyvtár, átmeneti állományok

-b[<meghajtó:>][<path>]

A PKPAK alapértelmezésként mindig az aktuális lemezegység aktuális könyvtárába írja be átmeneti állományait. Mivel ez — különösen floppykon dolgozva — állandó helyproblémákat okoz, célszerű tudatni a programmal, hogy hová szemeteljen. Ez a helyproblémák megoldásán túl gyorsítja is a munkát, főleg, ha megfelelő méretű ramdrive, vagy gyors winchester áll rendelkezésünkre.

Amennyiben a a PKPAK indítása előtt a PKPAKTMP változót meghatároztuk és célszerűen az AUTOEXEC.BAT fájlba a

SET PKPAKTMP=<drive:\munkakönyvtár>

sort beírtuk, akkor a -b parancsra, ha nem adunk meg meghajtó- és könyvtárnevet, a PKPAK az előzőleg kijelölt lemezre és könyvtárba szemetel. Másik lehetőség a PKPAK.CFG állomány ARCCOPY változójával az átmeneti könyvtár megadása.

A -b parancs, ha használjuk, mindig rögzítetten, a PKPAK utáni első helyen legyen. A -b parancs erősebb, mint a PKPAKTMP, vagy az ARCCOPY változó. A PKPAK arra használja az így megadott könyvtárat, hogy az új ZIP fájlt \$\$\$ kiterjesztéssel ezen a gyors meghajtón hozza létre, és ha elkészült, akkor egyszerűen rámásolja a már elavult

állományra. Főleg akkor ajánlatos a használata, ha lassú és hamar megtelő lemezzel dolgozunk.

Technikai parancsok

-o[tc] (= Old compatibility)

A parancs helye a -b mögött van. Hatására a PKPAK az ARC program által is kezelhető archívot hoz létre. A t szerepe az archív dátumozásánál jön elő: a legfrissebb bepakolt állományhoz igazítja az archív keletkezését. A c paraméter itt az alkalmazott tömörítési módszerek közül kizárja a squashingot, ami a SEA ARC programjában nincs meg.

-n[tc] (= New compatibility)

A PKPAK.CFG-ben letiltott, fejlettebb tömörítési és dátumkezelési módszereket lehet ezzel a paranccsal újra engedélyezni. A t a dátumozás, a c a squashing alkalmazás kapcsoló paramétere.

-h (= Help)

Ha nem írunk be archív nevet, a PKPAK vagy a PKUNPAK nem tudja értelmezni az általunk beírt parancsokat. Ha a -h parancsot használjuk, akkor a program kiírja az adható parancsok és fontosabb paramétereik listáját, valamint a parancssor szintaxisát.

-l (= Licence)

A legtöbb tömörítőtől eltérően az -l parancs nem az archív katalógust írja ki, hanem a program Licenc képernyőjét jeleníti meg.

Errorlevel értékek, batch alkalmazások

A PKPAK illetve a PKUNPAK által kijelzett hibakódok (errorlevel) jelentése:
0 Nincs hiba.

Csak figyelmeztető jellegű hibakód:

1 Rossz CRC, a fájl nem archív, ismeretlen tömörítési módszer, nem tud új archívot létrehozni, a squeeze tábla (a tömörítési algoritmus táblája) érvénytelen.

Programfutás megszakításával járó hibakódok:

2 Elfogyott a szabad RAM.

3 Nem találta a kibontandó archívot.

4 Nem tudja megnyitni az archívot.

5 Olvasási hiba az archívban.

6 Belső táblázat túlsordulása.

7 A lemez megtelt.

A fenti értékek ismeretében olyan batch-fájlokat írhatunk, amelyek az errorlevel értékének vizsgálatával ellenőrzik a programok futását, s jelzik — akár ékes magyar nyelven —, hogy mikor, milyen hiba történt.

LAN támogatás

Mind a PKPAK, mind a PKUNPAK úgy nyitja meg az állományokat a hálózatokon, hogy egyidejűleg más taszkok is használhatják azokat. A PKPAK átmeneti állományainak egyedi neveket ad, így nemigen kerül összeütközésbe a párhuzamosan futó egyéb alkalmazásokkal hálózati meghajtókon sem.

A PKPAK és a PKUNPAK a gyakorlatban

A PKPAK használatakor a következő szerkezetet használjuk:

PKPAK [-b] [-parancsok] archív [maszkok] [>kimenet]

A PKUNPAK használatakor a következő szerkezetet használjuk:

PKUNPAK [-parancs] archív [célkönyvtár\] [maszk] [>kimenet]

A PKPAK csomag programjait célszerű egy path-on lévő könyvtárba tenni, ahol a DOS könnyen megtalálja és nem kell a program minden egyes meghívásánál az elérési útvonallal bíbelődni.

A parancsok helye nem kötött, a PKUNPAK mögött bárhol lehetnek a sorban. Ha nem adunk parancsot, akkor az alapértelmezés a Extract parancs.

Az archív nevét mindig meg kell adni. Az Add, Update, Freshen, Move, selectIve backup és Delete parancsok használatakor az archív nevében nem szerepelhetnek dzsóker (* és ?) karakterek, az összes többi parancs használatakor igen. Az archív neve természetesen tartalmazhat elérési útvonalat is. A PKUNPAK után megadható célkönyvtár végére ne felejtjük el kitenni a \ jelet.

A maszkok lehetnek fájlnevek, melyekben a dzsóker (* és ?) karaktereket szabadon használhatjuk, tartalmazhatnak elérési útvonalat is mind a be-, mind a kikapcsoláshoz. A maszkok között listafájlt is megadhatunk.

A kimenetet nem maga az PKPAK vagy a PKUNPAK program, hanem a DOS használja. Segítségével átirányíthatjuk a különben csak a képernyőn megjelenő szövegeket egy megadott fájlba, vagy nyomtatóra. Hasznos ez a lehetőség, amikor például az archív katalógust ki akarjuk nyomtatni, vagy egy listafájllhoz akarjuk felhasználni, esetleg munkánkat és a PKWare programok üzeneteit akarjuk dokumentálni.

Gyakorlati feladatok, megoldások

1. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, de nem akarjuk a könyvtár tartalmát elveszíteni.

```
C:\MUNKA>PKPAK a:munka c:\munka\konyv\*.*
```

2. A C:\MUNKA könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az archív elkészültével ki akarjuk írítani a könyvtárat.

```
C:\MUNKA>PKPAK -m a:munka c:\munka\konyv\*.*
```

3. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, kivéve a *.BAK állományokat.

```
C:\MUNKA>NO \munka\konyv\*.BAK PKPAK -m a:munka c:\munka\konyv\*.*
```

4. Csak a napi munka során megváltozott állományokat akarjuk az A:-ban lévő lemezen tárolt MUNKA.ARC fájlban egy kicsit felfrissíteni.

```
C:\MUNKA>PKPAK -f a:munka c:\munka\konyv\*.*
```

5. A C:\MUNKA és a C:\MUNKA\KONYV könyvtárakból minden .DOC kiterjesztésű állománnyal aktualizáljuk az A:MUNKA.ARC állományt. A forrásállományokat nem akarjuk törölni.

```
C:\MUNKA>PKPAK -u a:munka c:\munka\konyv\*.doc c:\munka\*.doc
```

6. A C:\MUNKA\KONYV könyvtárból minden nem .COM kiterjesztésű állománnyal bővítjük az A:MUNKA.ARC állományt. A forrásállományokat nem akarjuk törölni.

```
C:\MUNKA\KONYV>NO *.com PKPAK -a a:munka *.*
```

7. A MUNKA.ARC archívba tévedésből bekerült BAK, ARC és ~DO kiterjesztésű állományokat kell eltávolítanunk.

```
C:\MUNKA>PKPAK -d a:munka *.bat *.~do *.arc
```

8. Meg akarjuk nézni, mi van a MUNKA.ZIP állományban.

```
C:\MUNKA>PKPAK -vcm a:munka
```

Tömör listát ad megjegyzésekkel, egyszerre egy képernyőnyi anyag jelenik meg.

```
C:\MUNKA>PKPAK -v a:munka
```

Teljes listát ad, egyenként felsorolva az állományokat.

9. Ki akarjuk listáztatni a MUNKA.ARC tartalmát egy szabadon szerkeszthető ASCII állományba, de csak a nevekre és a kiterjesztésekre vagyunk kíváncsiak. (Pl. egy listafájl készítését tervezgetjük.)

```
C:\MUNKA>PKUNPAK -v a:munka >lista.lst
```

10. Gyanús a lemez. Ellenőrizni akarjuk a MUNKA.ZIP épségét.

```
C:\MUNKA>PKUNPAK -t a:munka
```

11. Meg akarjuk nézni az egyik (pl. az ARC.DOC) állományt.

```
C:\MUNKA>PKPAK -cm a:munka arc.doc
```

12. Megtetszett a fenti ARC.DOC. Nyomtassuk ki az LPT1: kimenetre kapcsolt nyomtatón, ahol előzőleg grafikát nyomtattunk.

```
C:\MUNKA>PKUNPAK -pa a:munka arc.doc
```

13. A régi verzióju PKPAK programmal pakolt KAKUKK.ARC nevű állományt egy kicsit tömörebbé akarjuk tenni, de azért a régi KAKUKK.ARC-ot is meg akarjuk őrizni.

```
C:\MUNKA>PKUNPAK a:kakukk c:\atmeneti\
```

```
C:\MUNKA>PKUNPAK a:kakukk2 c:\atmeneti\*.*
```

14. A MUNKA.ARC-ból a .TXT kiterjesztésű állományok a C:\TXT könyvtárba, a .DOC kiterjesztésűek a C:\DOKSI könyvtárba csomagolandók ki.

```
C:\MUNKA>PKUNPAK a:munka c:\txt\ *.txt
```

```
C:\MUNKA>PKUNPAK a:munka c:\doksi\ *.doc
```

15. Maszek állományainkat titkosítva akarjuk a MUNKA.ARC archívba áthelyezni, hogy a C:-n nyoma se maradjon.

```
C:\MUNKA>PKPAK -mugna_találd_ki_mi_a_jelszó a:munka maszek.*
```

Az ősi ARC

A PC-ken már az első lemez meghajtók megjelenésétől kezdve használtak adat- és állománytömörítő segédprogramokat. A néhai CP/M-es időkben LBR (library) kiterjesztésű, a DOS elterjedésével később rendszerint ARC kiterjesztésű tömörített archívokat (ARC fájl) alkalmaztak.

A tömörítés és a tömörítő illetve kibontó programok fejlesztésében úttörő szerepet játszott 1985-től a SEA (System Enhancement Associates, Inc.).

Programjuk, az ARC széles körben elterjedt, hosszú időn keresztül szabványnak, viszonyítási alapnak számított. Elhibázott üzletpolitikájuk miatt azonban a felhasználók és a bedolgozásra kapható külső fejlesztők többsége elpártolt tőlük. Ha nyitottabb és rugalmasabb politikát folytatnak, és nem erőltetik a pert a PKWare ellen, (amit végül ugyan megnyertek, de azzal nem sokra jutottak), akkor a két cég együttműködéséből nagyszerű fejlesztések születhettek volna. Így azonban minden jó tulajdonsága elenére a felhasználókért folytatott versenyben az ARC lemaradt vetélytársai mögött és programjuk már nem tartozik az élmezőnybe, sem sebességét, sem tömörségét, sem kezelhetőségét, sem pedig extra szolgáltatásait tekintve. Azért ismertetjük mégis részletesen, mert a széles körben elterjedt és jól használható tömörítő programok közül ez volt az első.

A program legfontosabb parancsai hasonlóak a többiéhez:

a = Add	= Bepakolás, valamiféle tömörítéssel.
f = Freshen	= Frissítés, újat nem visz be.
u = Update	= Aktualizálás, újat is bevisz.
m = Move	= Áthelyezés (a forrásállományok törlésével).
d = Delete	= Fájl törlése az archívból.
e = Extract	= Kibont, visszaállít.
x = eXtract	= Kibont, visszaállít.
t = Test	= Az archív épségét teszteli.
l = List	= Az archívkatalógus kiírása.
p = Print	= Képernyőn megjelenít vagy printeren kinyomtat.
v = View/Verbose list	= Bővített archívkatalógust ad.

A program támogatja a listafájl alkalmazását is.

A bepakolás

Az állományok bepakolására a négy szokványos parancs szolgál, amelyek közül az a, az u és az m parancsokkal új archívokat is létre lehet hozni, nemcsak a már meglévőt bővíteni.

Az a (= Add) parancs mindenképpen bepakolja a megadott állományokat az ARC fájlba, függetlenül attól, volt-e már az archívban azonos nevű állomány és mikori

keltezéssel. Két azonos nevű állományt csak akkor tesz be az ARC fájlba, ha egy külön kapcsoló paraméterrel (lásd a kapcsolók ismertetését) erre kifejezetten utasítottuk.

Az u (= Update) parancs az Add parancshoz hasonlóan működik, de megvizsgálja, hogy van-e már a bepakolásra kijelölttel azonos nevű állomány az ARC fájlban. Ha igen, csak abban az esetben írja felül a már bent lévő, ha az régebbi keltezésű.

Az f (= Freshen) parancs kizárólag frissítést végez. Az Update parancshoz hasonlóan a már bent lévőnél frissebb keltezésű állományokat pakolja be, új állományokat azonban nem épít be az archívba. Értelemszerűen új archív létrehozására sem alkalmas. Használati területe a munkaállományok tömörített biztonsági másolatainak mindennapos frissítése a napi munka végeztével.

Az m (= Move) paranccsal sok felesleges munkát takaríthatunk meg. Az Add parancshoz hasonlóan keltezésre való tekintet nélkül minden kijelölt állományt bepakol, majd (ha nem történt hiba a tömörítés és az ARC fájl kiírása után) a már sikeresen bepakolt forrásállományokat kitörli, természetesen a readonly attribútumú állományok kivételével.

A fenti alapparancsok közül a három első használható az m (Move) kapcsolóval kombinálva. Az am parancs (Add + Move) teljesen egyenértékű az m (Move) paranccsal.

A Freshen és az Update parancsoknak a Move kapcsolóval való bővítése (fm, illetve um) azt eredményezi, hogy a bepakolásra kijelölt állományok az új ARC fájl sikeres létrehozását követően törlődnek, függetlenül attól, hogy az adott fájl frissítését el kellett-e végeznie az ARC-nak, vagy sem. Ebben eltér a hasonszórú tömörítő programoktól. (A tömörítők zöménél csak azok az állományok törlődnek a Move parancs vagy kapcsoló használatakor, amelyeknél ténylegesen megtörtént a frissítés.)

Ha a Move parancsot vagy kapcsolót a z kapcsolóval együtt használjuk, akkor (a teljes alkönyvtár sikeres bepakolása után) a kiürült könyvtárat is törli (ha nem volt benne readonly, hidden vagy system attribútumú állomány).

A bepakoló parancsok mellett kapcsolókat, módosítókat használhatunk, amelyek korlátozzák, kiterjesztik, kiegészítik, vagy módosítják a parancsok működését. (Ezek ismertetését lásd később a parancsok és a kapcsolók részletes leírásánál és a gyakorlati példák ismertetésénél.)

Basic nyelvű programjainkat, ha a tömörség az időnél fontosabb, tároljuk ASCII formában, mivel így az ARC jóval tömörebbre tudja bepakolni, mint a tokenizált formát. (Lásd a Basic kézikönyveket a kétféle mentéssel kapcsolatban.)

Félig-meddig ide tartozik még a c (Convert) parancs is. Ez a parancs az ARC régi verzióival készült ARC fájlokat alakítja át a minden valószínűség szerint tömörebb ARC 6.02 formátumra. A parancs hatására az ARC az átalakításra kibontott állományokat ideiglenesen egy \$ARCTEMP.CVT fájlban helyezi el az aktuális vagy az ARCTEMP változóval megadott munkakönyvtárban, majd a sikeres konvertálás után törli ezt az állományt.

Még egy fontos információ a bepakoló parancsokról: a már meglévő archív aktualizálásakor és bővítésekor az ARC nem akarja bepakolni sem az ARC fájl aktualizálandó verzióját, sem az azt megelőző BAK kiterjesztésű verziót. Ezzel gyakran helyet és felesleges munkát, bosszankodást takarít meg a felhasználónak.

Kipakolás

A kipakolás iránya és célja szerint több csoportra oszthatjuk a kipakoló parancsokat. Ezek a következők:

Egyszerű kipakolás

Az **e** (= Extract) = Egyszerűen kipakol mindent az adott archívból az adott maszknak megfelelően a megadott könyvtár(ak)ba. Természetesen különböző kapcsolókkal módosíthatjuk működését.

Az **x** (= eXtract) = Megegyezik az „e” parancssal. Mindkét fenti parancsra érvényes, hogy megadhatunk akár több célkönyvtárat is, pl. a \C könyvtárba a *.C, a \DOKSI könyvtárba pedig a *.DOC és *.txt állományokat pakoljuk ki. Nem bontható ki azonban egy menetben ugyanaz a fájl egyszerre két célkönyvtárba.

A program mind az **e**, mind az **x** parancs használatakor rákérdez, hogy felülírjon-e, ha egy olyan állományt akar kinyitni, amelyik a célkönyvtárban már szerepel. Ez az üzenet elnyomható, ha biztosak vagyunk abban, hogy tényleg csak az archívból kinyitható állományokra van szükségünk.

Átnézés, nyomtatás

A **p** (= Print) = A Print parancs a standard kimenetre nyitja a kipakolandó állományokat, alapértelmezésben a képernyőre, így ez a parancs alkalmas az ARC fájlba bepakolt szövegek megnézésére kibontás nélkül. Ne várjunk túl sokat ettől a lehetőségtől, mert a program gyorsan, szinte olvashatatlanul görgeti végig a fájl szövegét. Az is zavaró lehet, ha a szöveg 80 karakteresnél szélesebbre van formázva. Bináris állományok (program, grafika, nem ASCII adat) megnézése a Print parancssal legtöbbször sípoláshoz és a képernyő teleszemeléséhez vezet. A DOS átirányító > operátorával a kimenet egyszerűen átküldhető a nyomtatóra (>PRN, vagy >LPT1:, soros portra kötött printerek esetén: >COM1:). A nyomtatót nem tudja a nyomtatandó állománynak megfelelő grafikus vagy text üzemmódba átkapcsolni.

Az archív fájl épségének ellenőrzése

A **t** (= Test) = Az archív épségét ellenőrzi azáltal, hogy látszólag kibontja az archívot, de nem írja lemezre a kibontott állományokat, „kipakolás” közben pedig egy CRC ellenőrzést végez.

Program futtatása archívból

Az **r** (= Run) parancs = Az ARC egyedülálló lehetősége volt, hogy a bepakolt állományok közül a COM, EXE, BAT és a .BAS kiterjesztésűeket futtatni tudja. Ez tulajdonképpen egy kis csalással megy, mert először az aktuális (vagy a ARCTEMP változóval megadott) munkalemezre a megfelelő kiterjesztéssel kibontja a kiválasztott állományt \$ARCTEMP néven, és ezt futtatja, majd törli. Ha azonos néven, eltérő kiterjesztéssel több futtatható fájl van az archívban és nem adjuk meg a kiterjesztést, akkor egymás után mindet lefuttatja.

Az eredeti angol nyelvű kísérő dokumentációkban megemlítik, hogy az így futtatott programok számára már megoldott a paraméterátadás a parancssorból, de a hogyanra nincs semmi utalás. A kézikönyv is azt ajánlja, hogy a rendszeresen futtatott programokat mégiscsak gyorsabb és célszerűbb nyitva tartani és úgy futtatni, ARC nélkül.

Fájl törlése az archívból

Az ARC fájllokból a d (= Delete) parancs segítségével törölhetünk ki állományokat. Ilyenkor az ARC egy új ARC fájlt hoz létre, melyből egyszerűen kihagyja a törlésre kijelölt állományokat.

Amennyiben a b (= Backup) kapcsolóval kiegészítve használjuk a Delete parancsot, a régi ARC fájl nem törlődik ki a könyvtárból, hanem BAK kiterjesztéssel ott marad. Esetenként szükségünk lehet erre a lehetőségre is.

Megjegyzések, jelszavas védelem

Az ARC 6.02 verziója szerencsétlen módon nem tud megjegyzéseket fűzni sem az egyes bepakolt állományokhoz, sem az ARC fájlhoz. Ez meglehetősen hátránya a többi fájl-tömörítővel szemben. Jelszavas titkosítást ellenben tud végezni. Erre a program g (= Garble) kapcsolója szolgál. A kapcsoló a bepakoló és kipakoló parancsokkal használható együtt. Szigorú megkötés, hogy a g<jelszó> a parancsot és az esetleges többi kapcsolót követő utolsó helyen legyen. A jelszó tetszőleges hosszú lehet, hossza és tartalma nincs befolyással a tömörítésre, mivel a már tömörített állományt kódolja át. A jelszó nem tartalmazhat szóközt és közvetlenül a Garble kapcsoló után kell állnia, elválasztó szóköz nélkül. A program a megadott jelszót átkódolás előtt nagybetűssé konvertálja, így a jelszó nem „case sensitive”.

Meglehetősen primitív módon oldja meg az ARC a jelszavas titkosítást, de általános, „hétköznapi” használatra még megfelel.

Az ARC katalógus megnézése

A tömörített (archív) állományba bepakolt fájlokról készített ARC katalógus megnézésére az ARC az l (= List) és a v (= Verbose list, kibővített lista) parancsokat kínálja fel.

Alapértelmezésben mindkét parancs az összes bepakolt állományt kilistázza, ezt azomban szűkíteni lehet, ha szűrő maszkot adunk (pl. *.doc — csak a *.DOC állományokat listázza ki).

A List parancs az alábbi információkat adja a bepakolt állományokról:

Name = A bepakolt fájl neve.

Length = A bepakolt fájl eredeti hossza (ennyi helyet foglal el kibontva a lemezen).

Date = A fájl utolsó módosításának, vagy a bepakolásnak a dátuma (a bepakolásakor használt kapcsolóktól függően).

Total = A lista alján szerepel, hogy a szűrőfeltételeknek megfelelő fájllokból mennyi van az archívban, s hogy ezek kipakolásához összesen mekkora szabad lemezterület kellene.

A Verbose list parancs használatával (amely egyenértékű az lv kombinációval), a fentiekén kívül a következő információkat kapjuk meg:

SF = Stowage Factor, a tömörítettség mértékét jelöli. Egészen pontosan azt, hogy a tömörítéssel az eredeti méret hány százaléka szabadult fel. Az EXE és COM program-állományoknál ez átlagosan 0-30 %, az adatállományoknál (szöveg, DBF, WK?, stb.) rendszerint 10-80 %.

Size now = Az adott fájl tömörített mérete az ARC fájlban.

Time = Az adott fájl utolsó módosításának az időpontja.

CRC = A fájl bepakoláskor képzett ellenőrző összeg. Kibontáskor és a Test parancs kiadásakor használja az ARC az állomány épségének ellenőrzésére. Az ARC 16-bites CRC-vel dolgozik.

Stowage = Az adott fájl tömörítésére alkalmazott módszer neve. Ötféle bejegyzés lehet ebben a rovatban:

— = Nincs tömörítve.

Packed = Az ismétlődő bájt értékeket kódolta át.

Crunched = Lempel-Ziv sűrítési módszert alkalmaztak.

Squeezed = Huffman kódoló technika, mint a régebbi ARC verzióknál.

Deviant = A Lempel-Ziv módszer nem standard verziójával tömörítették. Átkonvertálható a Convert parancssal.

Subdir = Alkönyvtár. A benne lévő állományok a z kapcsoló használatával kilistázhatók.

A Verbose listing parancs arra is használható, hogy megtudjuk, ráférnek-e a céllemezre a kibontandó állományok. Így nem menet közben vesszük észre, hogy kevés a hely.

Önkinyitó (SFX) archívok

Az ARC-SFX létrehozása

Az ARC.EXE egymaga nem tud SFX állományt létrehozni. Ehhez még egy MKSARC.EXE nevű segédprogram is szükséges, ami benne van a programcsomagot tartalmazó ARC602.EXE nevű SFX állományban. Az MKSARC.EXE Wayne Chin és Vernon D. Buerg alkotása.

A már elkészített ARC fájlból a következőképpen tudunk SFX állományt létrehozni a DOS promptról:

```
MKSARC ARCHIV[.ARC] [SFXfájl]
```

A kiterjesztés alapértelmezése az archívnál ARC. Ha nem adunk meg külön SFX fájlnevet, akkor az is az archív nevét kapja meg, EXE kiterjesztéssel. Az SFX méretnövekedése a forrás ARC fájlhoz képest kb. 7,8 K.

Az SFX állományok kezelése kibontó programmal

A létrehozott SFX archív ugyanúgy kezelhető az ARC programmal, mint a hagyományos ARC fájlok, csak a fájl kiterjesztését is be kell írni, ha az ARC programmal hozzáfutunk.

Az ARC-SFX-ek, mint általában az SFX állományok, igen hasznosak olyankor, amikor nincs kéznél semmilyen kinyitó program, vagy nem akarjuk az SFX címettségét külön segédprogram előkerítésével terhelni.

Az ARC-SFX használata saját parancsaival

Az ARC-SFX értelmezni tud néhány parancsot, ugyanazokat, amelyeket a Vernon Buerg által írt ARCE kibontó program.

Az ARC-SFX állományok parancssorának szerkezete:

ARCSFX [maszk] [célkönyvtár] [/parancsok] [g<jelszó>] [>prn]

A maszk tartalmazhat * és ? (dzsóker) karaktereket.

A parancsok a következők lehetnek:

/h = Help. Kiírja az SFX által használható parancsokat.

/p = Print. Alapesetben a képernyőre, a DOS átirányító >prn parancsával a printerre nyitja az állományokat. Ez sem tudja a nyomtató üzemmódját (grafikus/text) átváltani.

/r = oveRtype. Rákérdezés nélkül felülírja kibontáskor a célkönyvtárban már meglévő azonos nevű állományokat.

/t = Test. Ellenőrzi az ARC-SFX épségét a CRC értékek alapján.

/q = Quiet. Kikapcsolja a különböző (alarm, stb.) hangjelzéseket.

/5 = Nem hoz létre alkönyvtárakat.

/g<jelszó> = Garble. Jelszóval titkosított állományok kibontásához való. Ha alkalmazzuk, ez legyen az utolsó parancs.

Parancsok, kapcsolók

Az ARC 6.02 egybetűs parancsokat használ, melyeket tetszőlegesen kiegészíthetünk a kapcsolók többségével.

A parancsokat akár kis- akár nagybetűvel írhatjuk, egyformán működnek. Egyszerre egy parancsot alkalmazhatunk.

Az ARC 6.02 parancsai

Bepakoló/átpakoló parancsok

- | | |
|--------------------|--|
| a (Add) | — Feltétlen hozzáadás, mindent bevisz. |
| u (Update) | — Aktualizálás, csak újat és frissebbet visz be. |
| f (Freshen) | — Frissítés, csak frissebbet visz be, újat nem. |
| m (Move) | — Áthelyezés, megegyezik az Add parancssal, de annyiban több annál, hogy a bevitt forrásállományokat törli. |
| c (Convert) | — Az ARC régebbi verzióival készült, vagy tömörítés nélkül pakolt állományok konvertálása az ARC 6.02 tömörítési módszereivel. |

Kipakoló parancsok

- | | |
|--------------------------------|--------------------------------|
| e és x (EXtract) | — Egyszerű kibontás. |
| t (Test) | — Épségellenőrzés. |
| p (Print) | — Megnézés vagy nyomtatás. |
| r (Run) | — Bekapolt állomány futtatása. |

Katalógus parancsok

- l (List) — Rövid ARC katalógus kiíratása.
v (Verbose list) — Teljes ARC katalógus kiíratása.

Törlőparancs

- d (Delete) — Bepakolt állomány törlése ARC fájlból.

Az ARC parancsainak hatását módosító kapcsolók

A PKZIP kapcsolói a parancsokhoz hasonlóan egykarakteresek, a kis- és nagybetűk egyenértékűek, és a Garble kapcsolót kivéve tetszőleges sorrendben használhatók. Az össze nem férő kombinációkat lásd a konkrét eseteknél.

m (Move) Áthelyezés. Az u és f parancsok mellett használható jól. Az am kombináció teljesen azonos a Move paranccsal, az mm kombináció értelmetlen (lásd a bepakoló parancsokat az első témakörben).

z () Könyvtárszerkezetet is tárolja (lásd a bepakoló parancsokról az első témakörben).
w (Warn) Figyelmeztető üzenetek elnyomása. Használjuk minél kevesebbet. A kapcsoló egyetlen ésszerű felhasználása az, amikor batch-fájl segítségével akarunk egy garantáltan hibamentes archívból egy rengeteg szabad helyet tartalmazó lemezre kinyitni állományokat a képernyő teleszemtelése nélkül. Ne használjuk a Test paranccsal!

n (Note) A program által normális üzemmódban kiírt üzenetek megjelenítését tilthatjuk le ezzel a kapcsolóval. Akkor érdemes használni, ha rengeteg állományt pakolunk be vagy ki, és nem vagyunk kíváncsiak arra, hogy éppen hol is tart az ARC a munkával. Ilyenkor gyakran a Warn kapcsolóval együtt használjuk. A List paranccsal kombinálva mindössze a fájlnevek felsorolása történik meg. Ezt a kapcsolót se használjuk a Test paranccsal!

o (Overwrite) A kapcsoló azt eredményezi, hogy kibontáskor a kint már meglévő állományokat az ARC rákérdezés nélkül felülírja.

b (Backup) Hatására az aktualizálás, bővítés, törlés után az ARC fájl régi verziója nem törlődik, hanem BAK kiterjesztéssel a helyén marad. Biztonsági okokból időnként érdemes alkalmazni.

s (Suppress compression) A kapcsoló hatására a program a kijelölt állományokat tömörítés nélkül pakolja be az ARC fájlba. Mind az öt archív aktualizáló parancs (a,u,f,m,c) mellett használható.

5 (level 5 compatibility) Az ARC 5.xx verzióval kompatibilis archívokat eredményez bepakoló parancsokkal alkalmazva.

v (Verbose) Kibővített kijelzés. Bármelyik paranccsal kiadható, de nincs mindegyikre hatással. Ha hozzáadunk, vagy törlünk állományokat az ARC segítségével, normális esetben a program nem jelzi ki azokat az állományokat, amelyekkel nem foglalkozott. A Verbose kapcsolóval mindenről üzenetet küld.

g<jelszó> A titkosítást biztosító Garble kapcsoló mindig a legutolsó helyen álljon. A jelszó tetszőleges hosszúságú lehet (mérete nem befolyásolja a tömörítést), közvetlenül a g után, szóköz nélkül kell beírni, nem tartalmazhat szóközt, a kis- és nagybetűk egyenértékűek. Akár minden egyes bepakolt állományhoz külön jelszót adhatunk, de ennek semmi értelme. Minél kevesebb jelszót használunk, annál jobb. A jelszóval titkosított állományok megjelennek ugyan az ARC katalógusban, ennél többet azonban csak a jelszó megadásával tehetünk velük.

Speciális szolgáltatások

Listafájl

Az ARC is használhat úgynevezett listafájlokat, melyekbe összegyűjthetjük azokat a fájlneveket és maszkokat amelyeket egyébként a parancssorba íránk. A listafájl egy közönséges, formázó és vezérlő karaktereket nem tartalmazó ASCII fájl.

Egy bevezető @ karakter jelzi a program számára, hogy listafájl következik. Abban az esetben, ha véletlenül maga a listafájl neve is @ karakterrel kezdődne, nem kell a listafájlt átneveznünk, egyszerűen csak egy / jelet kell elérnünk.

A listafájl az ARC esetén nemcsak egy létező állomány lehet, hanem egy másik program kimenete is. Például az

```
arc ln egyik | arc a másik @
```

parancssor azt eredményezi, hogy az EGYIK.ARC állomány ARC katalógusának tömör kilistázása lesz az alapja a MASIK.ARC fájl létrehozásának, vagyis az ER-GYIK.ARC archív tartalomjegyzéke alapján bővül a MASIK.ARC.

Munkakönyvtár

Az ARC alapértelmezésként mindig az aktuális lemezegység aktuális könyvtárába írja be átmeneti állományait. Mivel ez, főleg floppyval dolgozva állandó helyproblémákat okozhat, célszerű tudatni a programmal, hogy hová szemeteljen. Ez nemcsak a helyproblémákon segíthet, hanem gyorsítja is a munkát, főleg, ha megfelelő méretű ramdrive vagy gyors winchester áll rendelkezésünkre.

Az ARCTEMP, illetve a TEMP változó tartalmazza a munkalemez azonosítóját. A program először az ARCTEMP változót keresi, s ha nem találja akkor tér rá a TEMP-re. Mivel más programok saját céljaikra is használhatják a TEMP változót, ajánlatosabb az ARCTEMP használata.

A változó legegyszerűbben úgy állítható be a megfelelő értékre, hogy az AUTO-EXEC.BAT-ba beszurjuk a következő sort: SET ARCTEMP=D: ha a D: a ramdrive vagy egy merevlemez (partíció) neve.

Az ARC a gyakorlatban

Az ARC használatakor a következő szerkezetet használjuk:

```
arc parancs[kapcsolók][g<jelszó>] archív [maszkok] [kimenet]
```

Az ARC.EXE programot célszerű egy path-on lévő könyvtárba tenni, ahol a DOS könnyen megtalálja és nem kell az elérési útvonallal bíbelődni a program minden egyes meghívásánál.

Az előző részekben ismertetett parancsok közül egyet be kell írunk az ARC mögé, attól egy szóközzel elválasztva. Ha nem írunk be az ARC által értelmezhető parancsot, akkor a Help jelenik meg, amelynek az első része a licenc információkat tartalmazza, s az <Enter> gomb megnyomása után az ARC 6.02 parancsait és szintaxisát ismertető rész gördül fel. (A kapcsolók használata nem kötelező, leírásukat lásd az előző oldalakon.)

Az archív nevét mindig meg kell adni. Az Add, Update, Freshen, Move és Delete parancsok használatakor az archív nevében nem szerepelhetnek dzsóker (* és ?) karakterek, az összes többi parancs használatakor igen. Az archív neve természetesen tartalmazhat elérési utat is.

A maszkok lehetnek fájlnevek, melyekben szabadon használhatjuk a dzsóker (* és ?) karaktereket, tartalmazhatnak elérési utat is mind a be-, mind a kipakolásukhoz. A maszkok között listafájlt is megadhatunk.

A kimenetet nem maga az ARC program, hanem a DOS használja. Segítségével átírányíthatjuk a különben a képernyőn megjelenő szövegeket egy megadott fájlba vagy nyomtatóra. Hasznos ez a lehetőség, amikor az ARC katalógust például ki akarjuk nyomtatni, vagy egy listafájllhoz akarjuk felhasználni, esetleg munkánkat és az ARC üzeneteit akarjuk dokumentálni.

Gyakorlati feladatok, megoldások

1. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, de nem akarjuk a könyvtár tartalmát elveszíteni.

```
C:\MUNKA>arc a a:munka c:\munka\konyv\*.*
```

2. A C:\MUNKA könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az archív elkészültével ki akarjuk írítani a könyvtárat.

```
C:\MUNKA>arc m a:munka c:\munka\konyv\*.*
```

3. A C:\MUNKA\KONYV könyvtár összes állományáról gyűjtő archívot kell készítenünk az A: meghajtóban lévő lemezre, és az egész C:\MUNKA\KONYV alkönyvtárt a pakolás után ki akarjuk törölni.

```
C:\MUNKA>arc mz a:munka c:\munka\konyv\*.*
```

4. Csak a napi munka során megváltozott állományokat akarjuk az A:-ban lévő lemezen tárolt MUNKA.ARC fájlban felfrissíteni.

```
C:\MUNKA>arc f a:munka c:\munka\konyv\*.*
```

5. A C:\MUNKA és a C:\MUNKA\KONYV könyvtárakból minden DOC kiterjesztésű állománnyal aktualizáljuk az A:MUNKA.ARC állományt. A forrásállományokat nem akarjuk törölni.

```
C:\MUNKA>arc u a:munka c:\munka\konyv\*.doc c:\munka\*.doc
```

6. A C:\MUNKA\KONYV könyvtárból minden nem COM kiterjesztésű állománnyal bővítjük az A:MUNKA.ARC állományt. A forrásállományokat nem akarjuk törölni.

```
C:\MUNKA\KONYV>no *.com arc a a:munka *.*
```

7. A MUNKA.ARC-ba tévedésből bekerült BAK, ARC és ~DO kiterjesztésű állományokat kell eltávolítanunk.

```
C:\MUNKA>arc d a:munka *.bak *.~do *.arc
```

8. Meg akarjuk nézni, mi van a MUNKA.ARC állományban.

```
C:\MUNKA>arc l a:munka (Csak tömör listát ad)
```

```
C:\MUNKA>arc v a:munka (Teljes listát ad)
```

9. Meg akarjuk nézni, hogy az A:-n lévő MUNKA.ARC tartalma kinyitva elfér-e a B:-ben lévő, félig már megtelt lemezen.

```
C:\MUNKA>arc v a:munka b:*.*
```

10. Ki akarjuk listáztatni a MUNKA.ARC tartalmát egy szabadon szerkeszthető ASCII állományba, de csak a nevekre és a kiterjesztésekre vagyunk kíváncsiak. (Pl. egy listafájl készítését tervezzük.)

```
C:\MUNKA>arc ln a:munka >lista.lst
```

11. Gyanús a lemez. Ellenőrizni akarjuk a MUNKA.ARC épségét.

```
C:\MUNKA>arc t a:munka
```

12. Az A:-n lévő MUNKA.ARC-ból akarjuk futtatni a Norton Editort.

```
C:\MUNKA>arc r a:munka ne.com
```

13. Meg akarjuk nézni az egyik (pl. az ARC.DOC) állományt.

```
C:\MUNKA>arc p a:munka arc.doc
```

14. Megtetszett a fenti ARC.DOC. Szeretnénk kinyomtatni.

```
C:\MUNKA>arc p a:munka arc.doc >prn
```

15. A régi verzióju ARC programmal pakolt KAKUKK.ARC nevű állományt tömörebbé akarjuk tenni, de azért a régi KAKUKK.ARC-ot is meg akarjuk őrizni.

```
C:\MUNKA>arc cb a:kakukk
```

(A régi formátumu KAKUKK.ARC KAKUKK.BAT néven megmarad.)

16. A MUNKA.ARC-ból a TXT kiterjesztésű állományok a C:\TXT könyvtárban, a DOC kiterjesztésűek a C:\DOKSI könyvtárban csomagolandók ki.

```
C:\MUNKA>arc e a:munka c:\txt\*.txt c:\doksi\*.doc
```

17. Maszek állományainkat titkosítva akarjuk a MUNKA.ARC-ba áthelyezni, hogy a C:-n nyoma se maradjon.

```
C:\MUNKA>ARC umgna_találd_ki_mi_a_jelszó a:munka maszek.*
```

„Temetni jöttem...” — DWC

A program jelen verziója 1988-as. A programozó (Dean W. Cooper) azonban az egyik kísérő DOC fájlban közli, hogy a forrásállományok és a tömörítő algoritmusok jogát 3000 dollárért eladta, tehát nem valószínű, hogy a DWC új verzióval fog gyarapodni.

A DWC nevű tömörítő nem kompatibilis a többi tömörítővel, csupán a parancskészletet vette át az ARC-tól (SEA) és azt bővítette. Így aki az ARC programot ismeri, kevés új dologgal találkozik.

Az általános parancsszerkezet a következő:

DWC parancs[kapcsolók][g<jelszó>] archív[/s] [fájlok[/s]]

A program egyszerre egy parancsot fogad el, de korlátlan számú kapcsolóval. A parancs és a kapcsolók helye szigorúan rögzített. Amennyiben a g-vel titkosítani akarunk, az az utolsó kapcsoló legyen, és a jelszót is be kell írunk.

A program átgondolatlanságára jellemző, hogy bővítés/frissítés esetén sem magát az archívot, sem pedig annak backup-ját nem rakja hozzá a csomaghoz.

A DWC parancsai, funkció szerint

Bepakoló parancsok

a = Add

Egyszerű bepakolást végez. Figyelem! Nem lehet két azonos nevű állomány egy DWC archívban, még akkor sem, ha különböző könyvtárakból származnak!

m = Move

A bepakolt állományokat törli a forráskönyvtárból.

u = Update

Aktualizál, csak újat, vagy frissebbet visz be.

f = Freshen

Csak újat, vagy frissebbet visz be. Nem tud új archívot létrehozni.

c = Convert (internal add)

Egy előzőleg bepakolt állományt újracsomagol.

i, k = Comment (attribute add)

Egy maximum 44 karakteres megjegyzés fűzhető a fájlhoz. Ez megmarad akkor is, ha az adott fájl egy azonos nevű állománnyal felülírjuk. A komment az l (List) paranccsal nézhető meg, ha nincs path info.

Törlés az archívból

d = Delete (negative add)

A megadott állományt törli.

Kibontó parancsok

e, x = Extract

A két parancs teljesen egyenértékű, hatásukra a DWC kibontja az adott állományokat. Kibontáskor egy megadott célkönyvtár addig él, amíg egy újabb célkönyvtárat nem adunk meg.

p = Print

A standard kimenetre (azaz a képernyőre) bontja ki az az állományokat. Ez a DOS átirányító operátorával (>) a nyomtatóra is küldhető.

t = Test

A CRC értékek (8 bites) alapján ellenőrzi az archív épségét.

Az archív katalógus megtekintése

l = List

Kilistázza az archívkatalógust. Ez a rövid forma. A következő adatokat mutatja meg: Fájlnév, eredeti hossz, kelezés, komment vagy path info.

v = Verbose list

Bővített listát ad az archív katalógusról: Name (fájlnév), Length (eredeti méret), Stowage (tömörítési módszer), SF (méretcsökkenés %-ban), Size now (tömörített méret), Date, Time, Pass (jelszóval védett-e).

h = Header

A Header parancssal megadható, hogy melyik bepakolt fájl tartalmazza az archív fájlkommentet. A Header fájl méretére és tartalmára vonatkozóan nem adnak meg korlátokat. Ha használjuk a header parancsot, akkor mindenképpen meg kell adnunk egy header fájlt.

= = Make data equal newest file

Az archív kelezési idejét a benne lévő legfrissebb állományéhoz igazítja, illetve, ha fájlt is megadunk, akkor ahhoz.

A DWC kapcsolói

A g kapcsoló a hozzá tartozó jelszóval mindig a legutolsó helyen legyen. A kapcsolókat hozzáírjuk a parancshoz.

s

Hatására bepakolásakor nem tömörít, csak tárol.

b

A módosítás előtt egy backup másolatot készít az archívról.

n

Letiltja a normál programüzenetek kiírását. Batch alkalmazásokban hasznos, mert nem szemeteli tele feleslegesen a program a képernyőt.

w

Letiltja a figyelmeztető üzenetek kiírását.

!

Minden üzenetet letilt a DWC futásának idejére.

q

Be- és kikapcsoláskor alkalmából minden egyes állományra rákérdez.

o

Csak újabbakat bont ki.

i

Hozzáadáskor minden egyes bepakolt állományhoz bekéri a maximum 44 karakteres fájlkommentet.

z

Maximális tömörítésre optimalizál.

y

Maximális sebességre optimalizál.

=

Az archív keltezését a legfrissebb fájlhoz igazítja.

j

Bepakoláskor a fájl listában szereplő DWC archívokat nem úgy pakolja be, ahogy vannak, hanem hozzáfűzi őket az archívhoz.

r

Bepakoláskor elrakja a könyvtárszerkezetet is, kikapoláskor pedig felhasználja.

g<jelszó>

A jelszóval elpakolt állományok csak a jelszó megadásával bonthatók ki, nézhetőek meg és tesztelhetőek. Ez a kapcsoló az utolsó legyen a kapcsolók között. A kísérő dokumentációkban semmi utalás nincs arra, hogy mit lehet, illetve mit nem lehet használni a jelszóban, és arról sincs semmi információ, hogy milyen hosszú lehet.

-

Az utána következő kapcsolót lekapcsolja, letiltja.

<szám>

Az archívkatalógus bejegyzéseinek maximális száma. Érték 100 és 1800 között lehet. Az alapértelmezés 900, ami mintegy 30 K RAM-ot igényel.

/s

Egy speciális kapcsoló az /s, amit az archív, vagy a fájl lista után kell beírni. Hatására a *.* nemcsak az adott könyvtárral, hanem annak alkönyvtáraival is foglalkozik.

A DWC környezeti változói

A RamDrive használatát, illetve az átmeneti könyvtárakat támogató a DWC az alábbi környezetleíró változókat keresi: DWCTEMP, ARCTEMP, TEMP, TMP. Ha valamelyiket már definiáltuk, akkor a többivel nem kell törődnünk. Pl. SET TEMP=D:\TEMP az AUTOEXEC.BAT-ban.

DWCDEF= Az alapértelmezésnek vett parancsok és kapcsolók, ha nem adunk parancsot a DWC meghívásakor.

DWCADD= A kapcsolók alapértelmezése bővítéskor.

TZ= Time Zone. Alapértelmezése: TZ=PST8PDT EST+/-<szám>[EDT] A Greenwich-i középidejtől való eltérés megadásához.

Batch alkalmazások támogatása

A DWC program az összes tömörítő programok közül a legalaposabban támogatja a batch alkalmazásokat. Az egyes errorlevel értékek figyelembevételével egyszerűen kézbentartható, automatizálható a tömörítés és a kibontás.

A program által beállított errorlevel (hibajelzési) értékek:

- 0 Nem történt hiba.
- 1 A programot a felhasználó <CTRL C>-vel szakította meg.
- 2 Egy vagy több hiba is történt.
- 3 Egy vagy több figyelmeztetés történt.
- 10 Illegális paraméter.
- 11 Csak kapcsolókat adtunk, parancsot nem.
- 12 Több parancsot is adtunk, pedig csak egy megengedett egyszerre.
- 13 A g mellé nem írtunk jelszót.
- 14 Nem adtunk meg archívot.
- 20 Out of memory. Kifogytunk a RAM-ból.
- 21 Törléskor nem adtunk fájlnevet.
- 22 A Header parancshoz nem adtunk meg header fájlt.
- 23 Nem tudott a NULL portra küldeni.
- 24 Nem tudja lemezre írni az archívot.
- 25 Megtelt a céllemez.
- 40 Nem létezik a megadott archív.
- 41 Nem tud átmeneti fájlt csinálni.
- 42 Váratlan EOF (fájl vége jel) az archívban.
- 43 Az archív CRC hibás.
- 44 Hiba az archívban, stack overflow.
- 45 Hiba az archívban, méret hiba.
- 46 Nem tud átmeneti DWC fájlt csinálni.
- 47 Túl nagy a sérülés az archívon, nem használható.

- 48 Nem tudja az archívot létrehozni.
- 49 Betelt a lemez, de az átmeneti fájl még megvan az átmeneti könyvtárban.
- 50 Betelt a lemez.
- 51 A megadott archív nem létezik.
- 52 Az archív nevébe dzsóker karaktert raktunk.
Nem tudja megcsinálni az aktualizálást.
- 60 Azonos néven már van fájl az archívban.
- 61 Nem volt a megadott fájl az archívban.
- 62 Nem tudta a megadott fájlt konvertálni.
- 63 Nem tudta a megadott fájlt hozzáadni.
- 64 A titkosított fájlhoz a jelszót is meg kellett volna adni.
- 65 Kinyitáskor önmagát írta felül.
- 66 Ismeretlen tömörítési módszer.
- 67 A header fájl hiányzik.
- 68 Több header fájlt adtunk meg.
- 69 A megadott fájlokat nem találta az archívban.
- 70 Archív hiba.
- 71 Archív hiba.
- 72 Archív hiba.

Az önkicsomagoló DWC (SFX)

A DWC csomaggal is létre lehet hozni SFX-et, de kissé körülményesen. Ennek lépései:

1. Létrehozzuk a DWC archívot.
2. Belerakjuk a DWCSFX.EXE állományt az alábbi módon:

```
DWC as archiv dwcsfx.exe
```

3. Átnevezzük az archívot:

```
REN archiv.DWC archiv.exe
```

4. A MAKESFX programmal kialakítjuk a végleges formát:

```
MAKESFX archiv.exe
```

Bevezetés a ZOO-lógiába

Rahul Dhesi ZOO programja Magyarországon még kevésbé ismert, pedig rendelkezik néhány figyelemreméltó képességgel:

— Egy adatállománynak egyszerre több változatát, a ZOO szóhasználatával generációját tudja egyszerre megőrizni egyetlen archívban.

— Rendkívül hosszú (akár 64 K-ig terjedő) megjegyzés szövegeket fűzhetünk az egyes bepakolt állományokhoz.

— A PKZIP-en kívül egyedül csak a ZOO rendelkezik (a tömörítő programmal együtt terjesztett) olyan segédprogrammal, amely a sérült archív állományokból ki tudja szedni a még használható részeket.

— A programot már több operációs rendszerre és sokféle géptípusra adaptálták, így lehetővé válik bizonyos szintű, akár kétirányú adatcsere az Amiga, Atari, Unix stb. rendszerek és a DOS alapú rendszerek között.

— A programcsomagban megtaláljuk a különféle segédprogramok mellett magának a programnak a forráskódját is, két különböző rendszerre, így azoknak, akik szívesen bütykölnek maguknak programokat, ez igazi csemege. Sokat lehet tanulni belőle.

— Két külön parancskészlete van. A kezdőknek szóló készlet sokkal egyszerűbb, de alkalmazásával elesünk néhány funkciótól. A sokkal több variációs lehetőséggel bíró másik parancskészlet a gyakorlottabb felhasználóknak való.

Sajnos tömörség szempontjából az 1988-as keltezésű verzió nem tartozik a tömörítők élmezőnyébe. A tesztelt programok közül egyedül az ARC gyengébb nála. Ezeket a hátrányokat azonban sok esetben tökéletesen kiegyenlítik a program extra szolgáltatásaival járó előnyök.

A ZOO legfontosabb parancsai megfelelnek az előző programoknál már megismerteknek:

a = Add	= Bepakolás, valamiféle tömörítéssel.
f = Freshen	= Frissítés, újat nem visz be.
u = Update	= Aktualizálás, újat is bevisz.
m = Move	= Áthelyezés (a forrásállományok törlésével).
d = Delete	= Fájl törlése az archívból.
x = eXtract	= Kibont, visszaállít.
t = Test	= Az archív épségét teszteli.
l = List	= Az archívkatalógus kiírása.
h = Help	= Kiírja a parancsokat és kapcsolókat.
p = Print	= Képernyőn megjelenít vagy printeren kinyomtat.

A program támogatja a listafájl alkalmazását is.

A program használata

Kezdőknek ajánlott az alábbi szerkezet:

```
zoo -<kezdő_parancs> ZOOfájl [fájl_maszkok] [>kimenet]
```

Haladó ZOO-lógusok a következőképp dolgozhatnak:

```
zoo <parancs>[kapcsolók] ZOOfájl [fájl_maszkok] [>kimenet]
```

A ZOO kétféle parancskészlete természetesen átfedi egymást. A kezdő parancsok messze nem használják ki a ZOO lehetőségeit, azonban további kapcsolók nélkül használhatók.

Parancsok kezdőknek

A kezdőknek ajánlott parancsok egy bevezető – karakterrel használandók. A kis- és nagybetűk között itt nem kell különbséget tennünk. A parancs nevét sem kell teljesen beírni, elég az első, vagy első néhány karakter. A parancs rögzített helye a ZOO utáni első munkaterület.

A kezdő parancsok két lépésben dolgoznak. Először elvégzik a nevükkel is jelzett feladatot, majd egy Pack parancsot is végrehajtva a régi archívot BAK kiterjesztésűvé írják át, ami egyben az adatbiztonságot is növeli. A haladó parancsokkal ellentétben itt nem lehet több fájlgenerációt egyidejűleg őrizni, és a path információkat sem lehet tárolni.

Parancsok haladóknak (acDeghILPTuUvVx)

A haladó parancsok elé nem kell bevezető karaktert tenni. Egyszerre csak egy parancsot használhatunk, de annak működését, hatását kapcsolók egész sorával módosíthatjuk. A haladó parancsoknál a program határozott különbséget tesz a kis- és nagybetűk között.

A ZOO haladó parancsaival együtt, azok módosítására használható kapcsolók (aAcCdEfImMnNoOpPqSu1:./.@n+—=) részletes ismertetését részben a konkrét parancsoknál, részben a parancsok és kapcsolók összefoglaló ismertetésénél lehet megtalálni.

A kezdő és a haladó parancsok közötti kapcsolatok az alábbi táblázatban foglalhatók össze:

Kezdő Haladó (+ kapcsolók) A parancsok hatása

–add	aP:	Bővíti az archívot.
–freshen	auP:	Frissíti az archívot.
–update	aunP:	Frissíti és bővíti az archívot.
–move	aMP:	Áthelyez az archívba.
–extract	x	Kipakol az archívból.
–test	xNd	Teszteli az archívot.
–print	xp	Standard kimenetre nyit.
–delete	DP	Töröl az archívból.
–list	VC	Kiírja a ZOO katalógust.
–comment		c Megjegyzések hozzáfűzése.

Bepakolás

A ZOO bepakoló parancsal kezdőknek

–add (= aP:)

A szabványosnak tekinthető fájlhozzáadást végzi és felülír, ha a ZOO fájlban már van a bepakolandóval azonos nevű állomány. Nem vizsgálja a kelteztést. A parancs új archív létrehozására is használható.

-freshen (= auP:)

Csak az archívban már bent lévő állományokat frissíti fel, új állományokat nem pakol be, ezért értelemszerűen nem használható új archív létrehozására. Ellenőrzi a fájlok keltezését is, és csak a már bent lévőknél frissebb állományokat pakolja be.

-update (= aunP:)

A parancs szinte teljesen megegyezik az előbb ismertetett freshen paranccsal, csupán annyi a plusz, hogy be tud vinni új állományokat is, valamint létrehozhat új archívet is.

-move (= aMP:)

Az add parancs kibővítésének tekinthető. A bővülés annyi, hogy a sikerrel bepakolt forrásállományokat kitörli a forráskönyvtárból.

Bepakolás haladóknak

A ZOO-nak két haladó bepakoló parancsa van, az a és az u parancsok. A két parancs működését sokféle kapcsolóval befolyásolhatjuk. Mindkét parancs alkalmas új archívok létrehozására.

— Az „a” parancs a megadott fájlokat bepakolja az archívba. Ha a megadott generációs limitet már elértük, akkor törli a legrégebbi, felesleges verziót.

— Az „u” parancs felfrissíti az archívban lévő állományokat de új fájlokat nem visz be.

A fenti parancsok működését az alábbi kapcsolókkal módosíthatjuk (ne felejtsük, hogy a haladó parancsok használatakor mind a parancsoknál és kapcsolóknál megkülönbözteti a kis- és nagybetűket):

M (Move)

A ZOO az M kapcsoló használatakor kitörli a már bepakolt állományokat a sikeres bepakolás, Pack-olás és az új ZOO fájl hibaüzenet-mentes létrehozása után.

n (New files only)

Csak a ZOO fájlban még bent nem lévő állományokat pakolja. Az u paranccsal, vagy az u kapcsolóval kombinálva nemcsak frissíti az archívet a bent lévőnél frissebb állományokkal, hanem új fájlokat is bevisz.

P (Pack archive)

A bepakolás után egy Pack-olást is végez. Ez annyit jelent, hogy a törlésre jelölt fájlokat a program kihagyja az új archívból.

u (Update)

Az a parancs az u kapcsoló hatására az u parancshoz hasonlóan működik.

: (no path info)

Path információ nélkül, a puszta neveket tárolja az archívban. Ha nem használjuk, akkor a ZOO a fájlneveket a megadott elérési útvonallal együtt helyezi el.

I (read from standard Input)

Ha ezt a kapcsolót használjuk, akkor a bepakolandó fájlneveket a ZOO a standard bemenetről várja. A DOS alatti változatnál megadhatunk ugyan egy sorban több nevet is, de egyszerre csak egy nevet fogad el. Használatakor nem lehet az archív neve után fájlnevet írni a negyedik munkaterületre.

Az alábbi két kapcsolónak csak új archív létrehozásakor van hatása.

+ (enable generations)

Ez a kapcsoló az újonnan létrehozott archívokban engedélyezi a generációk használatát. (Lásd részletesebben a speciális szolgáltatásokról szóló fejezetben.)

- (disable generations)

A - kapcsoló csak a szimmetria kedvéért került bele a ZOO kapcsolók készletébe. Használatával az új archívokban letiltjuk a generációk alkalmazását. Alapértelmezés: +.

Kipakolás, teszt, betekintés, nyomtatás

A ZOO kipakoló parancsa kezdőknek

-extract (= x)

A parancs hatására a megadott állományokat a ZOO kibontja. Ha nem specifikálunk kibontandó állományokat, akkor mindent kipakol, amennyiben van elég hely a céllemezen és nem történt hiba a kipakolás során.

A ZOO tesztelő parancsa kezdőknek

-test (= xNd)

Nem bontja ki sem lemezre, sem printerre az archívban lévő állományokat, csak az állományok sértetlenségét ellenőrzi a CRC értékek segítségével.

A ZOO klíró parancsa kezdőknek

-print (= xp)

A print parancs a standard outputra nyitja ki a kijelölt állományokat. Ha ezt a DOS átirányító >prn parancsával nyomtatóra küldjük, akkor ki is nyomtathatjuk szöveges vagy grafikus állományainkat az archívból.

A print parancs valójában úgy működik, hogy az éppen kibontott állományt a program a munkakönyvtárba átmeneti állományként kibontja, majd onnan kiadja a képernyőre vagy a printerre, utána pedig törli az átmeneti állományt.

Kipakolás haladóknak

Az e és az x parancs egyformán használható kipakolásra. A parancsokkal a következő kapcsolókat alkalmazhatjuk:

N (Not save)

Nem tárolja a kibontott állományokat, csak az esetleges hibajelzéseket jeleníti meg.

O (Overwrite)

Normál helyzetben a ZOO kibontáskor visszakérdez, hogy kibonthatja-e az adott fájlt, ha a célkönyvtárban már van azonos nevű. Ilyenkor három választásunk van: ha felül akarjuk írni az adott fájlt, akkor az y-t, ha nem, akkor az n-t nyomjuk meg, ha pedig utána mindegyik már létező fájlt felülíratjuk, akkor az a-t.

o (Overwrite)

Azonos az O kapcsolóval, ha legalább kétszer adjuk ki, különben a program figyelmen kívül hagyja.

S (Supersede)

Akkor is felülírja a célkönyvtárban lévő azonos nevű állományokat, ha azok frissebbek az archívban lévőknél. Ha nem használjuk, a ZOO alapértelmezésként nem ír felül frissebb állományokat régebbiekkel. Használatának akkor van értelme, ha egy archivált állomány régebbi verziójával mindenképpen felül akarjuk írni a frissebb változatot.

p (Pipe data to standard output)

A kimenő adatokat átirányítja a standard kimenetre. A hibaüzenetek is ott jelennek meg. Amennyiben azonban CRC hibát talál a program, mind a képernyőn, mind az átírányítás helyén megjelenik a megfelelő hibaüzenet.

/ (eredeti path-ra nyit)

Minden szükséges alkönyvtárnak már léteznie kell. Ha nem használjuk a kapcsolót, akkor az aktuális könyvtárba nyit ki a program mindent. Ha duplázva írjuk be (//), akkor nem szükséges a célkönyvtáraknak létezniük, a ZOO létrehozza azokat.

d (Deleted too)

A töröltnek jelölt állományokat is kibontja.

dd (Deleted)

Csak a töröltnek jelölt állományokat bontja ki.

Az O, az N és a p kapcsolók kölcsönösen kizárják egymást.

Fájl törlése az archívból

Állománytörő parancs kezdőknek

-delete (= DP)

A delete parancs elegánsan kitörli a megadott állományt az archívból majd egy Pack-ot is végez. Ezáltal a fájl visszahozhatatlanul törlődött a ZOO fájlból. Hogy azért ne essünk kétségbe, a régi verzió még megmarad, benne az előbb kitörölt állománnyal, de már BAK kiterjesztéssel.

Törő parancs haladóknak

A D (Delete) parancs szolgál a haladóknak készített parancskészletben az eltávolítandó állomány törlésére. Az így törölt állomány az U (Undelete) paranccsal megmenthető, mivel még történt fizikai törlés, csupán a törlési jelzés került rá az adott állományra. Ha

a törölendő állománynak több generációja is van az archívban, akkor az 1-es kapcsolóval egyetlen fájlra korlátozhatjuk a törlés érvényességét.

A törlő és a visszahozó paranccsal használható kapcsolók: P, q, 1. Ezek közül a P kapcsoló arról gondoskodik, hogy a művelet végén a ZOO egy Pack-ot is végrehajt.

Pack parancs: P[EPq]

A parancs hatása annyi, hogy az archívban töröltnek jelölt, de még visszahozható állományokat kiejti az új archívból. Ha már létezik az adott könyvtárnak egy backup másolata BAK kiterjesztéssel, akkor a ZOO csak abban az esetben Pack-ol, ha a P kapcsolót is melléírjuk. Az E kapcsolóval elérhetjük, hogy az eredeti archívet nem őrizzük meg BAK fájlként a Pack-olás után. A Pack-olás során egy átmeneti archív keletkezik az aktuális könyvtárban, ami megmarad, ha megszakad a Pack-olás. Ha valami okból ezt az állományt a program nem tudja átnevezni az eredeti archív nevére, amikor befejeződik a Pack-olás. A Pack parancs sok szemetet kisöpör a ZOO fájlból és használata után sok hely felszabadul.

Megjegyzések

Kezdő paranccsal

-comment (= c)

A comment paranccsal megjegyzéseket adhatunk a bepakolt állományokhoz. A program egyenként bekéri a megjegyzéseket, majd a megjegyzés zárásának módjától függően lép tovább:

- Egy <enter>-rel átléphetjük (skip) a soron lévő fájlt, és változatlanul hagyhatjuk az eredetileg ott lévő megjegyzést, ha egyáltalán volt.

- Beadhatunk 1-től 65 535 karakter hosszúságban (64 K) egy új megjegyzést, amit egy /end végjelző zár le (kisbetűt kell használni az /end kapcsolóban!).

- Beadhatunk egy end-of-fájl karaktert (többnyire <CTRL d>) a promptra, aminek hatására több fájl kommentet már nem kér be a program (átugorja a még hátralévő állományokat).

Parancs a haladóknak:

A c parancs az A kapcsoló nélkül teljes mértékben megegyezik a comment paranccsal. Az A kapcsoló hatására az archív fájlhoz, mint egészhez fűzhetünk megjegyzést. Ez a megjegyzés kilistázható, ha 1A, LA, v, illetve V parancsokkal listázunk.

A ZOO katalógus megnézése

Kezdőknek szóló -llst parancs

-list (= VC)

Kilistázza az archív összes állományát a hozzájuk fűzött valamennyi megjegyzéssel egyetemben.

Haladóknak szánt klíró parancsok

A ZOO katalógus megtekintésére a ZOO négy haladó parancsa áll rendelkezésre (1LvV), 16 választható kapcsolóval kiegészítve (aAcCdfgmqvV@/1+—).

-l (List)

Az **-l** paranccsal előhívott listák tartalmazzák minden egyes fájl kelezési adatait, eredeti és jelenlegi (tömörített) méretét, és a CF-fel jelölt kompressziós faktort. Ha fájlneveket nem adtunk meg, akkor a törlésre jelölt fájlok kivételével a ZOO mindent kiír. A generációkat illetően az alábbi logika érvényesül: ha nincs megadva egyetlen fájlnev sem, akkor csak a legfrissebb generáció állományait írja ki. Ha legalább egy fájlt megadtunk és specifikálunk egy generációt, csak a megadott generációt listázza ki. Ha egy fájlnevet csupán a generációs jellel (:, vagy ;) adunk meg, annak az állománynak minden generációját kilistázza.

-L (List)

A fenti **-l** parancshoz hasonlít, azzal a különbséggel, hogy a törölteken kívül kiírja minden megadott fájl minden generációját.

-v (Verbose list)

Az **l**-hez hasonlít, de az archívhoz hozzáfűzött megjegyzéseket is megjeleníti.

-V (Verbose list)

Az **l**-hez hasonlít, de az archívhoz és a bepakolt fájlokhoz hozzáfűzött megjegyzéseket is megjeleníti.

Mind a **v**, mind a **V** kapcsolóként is használható.

A listázásra ható kapcsolók

a

A kapcsoló hatására a ZOO egyetlen sorban mind a fájlnevet, mind a path információt, mind az archív nevét kiírja.

A

Az archívhoz hozzáfűzött valamennyi megjegyzést megjeleníti.

g

Használatakor a listázó parancs az engedélyezett maximális generációs számot is kiírja.

m

Ez a kapcsoló jelenleg az Unix és a hasonló rendszereken alkalmazható. A DOS alatt nem.

C

A CRC értéket mint négyjegyű hexadecimális számot írja ki.

l

Soronként egy fájl listázását engedi meg.

/

E kapcsoló használata azt eredményezi, hogy a névvel együtt minden könyvtári információ kilistázásra kerüljön, akkor is, ha rövidített kiíratásokkal dolgozunk.

+

Ez a kapcsoló, amely egyben alapértelmezés a List parancsok mellett, azt eredményezi, hogy a generációs számok megjelenjenek a listában.

—

A — kapcsoló letiltja a generációs számok kiíratását.

Önkinyitó SFX archívok

A ZOO program a SEZ nevű segédprogram (jelenleg 2.2 verzió) segítségével tud a ZOO fájlokból önkibontó (SFX) állományt készíteni.

Az SFX készítés általános szerkezete:

sez ZOO_fájl SFX-fájl

A ZOO fájl kiterjesztését a SEZ alapértelmezésként ZOO-nak veszi, a kimeneti (SFX) fájl pedig EXE lesz.

Az SFX fájl nevében nem adhatunk meg sem könyvtár, sem lemez nevet.

Az így elkészített SFX-et az alábbi módon lehet a legkényelmesebben kinyitni:

1. SFX az A:-ban, célkönyvtár (\KAKUKK) a C-n előre elkészítve.

2. c:

cd \kakukk

a:SFX-fájl

Ezzel ki is mertültek a ZOO-SFX lehetőségei. Emmiatt csak olyan ZOO-fájlból érdemes SFX állományt készíteni, amely nem tartalmaz az SFX számára felesleges path információkat és kommenteket, mivel ezekkel nem tud mit kezdeni.

Parancsok, kapcsolók

Parancsok kezdőknek

-add / -a	Bővít.
-freshen/ -f	Frissít.
-update / -u	Aktualizál = frissít + bővít.
-move / -m	Áthelyez = bővít, majd törli a forrást.
-extract/ -e	Kipakol.
-test / -t	Ellenőriz.
-print / -p	Kiír/kinyomtat.
-delete / -d	Töröl.
-comment/ -c	Megjegyzéseket fűz hozzá.
-list / -l	ZOO katalógust kiír.

Haladóknak szánt parancsok

a (add)	Bővít.
c (comment)	Megjegyzéseket fűz hozzá.
D (delete)	Archívból töröl.
e (extract)	Kipakol.
x (extract)	Kipakol.
g (generation)	Generációk.
l (list)	ZOO katalógust kilistáz.

L (list)	ZOO katalógust kilistáz.
v (verbose list)	ZOO katalógust kilistáz.
V (verbose list)	ZOO katalógust kilistáz.
P (pack)	Pack-ol.
T (timestamp)	Dátumot igazít.
u (freshen)	Frissít.
U (undelete)	Visszahoz törölt fájlokat.
f (filter)	Szűrést végez.
h (Help)	Helpet ad.

Kapcsolók

Az egyes parancsok mellett már említett kapcsolókon kívül van még néhány általánosan használható:

c (Comment)

Az a és u parancsok mellett használva a program minden egyes bepakolt állományhoz bekér egy-egy megjegyzést, mint a comment parancs. Ha az adott állomány előző verziója megjegyzésekkel ellátva már bent volt az archívban, akkor a program felkínálja az előző kommentet. Azt vagy elfogadjuk, vagy megváltoztatjuk.

Az l (list) paranccsal kombinálva kijelzésre kerül a ZOO katalógussal együtt a kijelzett fájlokhoz tartozó megjegyzés is.

d (Deleted too)

Az archívval dolgozó legtöbb parancs csak azokkal a fájlokkal foglalkozik, amelyeket nem töröltünk. A d kapcsoló hatására a ZOO parancsai a törlésre kijelölt állományokat is látják. Ha megduplázzuk (dd), akkor pedig csak a törölt állományokkal dolgoznak a ZOO parancsok.

. (relative)

A / és // kapcsolókkal együtt használva a pont (.) kapcsolót, azt kapjuk, hogy a / jellel kezdődő kibontott útnév az aktuális könyvtárból fog indulni, ezáltal az aktuális könyvtárban jöhet létre alkönyvtárként a célkönyvtár.

A P (Pack) paranccsal együtt használva a Pack-olt archív az aktuális könyvtárban jön létre. Ezt azért csinálták, hogy lehetővé tegyék korlátozott lemezkapacitással, de több meghajtóval rendelkező felhasználók számára is nagy archívok Pack-olását.

f (Fast)

Ha az a, vagy az u parancs mellett használjuk az f kapcsolót, akkor a ZOO tömörítés nélkül pakol. Ha az l parancs mellett alkalmazzuk, akkor sűrített, többszlopos ZOO katalógus kiírást kapunk.

q (Quiet)

Alaphelyzetben a ZOO mindig kiírja, hogy éppen mivel mit csinál. A q kapcsoló ezt a kiírást tiltja le. Kipakolásakor például csak a fájlnevet jelenik meg. A ZOO katalógus listázáskor is letiltja a be- és a levezető szövegek kiíratását. Az lf paranccsal kombinálva egyszerűen csak az archívban lévő fájlnevek felsorolását eredményezi.

Ha megduplázzuk (qq), akkor elnyomja a figyelmeztető (warning) üzeneteket. Ha megtriplázzuk (qqq), akkor a hibaüzeneteket (error) is elnyomja. A végzetes hiba (fatal error) jellegű üzeneteket nem tudja elnyomni.

Speciális szolgáltatások

T (Timestamp)

A fájl keletkezését és időpontját a benne lévő legfrissebb bepakolt állományhoz igazítja. A q (Quiet) kapcsolót szokták csak vele használni.

f (Filter)

A ZOO működhet egy egyszerű tömörítő-kibontó filterként, amikor a standard bemenetről olvas és a standard kimenetre ír. Az fc a tömörítő, az fu a kibontó funkciót jelenti. A CRC értékekkel vizsgálja az adatok épségét. A tömörített adatsornak nincs belső archív szerkezete, és csak akkor tartalmaz több állományt, ha a bemeneti adatár már strukturált volt.

Generációk használata a ZOO programmal

A tömörítő programok nagy családjában jelenleg a ZOO az egyetlen olyan rendszer, amely lehetővé teszi, hogy a folyamatosan vagy időszakosan változó tartalmú állományokat azonos néven egyszerre több változatban is megőrizzen és kezeljen. Ezeket a változatokat a ZOO szóhasználatát átvéve mi is nevezhetjük generációknak. A program lehetővé teszi, hogy a felhasználó határozza meg, igényli-e ezt a szolgáltatást, s ha igen, akkor a bepakolt állományok közül melyik generációval akar foglalkozni és szolgáltatásait hogyan kívánja igénybe venni. A generációk kezelésére szolgáló parancs kiegészíthető további kapcsolókkal, melyek igen nagy rugalmasságot biztosítanak a program használata során.

Az alapelv a következő: az egy bepakolt ZOO fájlban egyidejűleg tartható fájlgenerációk számát a fájlgenerációs limit, vagy az archívgenerációs limit szabja meg. A kettő közül mindig a kisebbik érték. Speciális határesetként a 0 generációs limit a korlátok megszűnését jelenti. Így, ha a fájlgenerációs limit 0 és az archívgenerációs limit = 1, akkor egy újonnan létrehozott archívban induláskor minden fájl csak egy generációja lehet. A generációs limit értéke 0 és 15 között lehet. Ennél nagyobb érték megadása esetén a program modulo 16-ot számol (addig von le belőle 16-ot, amíg 16-nál kisebb értéket kap, s végül ezt helyettesíti be).

A fájlgenerációs limitnek a `gl=<szám>` paranccsal adhatunk egy konkrét értéket. A `gl+<szám>` parancs a megadott számmal növeli, a `gl-<szám>` parancs pedig a megadott számmal csökkenti a fájlgenerációs limitet.

A parancs másik alakja (`gc=<szám>`, `gc+<szám>`, `gc-<szám>`) a megadott fájl generációs számát állítja be. Amikor először kerül be a ZOO fájlba, minden fájl generációs száma 1 lesz. Minden egyes alkalommal, amikor ugyanazon a néven kerül egy fájl az archívba, eggyel nagyobb generációs számot kap.

Ha már túl sok generáció gyűlt össze és a ZOO fájl szinte kezelhetetlenül hatalmasra duzzadt, akkor a megfelelő `gc` parancs segítségével alapos fogyókúrára fogható. A `gc` parancs szintaxisa megegyezik a `gl` parancsével, kivéve, hogy az `A` kapcsoló nem alkalmazható a `gc` mellett.

A parancs harmadik alakja, a gA– lekapcsolja a generációs lehetőségek használatát. Új ZOO fájl létrehozásakor ez az alapértelmezés, a gA+ paranccsal azonban bekapcsolhatjuk. Ha a generációzás ki van kapcsolva, akkor a ZOO nem jelzi ki a generációs számokat a ZOO katalógus kiíratásakor és nem tárol el több generációt. A gA+ és gA– parancsok váltogatásával óvatosan bánjunk, mert az így kezelt ZOO fájl viselkedése néha nem könnyen érthető meg.

Adatmentés sérült ZOO fájlkból

A @m[,n] kapcsolóval megadhatjuk a ZOO-nak, hogy a ZOO fájlban hol van az a rész, ahonnan ki akarunk pakolni. A @ jel utáni első szám a pozíció, majd szükség esetén egy vessző után a második szám az esetleges rövidülés miatti kompenzáció. A beírandó egész számokat a csomag részeként szállított FIZ program segítségével kaphatjuk meg.

A ZOO katalógus kiírása:

zoo l@m,n ZOO fájl

ahol m és n a FIZ-zel kapott értékek.

Kibontás:

zoo x@m,n ZOO fájl

A ZOO a gyakorlatban

A ZOO dzsóker karakterei

A ZOO a többi tömörítőtől egy kicsit eltérően kezeli a dzsóker (* és ?) karaktereket s emellett még néhány más karaktert is. Ez arra vezethető vissza, hogy a programot először nem a DOS-ra tervezték. A ZOO csak listázáskor vagy kibontáskor fogad el a könyvtárnevekben dzsóker karaktereket, ZOO fájlba beíró parancsokkal nem.

A ZOO a következő dzsóker karaktereket értelmezi: * ? / — : ;

Az egyes ZOO dzsóker karakterek szerepe, hatása:

*

Bármely karaktert helyettesít és bármennyit. Pl.: ka*.xxx megfelel a ka.xxx, kalap.xxx névnek. Mint a DOS-ban is. A k*r.xxx megfelel a kar.xxx, a kever.xxx, kr.xxx neveknek, a *z megfelel az összes z-vel végződő névnek. Ez utóbbi használat már eltér a DOS-nál megszokottól.

?

Megegyezik a DOS-ban ismert használattal.

/

Ha egy megadott szerkezet bármely részében tartalmaz egy / jelet, akkor a jel utáni path-nak megfelelő útvonalon keresi csak a megadott állományt.

c-c

Karakter tartomány megadható. Akkor használható, ha a fájlnev első betűje alapján választhatunk. Pl. az a-val, b-vel és c-vel kezdődő fájlokat az alábbi szerkezet szűri ki: a-c*.*. Ebből következik a jótanács, hogy ha lehet, ne használjunk fájlnevekben – karaktereket, ha ZOO-val is dolgozunk.

: és ;

A fájlnev mögött kell közvetlenül elhelyezni, ha megadott generációkkal akarunk foglalkozni. Ha a :, illetve ; generációs jel mögött nem adunk meg számot, akkor a ZOO az adott fájl összes generációjával foglalkozik. Különleges eset a 0 szám, amikor a fájl legutolsó generációja lesz a ZOO célja, míg a <CTRL 0> az utolsó kivételével mindegyiket kijelöli. Ha a generációs karakter elé nem írunk fájlnevet, akkor az összes fájl megadott generációival foglalkozik. A magában álló generációs jel az összes fájl összes generációját jelöli.

Végül meg kell jegyezni:

- a * és ? karakterek szabadon kombinálhatók.
- a * karakter az összes fájlnevet kiválasztja.
- a *.* a pontot tartalmazó neveket választja csak ki.

HYPER — a mélynövésű óriás

A HYPER nevű tömörítő program két német programozó, Peter Sawatzki és Klaus Peter Nischke alkotása. A program szinte csak a standard, szabványos tömörítő funkciókat tudja, de azokat egyszerűen és gyorsan. A HYPER25.EXE az alábbi állományokat tartalmazza:

HYPER.EXE	A tömörítő program.
HYPER.REF	A program ismertetője.
HYPERTV.PAS	Példaprogram HYP fájlokhoz.

Bepakolás

A HYPER természetesen legyen az aktuális könyvtárban vagy path-on. A szintaxis:

HYPER [parancsok] archív [fájl maszk/ok]

A HYPER parancsait mínusz (–) jellel kell bevezetni. A parancsok a parancssorban bárhol lehetnek, nincs kötelező helyük és egy közös bevezető – jel mögött kombinálhatók is. Ha nem adunk meg parancsot, akkor az alapértelmezés –x, azaz kikapolás.

Az archív kiterjesztése alapértelmezésben HYP, de ha egy HYPER SFX-et akarunk kezelni, semmi akadálya, csupán EXE kiterjesztéssel kell az archív nevét ellátni. A HYP fájl nevében is használhatunk dzsóker karaktereket.

A fájlmaszkok tartalmazhatnak dzsóker karaktereket és path információt is. Ha nem adunk meg fájlmaszkot, akkor az alapértelmezés *.* lesz.

A bepakoló parancsok

- a = Add. Sima hozzáadást végez.
- m = Move. A megadott állományokat bepakolja, majd törli a forrásállományokat.
- u = Update. Csak újat, vagy frissebbet pakol be.
- f = Freshen. Nem visz be új állományt, csupán frissíti a meglévőket, ha a bepakolandó állomány a frissebb.
- r = Recurse subdirectories. Az adott könyvtár alkönyvtáraiból is bepakol.
- p = store Path information specified and recursed into. A megadott path-t is megőrzi, illetve az abból nyíló könyvtárakból bepakolt állományok path információit is.

Kipakolás

Szintaxisa:

HYPER [parancsok] archív [könyvtár] [fájlok] ...[>kimenet]

A parancsok alapértelmezése –x. Ha nem adunk meg célkönyvtárat, akkor az aktuális könyvtárban bontja ki az állományokat. Ha nem adunk meg fájlneveket, akkor mindent kinyit.

A kibontó parancsok:

- x = eXtract files. Kibont. Ez az alapértelmezés, ha nem adunk ki parancsot.

-p = restore Path information. Visszaállítja az eredeti könyvtárszerkezetet, az **-x** paranccsal együtt használva minden a helyére kerül vissza.

-o = Overwrite. Kérdezés nélkül felülírja a célkönyvtárban található azonos nevű állományokat a kibontott fájlokkal.

Archív katalógus a HYPER-rel

Szintaxis: HYPER **-v** archív [fájlok]

-v = View

Mind az archív nevében, mind a fájlok neveiben megengedett a dzsóker karakterek (? , *) használata. Ha nem adunk fájlnévet, mindent kiír.

Törlés az archívból

Szintaxis:

HYPER **-d** archív [fájlok]

-d = Delete

A fájlnevekben a dzsóker karakterek megengedettek. Ha nem adunk meg törlendő nevet a **-d** paranccsal, akkor mindent töröl az archívból.

HYPER-SFX készítése, használata

Az SFX készítése elég primitív. Az alábbi szerkezettel kell dolgozni:

```
copy /b HYPER.EXE+HYPfájl.HYP SFX-fájl.EXE
```

Ahol

— HYPER.EXE maga a tömörítő.

— HYPfájl.HYP a HYPER-rel előzőleg elkészített archív.

— SFX-fájl.EXE az önkinyitó EXE állomány.

Az így elkészített SFX mintegy 16 K-val nagyobb a HYP fájlnál.

A HYPER-SFX kétféleképp kezelhető:

— Külső HYPER programmal.

— Belső HYPER programmal.

Pl. egy HYP-SFX.EXE bővítése egy KAKUKK.DOC-kal az alábbi módokon lehetséges:

```
HYPER -a HYP-SFX.EXE KAKUKK.DOC
```

vagy

```
HYP-SFX -a KAKUKK.DOC
```

A többi parancsra ugyanez érvényes.

EXE és COM fájlok tömörítői

Ezek a programok csak a végrehajtható EXE és COM állományokat tömörítik, a hagyományos tömörítőkhöz hasonló hatásfokkal. Fő előnye ennek a módszernek, hogy az így zsugorított állományok közvetlenül futtathatók maradnak, nem kell kibontásukhoz segédprogram. Sajnos nem minden EXE és COM állomány viseli el az ilyen zsugorítást, ezért az eredeti állományokat csak akkor töröljük, ha alaposan meggyőződünk arról, hogy a zsugorított fájl hibátlanul, minden funkcióját megőrizve működik.

Ebbe a csoportba két nevesebb programcsomag tartozik:

PKLITE (1.03, PKWare)

LZEXE (0.91, Vernon Bueg)

Akkor célszerű alkalmazásuk, ha a választott programokat floppyról vagy lassú winchesterről akarjuk futtatni. Ilyenkor ugyanis a beolvasás jelentősen gyorsul, a kibontással járó lassulás pedig többnyire elhanyagolható.

A két vizsgált rendszer közül a PKLITE kezelése egyszerű, és mind a COM, mind az EXE fájlokat tömöríti. Nem lehet azonban vele túl apró vagy belső overlay technikát alkalmazó programokat összenyomni. (Ezek nagy részét maga a tömörítő jelzi.) Előnye még, hogy a tömörített programok eredeti állapota szinte mindig visszaállítható, ha nem tartozik a fenti kategóriába. A kifejezetten a 386-os processzorra készült EXE állományok jó része szintén nem zsugorítható a PKLITE-tal.

Az LZEXE csomag is jól használható, de kissé nehézkessé teszi használatát, hogy a COM fájlokat először EXE-vé kell átalakítani.

Egy kellemes meglepetés — PKLITE

Phil Katz csapatának, a PKWare cégnek gyöngyszeme a PKLITE. Eddigi tömörítőik után ez a kis program is igen kellemes mindazoknak, akik lemezén helyproblémák vannak. A program átlagban 30-35 százalékkal tudja csökkenteni a lemezen a végrehajtható (EXE és COM) állományok helyigényét, anélkül, hogy a futásidőt észrevehetően növelné.

A PKLITE a PKZIP-hez hasonlóan tömörít, de hozzáad egy kis fejléct a program elejéhez, ami indításkor kibontja azt. Ha elindítunk egy PKLITE-tal tömörített programot, akkor a program automatikusan kinyílik a memóriában és futni kezd. A tömörítési eljárás egyáltalán nem változtatja meg a program működését, csupán a tároláshoz szükséges helyet csökkenti. A legtöbb PKLITE-tal tömörített programhoz nem kell plusz memória. A PKLITE lehetővé teszi a tömörített fájlok eredeti méretre való visszaállítását is.

Hardver igénye alig van, bármilyen IBM PC-n fut, DOS 2.0, vagy magasabb verziók alatt. Magyarországon a legtöbben a 3.30-as DOS-t használják, és az alacsony DOS verzió egyre ritkábban okoz gondot. A program MS-DOS, PC-DOS, Tandon-DOS és Compaq-DOS alatt egyaránt futtatható. A tapasztalat szerint az OS/2-t nem szereti, viszont az Unix DOS ablakában fut.

Egy PKLITE-tal tömörített program futtatása maximum 4 K többletmemóriát igényel a memóriába való kinyitáshoz. Futtatáskor a program már ugyanannyi memóriához fér hozzá, mint tömörítés nélkül.

A plusz szolgáltatásokat is tartalmazó regisztrált változatot Budapesten is meg lehet rendelni. (Például a Cédrus Floppyland boltjában.) A regisztrált felhasználó megkapja a PKLITE jelenlegi verzióját tartalmazó lemezt, a program nyomtatott kézikönyvét, valamint a PKLITE következő verzióját, mielőst az megjelenik.

Számtalan előnye mellett azt is érdemes megjegyezni, hogy mikor nem érdemes a PKLITE programmal próbálkozni.

Nem használhatjuk az alábbi esetekben:

- MS-Windows alá írt programoknál.
- Másolásvédett programoknál.
- Belső overlay-t alkalmazó programoknál (FoxPro, MS CodeView, MS-C2, FSD, Turbo Debug, stb.).
- OS/2-re és 386-os processzorra tervezett programoknál.
- Ha előzőleg már más EXE-COM tömörítővel tömörítettük a forrásállományt.

A PKLITE használata

A PKLITE parancsszerkezete az alábbi leírást követi:

```
PKLITE [kapcsoló] [path]bemenő_fájl [path][Kimenő_fájl]
```

ahol:

PKLITE = A PKWare tömörítő programja.

kapcsolók

- a Tömöríti a belső overlay-s fájlokat is.
- b Az eredetiből .BAK fájlt csinál.
- e Extra kompressziós módszer (csak a regisztrált verzióban).
- l Software license képernyőt hozza elő.
- n Nem tömörít belső overlay-es állományokat.
- o Ha létezik output fájl azonos névvel, felülírja.
- r Eltávolít minden extra EXE adatot.
- u Az időt a rendszerórához igazítja.
- x Kibont egy tömörített fájlt.

[path]bemenő_fájl

A tömörítendő (vagy kibontandó) végrehajtható fájl neve és elérési útja, másnéven a forrásállomány.

[path][Kimenő_fájl]

A cél fájl neve és elérési útja. Így átnevezhető és/vagy áthelyezhető a tömörített, vagy kibontott fájl.

Megjegyzés: a [és] szöveges zárójelek között szereplő paraméterek nem kötelezőek.

Néhány fejből tartandó fontos dolog a PKLITE használatával kapcsolatban:

- A program nem tesz különbséget a kis- és nagybetűk használata között.
- A kapcsolók előtt bevezető – vagy / jelet kell tenni.
- A kapcsolók kombinálhatóak (pl.: „-o -b” vagy „-ob”).
- Ha a forrásfájlnak nem adtunk kiterjesztést, a PKLITE automatikusan egy EXE, vagy COM fájlt keres az adott névvel. Ha mind a kettőből talál, mindkettőt összenyomja.
- Ha nem adtunk meg kimeneti fájlt a parancssorban, akkor a tömörített fájl ugyanazt a nevet kapja, mint a forrásfájl, és ugyanoda kerül, azaz felülírja a forrásfájlt.
- Ha a PKLITE-nek semmiféle paramétert nem adunk, akkor egy Help képernyőt ír ki, amelyben felsorolja a program kapcsolóit.
- Végül még egy hasznos tanács. A PKLITE programot soha ne használjuk az eredeti lemezekről, hanem mindig csak másolatról. Az eredeti lemezeket őrizzük meg biztonságosan.

A kapcsolókról

-a (Always compress)

Ezzel a kapcsolóval arra utasítjuk a PKLITE programot, hogy akkor is tömörítse a kijelölt fájlokat, ha azok belső overlay-t használnak. Azok a programok, amelyek EXE fájlokból betöltött overlay-eket használnak, tömörített formában nem mindig futnak hibátlanul. A PKLITE észreveszi, ha egy programfájl belső overlay-t tartalmaz. Amikor nem használjuk az -a kapcsolót, minden ilyen esetben visszakérdez, hogy tömörítheti-e.

C:\>PKLITE -a *.exe

A fenti példában a PKLITE minden EXE fájlt tömöríteni fog a gyökérvetvénytárban, tekintet nélkül arra, hogy alkalmaznak-e overlay-t. A program nem kérdez vissza, és a tömörített állományok felül fogják írni az eredeti állományokat, mivel nem adtunk meg

kimeneti könyvtárat és fájlneveket. Az overlay adatokat nem tömöríti a PKLITE, azonban az overlay előtt elhelyezkedő képbetöltő (load image) adatokat igen. Némely program igényli, hogy az EXE fájlokban lévő overlay adatok az EXE fájl egy megadott helyén legyenek. Ezek a programok nem fognak megfelelően futni, ha PKLITE-tal tömörítjük őket. Azok a programok amelyek így nem működnek rendesen, többnyire visszaalakíthatók eredeti állapotukba az `-x` kapcsolóval.

-b (= Backup)

Az eredeti fájl változatlan névvel, de BAK kiterjesztéssel megmarad ugyanabban a könyvtárban. Ha kimeneti fájlt is megadunk, akkor a PKLITE a `-b` kapcsolót figyelmen kívül hagyja.

-e (= Extra compression)

Csak a Professional változatban szerepel. A legtömörebb, legkisebb végrehajtható fájlokat ez a kapcsoló biztosítja, egyben az így előállított program van leginkább védve az illetéktelen kibontással szemben. Az `-e` kapcsolóval tömörített fájlok nem alakíthatók vissza eredeti méretükre a standard PKLITE `-x` opciójával. Programfejlesztőknek kifejezetten előnyös. Megéri a regisztrálási díjat.

-l (= Licence)

A PKLITE licenc képernyőjét hozza elő.

-n (= No compress)

Nem tömörít belső overlay-t használó programokat. Arra utasítja a PKLITE-ot, hogy ugorja át kérdezés nélkül az overlay-t alkalmazó programokat.

-o (= Overwrite)

A kimeneti fájl felülírja a célkönyvtár esetleg már létező azonos nevű állományát. Alaphelyzetben a PKLITE rákérdez, ha a kimeneti állománnyal azonos nevű állomány már van az adott célkönyvtárban. Az `-o` kapcsoló ezt a rákérdezést iktatja ki. Csak akkor szükséges, ha megadunk kimeneti fájlt.

-r (= Remove extra data)

Eltávolítja az extra EXE adatokat. Ez a kapcsoló ugyan tömöríti a fájlokat, de kiejti a load image adatok után fűzött adatokat. (Setup információkat, overlay adatokat, debug információkat.) Legyünk óvatosak vele, mert az így kiejtett adatokat az `-x` kapcsolóval sem lehet visszahozni.

-u (= Update)

A tömörített fájl dátumát aktualizálja. Alapértelmezésként a PKLITE nem bántja a fájl dátumát. Akkor kell az `-u` kapcsolót használnunk, ha azt akarjuk, hogy az új fájl friss keletkezési dátumot kapjon.

-x (= eXtract)

Az előzőleg PKLITE programmal zsugorított EXE és COM fájlok visszaalakítására szolgál. Nem minden esetben bontható ki épségben az `-e` paranccsal és gyakran az `-r` paranccsal tömörített program sem. Együtt használható a `-b`, az `-o` és az `-u` kapcsolókkal, de ez esetben néhány fontos dolgot meg kell jegyeznünk:

— Az `-x` és a `-b` együttes alkalmazásakor a kiterjesztett futtatható állomány lesz a helyreállított, a BAK állomány pedig a tömörített változat.

— Az `-x` és az `-o` együttes alkalmazása különösen veszélyes, mert ha a kibontás sikertelen, akkor elvész a program.

— Az `-x` és az `-u` együtt a kipakolás dátumára aktualizál, mivel a PKLITE a fájl eredeti dátumozását nem tárolja el az állományban.

Gyakorlati példák

1. Az alábbi példában az ATTRIB.EXE fájl össze lesz tömörítve, és a tömörített változat felülírja az eredetit:

```
C:\DOS>PKLITE attrib.exe
```

Ha kiadjuk a fenti parancsot, a PKLITE az alábbi információt írja ki a képernyőre:

```
PKLITE (tm) Executable File Compressor. Version 1.00
```

```
Copyright 1990 PKWARE Inc. All Rights Reserved.
```

```
Compressing attrib.exe
```

```
Original Size: 10656 Compressed Size: 6790 Ratio: 36.3
```

2. Miután egy fájlt tömörítettünk a PKLITE-tal, az kiír néhány értékes információt. A PKLITE felsorolja az eredeti fájl nevét, méretét, a tömörített méretet és a tömörítési arányt. A kimeneti fájl és annak helye szintén kiírásra kerül, ha az más mint az eredeti. A következő példa azt mutatja be, mi történik akkor, ha a kimeneti fájl más néven és más útvonalon van, mint a forrásfájl:

```
C:\> PKLITE light.exe A:smlight.exe
```

```
PKLITE (tm) Executable File Compressor. Version 1.00
```

```
Copyright 1990 PKWARE Inc. All Rights Reserved.
```

```
Compressing: LIGHT.EXE into file A:SMLIGHT.EXE
```

```
Original Size: 18466 Compressed Size: 12299 Ratio: 33.4
```

Hibaüzenetek

Syntax error

A parancssorban elírtunk valamit. A hibajelzés után a PKLITE kiírja saját Help képernyőjét is. (errorlevel 1)

File extension must be EXE or COM

A megadott fájl nem EXE vagy COM kiterjesztésű. Nevezzük át EXE vagy COM kiterjesztésűre a tömörítés előtt. (errorlevel 2)

Cannot open input file

A tömöríteni kívánt állományt a PKLITE nem tudta megnyitni. Lemezhiba, géphiba, elírt név vagy rossz path miatt, esetleg nem is létezik az adott állomány, másik program blokkolta a hálózaton, ráült egy rezidens vagy párhuzamosan futó program. (errorlevel 3)

Could not open output file

A PKLITE nem tudta létrehozni a kimeneti állományt. Vagy lemezhiba történt, vagy a fájlt leblokkolta egy másik program. Számos ok elképzelhető. (errorlevel 4)

Write error

A program tulajdonképpen már megcsinálta az állományt, de nem sikerült kiírnia. Lemezhiba illetve multitaszk vagy hálózati üzemmódban foglalt állomány lehet az oka. (errorlevel 5)

Disk full error

Teljesen megtelt a céllemez. Takarítsuk le róla a felesleget! (errorlevel 6)

Read error

A PKLITE nem tudja beolvasni a megadott forrásállományt. Nagy valószínűséggel sérült a kijelölt program. (errorlevel 7)

Create error

A célállományt nem sikerült létrehozni. Ellenőrizzük a meghajtót és a megadott útvonalat, valamint a céllemezen a rendelkezésre álló szabad területet. (errorlevel 8)

Memory error

Teli van a gép memóriája mindenféle maradvéccel. Legalább 85 K szabad hely kell a RAM-ban a program sikeres lefuttatásához. (errorlevel 9)

Cannot compress file into itself

A tömörített állományt nem tehetjük önmagára. Rosszul írtuk be a parancssort. A „PKLITE kakukk.exe kakukk” sor helyett elég a „PKLITE kakukk.exe” is. (errorlevel 10)

EXE header error

Az EXE állományok fejrészéből igen sok információt olvas ki a PKLITE a tömörítéshez. Ha viszont ez sérült vagy nem szabványos, akkor a program nem tud vele mit kezdeni. Néha akkor is ezt az üzenetet kapjuk, ha a memória kevés. Ellenőrzésére nagyobb RAM területet felszabadítva ismételjük meg a fenti parancssort. (errorlevel 11)

No extract code error

Az -x kapcsoló alkalmazásakor jöhet elő ez a hibaüzenet. Oka lehet, hogy egy nem tömörített, más programmal tömörített, vagy a PKLITE Professional verzió -e kapcsolójával tömörített állományt akartunk kibontani eredeti méretére. (errorlevel 12)

Data error

Közönséges adathiba. Az állomány használhatatlan. (errorlevel 13)

Compressing many files into one file error

Ejnye, nem illik több állományt egybe tömöríteni: „PKLITE *.EXE akarmi.EXE” Tessék szépen kijavítani! (errorlevel 14)

Output wildcards error

A kimeneti állományba illegális dzsóker karaktereket tettünk, valószínűleg tévedésből. Javítsuk ki a kimeneti fájl magadását. (errorlevel 15)

A PKLITE segédprogramjai

CHK4LITE.EXE

A PKLITE 1.03 verzióját kíséri a CHK4LITE.EXE program, amellyel azt ellenőrizhetjük, hogy a tömörítendő programokat tömörítették-e már előzőleg a PKLITE programmal. A program az errorlevel változó értékét az alábbiak szerint állítja be:

0 = Nem tömörítették.

1 = A tömörítendő programok közül legalább egyet már összenyomtak a PKLITE segítségével.

2 = A PKLITE Professional —e kapcsolójával tömörítettek legalább egyet a megadott állományok közül.

HDROPT.EXE

Előzőleg LZEXE programmal tömörített, majd az UNLZEXE programmal visszaállított állományokat nyersen a PKLITE nem tudja fogyasztani, elrontja a gyomrát az EXE headertől. Ezen segít a HDROPT.EXE azzal, hogy visszaállítja a szabványos EXE headert.

Az ezerarcú LZEXE

Az LZEXE 0.91 csomag egy francia programozó alkotása. Arra szolgál, hogy az EXE és COM állományokat kisebb helyen tárolhassuk, kibontó segédprogram igénybevétele nélkül. A csomag a következő programokat tartalmazza:

COMTOEXE.EXE

A COM állományok EXE állományokká alakítására, mivel a csomag csak az EXE fájlokat tudja tömöríteni. Használata:

```
COMTOEXE fájlnev[.com] [EXE-fájlnev[.EXE]]
```

UNLZEXE.EXE

A tömörítés előtti állapot visszaállítására szolgál. Csak EXE fájlok visszaállítására képes.

UPACKEXE.EXE

A Microsoft EXEPACK programmal tömörített állományok további zsugorításához először ezzel a programmal kell kinyitni az adott állományokat, az LZEXE csak utána tud dolgozni megfelelően.

INFOEXE.EXE

Francia nyelvű szöveget ír ki a program készítőjéről.

LZ.EXE

Az egyik tényleges tömörítő. A régi EXE állományokat OLD kiterjesztéssel megőrzi, meg kell adni valamilyen fájlnevet (dzsóker karakterekkel is lehet operálni, ha több állományt egyszerre akarunk zsugorítani).

```
LZ [path]<fájl>[.exe]
```

LZEXE.EXE

A másik fájl-tömörítő. Egy adott állományt tömörít.

```
LZEXE fájlnev[.EXE]
```

LZESHELL.EXE

A leírás szerint összefogja a COMTOEXE és az LZEXE működését.

Az LZEXE sem alkalmas minden programállomány tömörítésére. Nem tudja a belső overlay-vel működő programokat tömöríteni, valamint azokat az EXE állományokat sem, amelyeket pl. egyszerű átnevezéssel készítenek COM fájlokból, mert a fájl elejéről hiányzik a szabványos MZ.

A besorolhatatlan KK

Masa István programcsomagja átmenet a fájl-tömörítők és a winchester-duplázók között. Lehetővé teszi, hogy a bepakolt állományokat szekvenciálisan kibontva újra olvassuk. Igen jól használható olyan szövegek archiválásához, amelyeket nem akarunk módosítani, csupán szeretnénk olvasni, nyomtatni, más programokba átküldeni vagy kimásolni. Lehetséges alkalmazási területei a szövegeket szekvenciálisan csak egy irányban olvasó help rendszerek és programismertető, valamint egyes hipertext rendszerek.

A szövegállományokat a csomag KK.EXE programjával lehet összezsugorítani. Az így tömörített állományokat az Alaplap mágneslemez melléklete révén ismertté vált rezidens KKINS program segítségével mindenféle paraméterezés nélkül nyitva látjuk, és amíg a KKUNINS programmal be nem csukjuk, addig tetszőleges programmal tudjuk azokat olvasni, másolni, stb. A végleges kibontás kétféle is lehet: a KKK programmal, vagy úgy, hogy a KKINS meghívása után és a KKUNINS használata előtt a programot más néven egy tetszőleges könyvtárba átmásoljuk.

A KK és családja szinte primitíven egyszerűen használható s mivel hihetetlenül apró, nagyon kevés helyet foglal el. Hasznossá kb. 1 K méret fölött kezd válni, ennél kisebb szövegek esetén a KK alkalmazása gyakran még méretnövekedéssel is jár.

A KKINS programot mindig a főprogramból (pl. Norton Commander) indítsuk, és az abból való végleges kilépés előtt a KKUNINS programmal zárjuk le. A belépés és a kilépés között a KK-val összenyomott állományokat a különféle sorosan olvasó programok (mint a DOS COPY parancsa is) kiterítve látják.

Fújd fel a winchestered!

A PC felhasználók visszatérő gondja, hogy teli van a... winchesterük is. Különböző backup és fájl-tömörítő programokkal készített archív lemezeink gyakran elkeverednek, megsérülnek. Tömörített állományainkat is a winchesteren tárolva, kibontáskor duplán foglalnak helyet, ha egyáltalán marad elég szabad hely a kibontáshoz.

Egyre gyakrabban jelennek meg viszont a piacon olyan hardveres kiegészítők, amelyek célja a winchesterek kapacitásának bővítése. Ezek egy része csak vezérlőkártya, amely az adatokat nagyobb írássűrűséggel viszi fel a lemezekre, de az állományok szerkezetével nem foglalkozik. Az ilyen bővítők a kapacitás kedvéért a sokkal fontosabb adatbiztonságot áldozzák fel, ezért nagy karrierre nem számíthatnak.

A másik bővítési irányzatot azok a rendszerek jelentik, amelyek a már évek óta bevált és biztonságosan alkalmazott tömörítő eljárásokat integrálják egy rezidensen működő winchestervezérlő programba. Ezek a kiírandó állományok szerkezetének elemzése alapján a megfelelő tömörítő eljárással először összenyomják a lemezre szánt állományt és az így zsugorított adathalmaz kerül a lemezre. Visszafelé mindez úgy történik, hogy a beolvasandó adatállomány vagy programfájl először kinyílik, és azt a behívó program nyitva, tömörítetlenül látja.

Ennek az eljárásnak vannak elsősorban hardveres, vegyesen hardveres és szoftveres, valamint tisztán szoftveres változatai is. A tárgyalt programok közül a NewSpace a tisztán szoftveres megoldás egyik példája. A Stacker szoftveres verziója szintén ide tartozik, a koprocesszor-kártyás verzió pedig a vegyes rendszerekhez. A harmadik típus képviselője az Expanz! Plus, amely elsősorban hardveres megoldással kínál valós idejű (real time) adattömörítést.

Az egyes programok ismertetése előtt azonban meg kell jegyezni, hogy a hazai árviszonyok és a szoftverek megszokott terjedési csatornái mellett elsősorban a NewSpace és a Stacker elterjedésére lehet számítani. Ennek megfelelően a könyvben jóval több szó esik az említett két programról, mint az Expanz! Plus rendszerről.

STACKER, te csalfa!

Mit tud a program?

1. Átlagban megduplázza a winchester kapacitását, anélkül, hogy növelné a lemezreírás fizikai sűrűségét. Működésének alapelve, hogy a Stacker használatába bevont meghajtókra mindig tömörítve ír, és onnan kicsomagolva olvas, az AT-gépeken szinte észrevehető lassulás nélkül. (A lassú XT-gépekre sajnos ez nem áll, ezeknél a lassulás miatt célszerűbb más megoldást keresni.)

2. A tömörítés határfoka eléri, sőt néha meghaladja a PKZIP-ét, viszont az installálás után nem kell paramétereznünk.

3. Nem kell időt és energiát rabló ki- és becsomagolással vesződni, mivel a Stackeren keresztül a tömörített állományokat mindig kibontva látjuk.

4. A floppy-ra nem tömörít, tehát régi lemezeinket bátran használhatjuk és a Stacker nélküli gépekkel zavartalanul tudunk adatot vagy programot cserélni.

5. Pofonegyszerű az installálása, igen jó a kézikönyv, rengeteg hasznos tanáccsal a különböző gépekhez. Az installálás után többet nem kell paraméterezgetnünk, mindent automatikusan a háttérből intéz.

6. A lemez nem másolásvédett, ezért hardverhiba, DOS-csere, vírusfertőzés vagy más ok miatt szükségessé váló rendszer-újraépítés akárhányszor elvégezhető.

7. A nagyméretű winchestereket a 3.30-as DOS alatt 16 megabájtonként tudja felfűjni, így több partícióval kell számolni. Megfelelő DOS (Tandon, Compaq, DR DOS illetve az MS-DOS 4.00 felett) használata esetén tetszőleges méretű partíciókat (250 megáig) alakíthatunk ki az egybefüggő winchesteren. A 3.30-as DOS használatakor a Stacker a 16 MB-nál nagyobb partíciókat kétfelé osztja, a 16 MB-os részt és a maradékot két menetben tudja megduplázni.

8. Ha az installáláskor a SWAP opciót kértük, akkor nem kell átírnunk batch állományainkat, a Stacker udvariasan megcseréli a meghajtó-neveket.

9. Rezidensen csak 30 K a helyigénye, de partíciónként további 3 K-ra van szüksége. Ez megfelelő memóriakezelő programmal áttehető a 640 K feletti részbe, vagy az EMS-területre.

10. Installáláskor megválasztható, hogy csupán a szabad lemezterületet duplázza-e meg, vagy az inkrementális (bővítő) installációt választva a már winchesteren lévő programokat is fokozatosan pakolja be a Stacker által kezelt részbe. Ez utóbbi megoldás lassabb ugyan, de nem kell kimenteni, majd újra visszacsomagolni mindent, és az eredeti könyvtárszerkezet is megmarad.

11. Gyorsulhat az adatforgalom a merevlemezzel, mert a fizikailag kiírásra/beolvasásra kerülő rész a tömörítésnek köszönhetően kisebb, azonkívül az inkrementális installáláskor a program a sűrített állományokat egybefüggő, könnyebben elérhető fájlba zsúfolja össze.

12. Installáláskor kérhetjük, hogy egy cache terület kijelölésével tovább gyorsuljon a program. Mivel azonban ez a választás a szokványos 640 K-ból csíp le, célszerűbb külső (PC-Cache, NCache, stb.) programot használni, ami a 640 K felett, vagy az EMS területen foglal le helyet. A Stacker ezekkel is kiválóan együttműködik.

13. Megfelelő memóriamenedzsert használva a gyakran használt többi rezidens programmal együtt a HIMEM, vagy EMS területre helyezhetjük a Stacker-t is.

Mit nem tud?

A tisztesség kedvéért essék szó a veszélyekről is:

1. Ha valaki a kézikönyvben megadott minimálisan 1 megabájtól kisebb szabad lemezterülettel kezd neki az installálásnak, elveszítheti a merevlemez teljes tartalmát.

2. Keveredést okozhat, ha installáláskor elfelejtünk SWAP-ot kérni. Ez manuálisan is beállítható, akár utólag.

3. A Stacker által nem értelmezett vagy nem kezelhető (pl. A:, B:) meghajtókat a Norton Commander 0 bájt szabad területtel és 0 bájt kapacitással jelzi ki. Ez persze csak a kijelzés, ettől még ugyanúgy írja és olvassa ezeket a lemezeket, mint a többi. A PathMinder és a PCTools jobban látja ezeket az adatokat.

4. A merevlemez meghatározott területére közvetlenül író egyes programok kárt tehetnek a Stacker által ugyanott összesűritett állományokban.

5. SWAP esetén a C:\-ben lévő AUTOEXEC.BAT-hoz nem szabad hozzányúlni. Az igazi AUTOEXEC.BAT a SWAP-olt meghajtón van, az igazi CONFIG.SYS-szel együtt.

6/Egyes nagy helyigényű programok nem férnek össze vele. A megoldás: DOS 5.0. Ekkor a driver a programokat és magát a Stacker-t is a 640 K terület fölé löki, és a DOS 32 megás korlátját sem kell kerülgetnünk.

7. DR DOS 5.0 mellett nem használható a SWAP funkció.

8. Azok a programok, amelyek szintén sokat vermenek (pl. a Chessmaster 2000 sakkprogram) könnyen kiakaszthatják a rendszert, a Stacker-rel egyetemben.

9. Ha 8 K-nál nagyobb clusterekkel építetjük fel telepítéskor a STACVOL-t, akkor a 3 helyett 8 K lesz a plusz memóriaigénye, a programnak minden egyes Stackerrel bővített partícióján.

A Stacker telepítése

1. A Stacker szoftver verziója megéri az árát (1991 közepén 18 000 Ft + áfa). Úgy bővíthetjük vele lemezkapacitásunkat, hogy nem kell hardvert szerelni.

2. Installálás előtt a leírás figyelmes elolvasásával sok felesleges munkát és bosszúságot takaríthatunk meg.

3. Mind a program, mind a kézikönyv többször is figyelmeztet, hogy ugyancsak installálás előtt célszerű:

— csinálni egy biztonsági másolatot;

— lefuttatni a DOS CHKDSK-jét, ha kell, /f opcióval is;

— kitakarítani a memóriából a felesleges rezidens programokat.

Ez utóbbit úgy tehetjük meg, hogy a CONFIG.SYS és az AUTOEXEC.BAT megfelelő sorainak elejére REM-et, vagy :-ot írunk. Ezután újraindítjuk — most már rezidensek nélkül — a gépet.

4. A Stacker-t az INSTALL programmal kényelmesen lehet installálni. (Jobb, mint manuálisan, aprólékosan paraméterezgetni.)

5. Ha több tömöríteni szánt partíciónk vagy winchesterünk van, akkor az INSTALL programot mindegyiken végig kell futtatnunk. 3.30-as DOS-t használva a 16 megabájtostnál nagyobb partíciókat is csak több nekifutással lehet „felfűjni”. (Nem a Stacker, hanem a DOS bűne.) A Stacker helyigénye mintegy 30 K, plusz további 3-3 K minden Stacker-rel bővített partícióként.

Mindent előre átgondolni

Mielőtt nekilátnánk a tényleges munkának, először át kell gondolnunk, hogy mi is van a gépben és hogyan akarjuk használni:

- Hol van a CONFIG.SYS és az AUTOEXEC.BAT?
- Milyen rezidens programokat használunk?
- Koprocesszor-kártyás vagy szoftveres verziót fogunk telepíteni a gépre?
- Van-e minimálisan 1 MB szabad hely azon a winchester partíción, ahová telepíteni akarjuk? Ha nincs, akkor legalább átmenetileg csináljunk helyet néhány állomány kirakásával.
- Akarunk-e cache támogatást. A Stacker saját beépített lemezgyorsítója a DOS által is használt 640 K területből csíp le, ezért jobb a PC-Cache, az NCache vagy egyéb külső lemezgyorsító használata.
- Az meglévő üres területet akarjuk-e megduplázni, vagy a használt területet akarjuk összenyomni. Az üres területekkel pillanatok alatt végez, az inkrementális telepítés azonban meglehetősen lassú, mert mindent rögtön átrendeznie és tömörítenie kell.
- Milyen DOS-t használunk, mekkora partíciókat kezelhet az adott DOS verzió. Ajánlatos olyan DOS-t használni, amelyik nagy partíciókat is kezel.
- A winchesternek mekkora és melyik részét akarjuk megduplázni?
- A winchesteren lévő programok, batch-fájlok igénylik-e a SWAP alkalmazását? (A DR-DOS sajnos a Stacker-nek ezt a funkcióját jelenleg nem támogatja.)
- Használunk-e EMS-t, LAN-t, Memory Manager programot? Zömüket átmenetileg kikapcsolhatjuk egy-egy jól irányzott REM paranccsal a CONFIG.SYS-ben, s újraindításkor már nem zavarnak.

A Stacker használatba vétele

1. Előkészítés

- Biztosítsuk, hogy a bővítendő lemezen legalább 1 megabájt szabad lemezterület legyen.
- Indítsuk a gépet újra egy rendszerlemezzel, ahol az AUTOEXEC.BAT-ban csupán ennyi van: SET COMSPEC=C:\COMMAND.COM
- Ha már úgyis van rendszerlemez a gépben, a bővítendő winchester partíciókra futtassuk le a CHKDSK programot (pl. CHKDSK C:). Ha a program hibás területet jelez, akkor ismét futtassuk le, de már /F paraméterrel, szükség esetén újra és újra, amíg ki nem javította a hibát. Ha nem tudja kijavítani, vigyünk el szervízbe. Egy hibás tároló komoly adatvesztést okozhat.
- Miután a CHKDSK már nem talált hibás szektorokat, és a lemezzel történt indítás miatt rezidens (TSR) programok sincsenek a gépben, áttérhetünk a Stacker tényleges telepítésére.

2. Az első telepítés

- Tegyük a Stacker lemezt az A: meghajtóba és indítsuk el onnan az INSTALL programot: A:>INSTALL
- Csak a program kérdéseire kell válaszolgatnunk. Az INSTALL megkeresi a CONFIG.SYS és az AUTOEXEC.BAT állományokat a winchesteren, megkérdezi, hova akarjuk a Stacker állományait tenni. (A koprocesszor-kártyás verzió sajnos elég ritka, ezért itt most csak a szoftveres verzióval foglalkozunk.)
- Ne kérjünk cache támogatást. Egy külső cache programmal jobban járunk.

— Válasszuk ki, hogy a szabad területről akarunk-e bővített partíciót csinálni (Free space installation, ez jóval gyorsabb), vagy a lassúbb inkrementális módszert választjuk. Ez utóbbi azoknak javasolható, akik kín-keservesen létrehoztak egy használható program struktúrát és nem akarják azt elveszíteni.

— Ha az inkrementális módszert választottuk, akkor a program a winchester adott partíciójának az állományait szépen belesűrti egy rejtett rendszerállományba, összerendezve, hogy jobban ki tudja használni a rendelkezésre álló helyet. Itt célszerű lesz a SWAP opció választása is, hogy ne kelljen a batch-fájlokat átjavítani.

— Mivel a legelterjedtebb DOS verziók csak 32 megabájtig támogatják a winchester partíciókat, ezeken a verziókon a Stacker 16 megabájtos STACVOL.00? állományainak segítségével 32 megás partíciókat hozhat létre. Fejlettebb DOS verziókkal 256 megás partíciók is kezelhetők.

— Akár a szabad területet (free space), akár az inkrementális installációt választottuk, a végén a Stacker kéri, hogy vegyük ki a lemezt és az ENTER-rel indítsuk újra a rendszert.

3. További területek bővítése

Amennyiben még maradt bővítendő lemezterület és hibát sem észleltünk (ez a leggyakoribb), akkor ismételjük meg a telepítési procedúrát minden egyes bővítendő területre:

— Indítsuk lemezzel a rendszert.

— A:>INSTALL

— ...

4. A Stacker és az EMS

Megfelelő memóriamenedzser programmal a Stacker jól érzi magát az EMS területen is. Ehhez csak annyit kell tenni, hogy installálás után CONFIG.SYS-be be kell írni a megfelelő felpakoló parancsot. Néhány példa:

— 80286-os gépen a QRAM segítségével:

```
DEVICE=C:\QRAMLOADHI.SYS C:\STACKER\STACKER.COM C:\STACVOL.000
```

— 80386-os rendszeren pl. 386MAX-ban:

```
DEVICE=C:\386MAX\386LOAD.SYS      PROG=C:\STACKER\STACKER.COM  
C:\STACVOL.000
```

— DR DOS alatt:

```
HIDEVICE=C:\STACKER\STACKER.COM C:\STACVOL.000
```

Karbantartás

A bővített winchester állományainak karbantartását több segédprogrammal is segíti a Stacker.

1. SATTRIB

A DOS attrib parancsát helyettesíti. Mind a négy attribútumot változtatni lehet vele. A STACVOL-ok mindig rejtett rendszer és readonly állományok legyenek!

A parancs alakja:

```
SATTRIB [+I-A][+I-H][+I-R][+I-S] [fájlnév][.kit]
```

Ahol a + bekapcsolja, a – pedig kikapcsolja az adott attribútumot.

A = Archív.

H = Hidden (rejtett).

R = Read-only (csak olvasható).

S = System (rendszer).

Példa: SATTRIB +H +R +S TITKOS.DOC

Rejtett, rendszer, csak olvasható állománnyá teszi a TITKOS.DOC-ot.

2. SCHECK

A DOS CHKDSK parancsát helyettesíti. DESQview környezetben, illetve a Microsoft Windows alatt 386-os bővített üzemmódban nem működik. Paraméterezése a CHKDSK-t követi, de bővebb annál:

SCHECK [drive:] [/F]/[D]/[V]/[?]

Ahol

drive: = Az ellenőrizendő partíció neve. Ha nem adunk meg semmit, akkor az aktuális meghajtót ellenőrzi.

/F = A CHKDSK-hoz hasonlóan rendbeteszi a sérült részeket.

/D = Részletes hibajelentést ad.

/V = Az éppen ellenőrzött állományok neveit kiírja.

? = Help-et ad.

Az SCHECK által kiírt jelentés baloldalán a DOS által is látott adatokat olvashatjuk, a jobb oldal a ténylegesen elfoglalt helyet mutatja. Hiba esetén a /F paraméterrel futtassuk.

3. SCREATE

Ez az INSTALL program lelke. Használhatjuk a paraméterek kézi beírásával is, de ezt legfeljebb szakembereknek ajánlom, az átlagfelhasználó jobban teszi, ha az INSTALL programot használja. Biztonságosabb és jóval kényelmesebb. Nem működik, ha a DOS fastopen-je aktív. Paraméterezése:

SCREATE [D:]STACVOL.00X [/S=sss.s]/[I/[L=iii.i]]/[C=d2:]/[M]/[P]

Ahol:

d: = Az a partíció, ahová a STACVOL kerül. Ha nem írjuk be, akkor az aktuális meghajtót értelmezi.

STACVOL.00X = A tömörített fájl neve. Sorszámozása 000-tól kezdődik.

/S=sss.s = A STACVOL mérete kilobájtban, szabad területre történő (free space) installációnál. A minimum 1 K.

/I = Inkrementális installációt végeztet. Legalább 1 MB szabad terület kell hozzá. A 100 megabájt nál nagyobb lemezekre 10 megabájt onként további 100 K számolható. A kapcsoló nem használható együtt az /S kapcsolóval.

/L=iii.i = Csak az /I kapcsolóval kombinálva használható. Azt adhatjuk meg, hogy mekkora lemezterület maradjon meg tömörítetlenül az inkrementális módszernél. A 0 érték a minimumot hagyja, ami 100 K, ha /L-et nem használunk, akkor 1 megabájt marad.

/Cd2: = Csak az /I kapcsolóval használható. Azt adhatjuk meg, hogy rendszerindításkor melyik legyen a betöltő (boot) meghajtó. Használatakor a CONFIG.SYS és a benne szereplő device driver nem tömörítődik, és indításkor a STACKER.COM előtt töltődik a memóriába.

/M = Monokróm illetve fekete-fehér képernyőkhöz igazodik.

/P = Kitöm egy részlegesen kiírt STACVOL fájlt. Csak az esetleges hibák korrigálásakor használjuk.

4. SDIR

A DOS DIR parancsának kibővített verziója. A parancs használata:

```
SDIR [drive:][path\[fájlnév][.kit] [/H]/P]/W]/?]
```

Ahol

drive = A lemez neve. Ha elhagyjuk, akkor az aktuális lemez.

path = A listázandó könyvtár. Ha elhagyjuk, akkor az aktuális könyvtár. A fájlnev és a kiterjesztés dzsóker karaktereket is tartalmazhat.

/H = A rejtett állományokat is kijelzi.

/P = Egy-egy betelt képernyő után billentyűlenyomással lép tovább.

/W = Csak a fájlneveket írja ki és egy összesített tömörítési arányt. Ha ezt a kapcsolót nem használjuk, akkor az egyes állományoknál kiírja, hogy az eredeti méret a jelenleginek hányszorosa volt.

? = Help-et ad.

5. SREMOVE

Ha a winchesterről el akarunk távolítani egy Stacker-rel felszerelt partíciót, akkor az SREMOVE parancsot kell kiadnunk, a feleslegessé vált partíció nevével kiegészítve. Pl. SREMOVE d:

Az így eltávolított állomány visszahozhatatlanul eltűnik. Törlés előtt a biztonság kedvéért még kétszer rákérdez, hogy legyen időnk meggondolni.

6. SSWAP

Ez a segédprogram arra szolgál, hogy a programok számára észrevétlenül felcserélhessünk két partíciót. A program meghívását az INSTALL a CONFIG.SYS-ben helyezi el, ha SWAP-ot kértünk:

```
DEVICE=C:\STACKER\SSWAP.COM C: D:,E: F:
```

A program a megadott partíciókat párosával megcseréli. A párok közé vesszőt kell tenni.

7. STACKER.COM

Ez a program rezidens módban állandóan dolgozik. Legbiztonságosabb az INSTALL programmal berakni a CONFIG.SYS-be a STACKER.COM meghívását, de manuálisan is lehet. A szintaxis a következő:

```
DEVICE=C:\STACKER\STACKER.COM [/B=xxxx] /M=nn STACVOL.000 [STACVOL.001 ...]
```

Egyértelműen meg kell adni, hogy hol van a STACKER.COM és hol vannak a megfelelő STACVOL-ok.

/B=xxxx = A koprocesszor kártya báziscíme.

/M=nn = A kijelölt cache mérete kilobájtban (0-64). Válasszuk a 0-t !

8. SWAPMAP

Ez a segédprogram segít felderíteni, hogy mi hol volt a betöltéskor. Paramétere nincs.

A Stacker állományai

Avagy amit ne töröljünk ki a winchesterről:

1. STACVOL.XXX állományok.

Rejtett, rendszer, read-only attribútummal. Ezek tárolják adatainkat. Az adott partíció gyökérkönyvtárában található.

2. A \Stacker könyvtár

A Stacker lemezről bemásolt állományokat tartalmazza.

3. AUTOEXEC.BAT

SWAP esetén pl. a C:-ben az AUTOEXEC.BAT 2665 bájt hosszú és read-only. Nem szabad módosítani! Az igazi a megcserélt partíción van.

4. AUTOEXEC.STC

Az utolsó Stacker installálás előtti AUTOEXEC.BAT fájl tartalma kerül ide.

5. CONFIG.SYS

Ez a fájl tartalmazza azt az utasítást is, ami aktivizálja a Stacker-t.

6. CONFIG.STC

Az utolsó Stacker installálás előtti CONFIG.SYS fájl tartalma kerül ide.

7. Stacker.LOG

Azoknak az állományoknak a listája, amelyek az installáció során nem kerültek bele a STACVOL-ba. Magyarázatokkal:

Didn't fit = Túl nagy volt a fájl. (Ritkán fordul elő.)

Hidden = Rejtett állományokat nem másol.

Unable to delete = Az eredetit nem tudta törölni.

Read error = Olvasási hiba.

8. STACSYS.BAK

Az utolsó Stacker installálás előtti CONFIG.SYS fájl tartalma kerül ide is.

Hibaüzenetek

A Stacker fejlett hibakezelését mutatja, hogy a kézikönyvnek majdnem fele a hibaüzenetekkel és a hibák kijavításával foglalkozik. Terjedelmi okokból itt ezeket nem lehet ismertetni. Hiba esetén célszerű egy szótárral vagy egy angolul jól tudó kollégával kikeresni a kézikönyvből a megfelelő üzenetet és a megoldást.

Új (NewSpace) a Nap alatt

A winchester-felfújó programok közül ez az első olyan, amely rendkívül alacsony ára miatt (500 Ft) mindenki számára elérhető. A NewSpace kísérő dokumentációja azt igéri, hogy megduplázza a winchester kapacitását. Hiszem, ha látom! Nézzük meg, mit tud, s miben gyengélkedik ez a figyelemre méltó programcsomag.

Sok kellemes meglepetés

— A lemez tartalmaz egy kinyomtatható, olyasmányos, korrekt módon a korlátozások java részét is ismertető — persze angol nyelvű — kézikönyvet.

— Az installáló program jó, nem jelent gondot a program telepítése. Aki meggondolta magát, adatvesztés nélkül megszakíthatja a telepítési folyamatot, és később is rendelkezésre áll az uninstall lehetősége.

— Az alig (általában csak 5-30% arányban) zsugorítható EXE, COM, és SYS állományokat, az AUTOEXEC.BAT-ot és a normál DOS működéséhez szükséges állományokat nem bántja, így a winchester a NewSpace nélkül is használható marad. Magunk határozhatjuk meg, mekkora rezidens része legyen (Vigyázat, minél kisebb a lefoglalt memória, annál inkább lassul a program.) A tömörítésből tetszőlegesen kizárhatjuk a meghatározott állományokat.

— Korrekt módon a fizikailag tényleges szabad lemezterületet jelzi ki, nem pedig valami becsült értéket az addigi átlagos vagy becsült tömörítési határfok alapján. Ha frunk a NewSpace programmal bővített lemezre, abban a kellemes meglepetésben lesz részünk, hogy (a felvitt állományok jellegétől függően) ténylegesen a kijelzett szabad lemezterület többszöröse áll rendelkezésünkre.

— Az installálási procedúra után az AUTOEXEC.BAT-ból indíthatjuk, és nem is kell többet foglalkoznunk vele. 100 megabájtig használható, csak utána nő túl nagyra a rezidens rész.

— A shareware változatban a legutolsó, a regisztrált változatban az utolsó 99 törölt vagy felülírt állomány adatvesztés nélkül visszahozható, ha van elegendő szabad hely a merevlemezen.

— Az adat jellegű állományokat (hacsak szándékosan le nem tiltjuk), mindig tömörítve írja a felpumpált lemezre.

— A NewSpace programmal kitágított winchesteren kívül az összes többi perifériát (ramdrive, floppy, másik winchester, stb.) továbbra is változatlanul, a normál DOS formátumban kezeli a gép. Már 255 K szabad winchester területtel elindítható a program.

Amin még van mit javítani

— Nem működik együtt a multitaszk rendszerekkel (Windows, DesqView, TopView, stb.).

— Hálózatban nem működik a szerver gépeken, csak a hálózati munkaállomásokon.

— Egy gépben csak egyetlen winchesterre, illetve partícióra telepíthető. További merevlemezek illetve partíciók bővítése a NewSpace programmal már nem lehetséges.

— Nagyobb kapacitású merevlemezek esetén (40—100 megabájt) már jelentőssé válik a rezidensen lefoglalt RAM terület növekedése, ezért az ilyen lemezeken a nagy helyigényű programok futtatása problémát okoz. Bár a lefoglalt RAM terület beállítható kisebb értékre is, ekkor lassul a program. (A programnak viszont van nagyméretű winchesterekre kialakított, regisztrált változata is!)

— A leírásban és a különböző ismertetőkből közölt megduplázás csak becslés, ami a konkrét állományok jellegétől függően változik. A tömörítés hatásfoka nem haladja meg a hagyományos fájl-tömörítőkét (PKZIP, PAK, LHA, PKPAK, stb.). Ez a gyakorlatban annyit jelent, hogy a szöveges állományok feleakkora/egyharmadnyi helyet foglalnak el, mint tömörítés nélkül, a DBF állományok pedig kitöltöttségtől függően egyötödnyi/tizedannyi helyet igényelnek.

— A szinte teljesen megtelt winchesteren meglehetősen körülményes az uninstall eljárás de ez nem elsősorban a program hibája.

— A BIOS-t megkerülve közvetlenül a winchesterre író programok (pl. XTREE, stb.) megzavarhatják a program működését, mivel beleírhatnak a NewSpace adatállományba.

Lépésről lépésre...

A program nyugodtan ajánlható a kispénzű és kezdő felhasználóknak, akik csak néhány, nem túl nagyigényű programot használnak, és gépükben egyetlen, 40 megabájtól nem nagyobb kapacitású winchester található.

Ugyancsak megfelelő olyan irodai felhasználóknak is, akik egy-egy gépen csak öt-hat programot futtatnak (szövegszerkesztés, adatbáziskezelés, táblázatkezelés, könyvelés, számlázás, stb.) és új, nagyobb winchesterek beszerzésére egyelőre nincs kilátásuk.

Az első teendő, amire a shareware lemezen található angol nyelvű kézikönyv is figyelmeztet, hogy csináljunk a NewSpace lemezről egy másolatot és mindig azzal dolgozzunk. A felpumpálandó winchesterről is célszerű teljes backup-ot készíteni.

A bővítendő winchesteren a program működéséhez legyen legalább 225 K szabad terület. A NewSpace lemez másolatáról indítsuk el az INSTALL.BAT programot: A:>install, és válaszoljunk sorban a képernyőn megjelenő kérdésekre. Ezután már csak tömöríteni kell a winchesteren lévő állományokat a NEWSPACE COMPRESS paranccsal.

Mit tömörít és mit nem?

A NewSpace program semmiképp nem tömöríti a következő („exempt” = kivétel) állományokat:

- az AUTOEXEC.BAT fájlt,
- a *.SYS állományokat,
- a *.COM állományokat,
- az *.EXE állományokat,
- a rejtett és rendszer attribútumú állományokat (hidden, system),
- az eszközmeghajtó állományokat (device drivers).

A tömöríthető állományokat a leírás az „eligible” (alkalmas) jelzővel illeti. Átnevezéssel, vagy az adott állományok, állománycsoportok direkt kijelölésével az „alkalmas” állományokat átminősíthetjük „kivétel” állományokká.

Az INSTALL működése

1. A céllemezen létrehoz egy \NEWSPACE könyvtárat (ennek nevét megváltoztathatjuk, de minék) és oda bemásolja a NewSpace állományait.

2. Módosítja az AUTOEXEC.BAT fájl path bejegyzéseit, a path-ra fűzve a NewSpace programokat tartalmazó könyvtárat. Ha nem volt AUTOEXEC.BAT, akkor létrehoz egyet a megfelelő útvonal bejegyzésekkel.

3. Beírja az AUTOEXEC.BAT-ba a NewSpace indítását végző sort, hogy bekapcsoláskor automatikusan elindulhasson a NewSpace.

4. A CONFIG.SYS-ben megfelelően megnöveli a FILES értékét. Ha esetleg nem volna CONFIG.SYS állomány, a program létrehoz egyet, a ha nem volna FILES bejegyzés, akkor beszúr egyet.

5. Előkészíti a merevlemezt a tömörített állományok befogadására.

6. Újraindítja a rendszert egy reboottal, hogy a NewSpace aktív legyen.

Ezek után maga az installálás kész, a NewSpace másolati lemezét félre lehet tenni, ezen a gépen aligha lesz rá újra szükség.

Az INSTALL futtatása után a még „nyitva” lévő állományokat be kell pakolni, immár tömörítve a NewSpace területére. Ezt a NEWSPACE COMPRESS, vagy NEWSPACE C paranccsal hajthatjuk végre. Csakúgy mint az INSTALL futtatása, szintén egyszeri alkalommal végezendő el. Ettől kezdve ugyanis a program automatikusan minden állományt, amely nem „tiltott” a NewSpace számára, tömörítve tárol az ily módon kibővített meghajton.

A sűrítés bármikor megszakítható a <CTRL BREAK> gombokkal. A megszakítás után a program a NEWSPACE COMPRESS parancs ismételt kiadásával folytatható.

Tömörítés közben a program midíg kiírja az éppen kezelt állomány nevét. Ha a NewSpace lemeznevet is meg akarjuk adni a COMPRESS utasítás mellett, akkor a parancssor végére kell beírnunk.

A programhoz Help kiírást kaphatunk, ha a COMPRESS után /? jeleket teszünk.

A tömörített állományok kezelése

A tömörített állományokat másolhatjuk, átnevezhetjük használatba vehetjük, éppen úgy, mintha nem is tömörítettük volna.

A NEWSPACE STATUS <fájlnév>, vagy rövidítve NEWSPACE S <fájlnév> paranccsal pedig megkaphatjuk a tömörítési adatokat is. Ha egy olyan állomány státuszát kérdezzük meg, amely törölt, de még visszaállítható állapotban van, a válaszban a fájlnev mellett az „Erased but recoverable” szöveg áll.

Természetesen nemcsak egyes fájlok, hanem fájlcsoportok státuszát is lekérhetjük dzsóker karakterekkel (* és ?). A névben útvonal is megadható. A STATUS parancs helpjét a megszokott módon, a /? kapcsolóval, a teljes lemez státuszát pedig a NEWSPACE DISK, vagy ezt is rövidítve a NEWSPACE D paranccsal kérhetjük be.

A törölt, de még visszaállítható állományok visszahozására a NEWSPACE RECOVER (rövidítve NEWSPACE R) parancs szolgál. Kiadásakor a program kiírja az adott meghajtó törölt, de visszaállítható állományainak listáját. A shareware verzió csak egyetlen állomány visszaállítására képes, csak a regisztrált verzió tud maximum 99 állományt visszahozni, ha van elegendő szabad lemezterület. A parancs mellé megadhatunk meghajtó nevet, vagy a /? help kapcsolót.

Ha a NEWSPACE RECOVER parancs után megadjuk a program számára a visszaállítandó fájl nevét is, akkor a NewSpace az adott állományt visszaállítja, ha még lehetséges. Ha a törölt állománnyal azonos néven (pl. ABCD.TXT) már van egy másik állomány, azaz a törölt állományt felülírtuk, akkor a jelenleg hozzáférhető állományt a program átnevezi (a fenti névből ABCD.)) lesz) és a törölt állomány a visszahozás után visszakapja eredeti nevét.

A NewSpace rezidens része

Az installálás után az AUTOEXEC.BAT az alábbi sorral bővül: NEWRES [paraméterek]

Ha a rendszert lemezről indítjuk, akkor nem fér hozzá a NewSpace által tömörített állományokhoz, hacsak az AUTOEXEC.BAT állomány megfelelő sorát nem adjuk ki közvetlenül a DOS promptról, vagy egy batch fájlból.

A következő paraméterek adhatók:

/d=x ahol x a NewSpace programmal bővített meghajtó neve, pl.: /d=e;

/p=nn ahol nn a munkaterület számára lefoglalt RAM terület, pl.: /p=20, azaz 20 K területet foglal le (ez utólag manuálisan is módosítható az AUTOEXEC.BAT-ban).

Ha a NEWRES után kiterjesztéseket is írunk (pont nélkül), akkor az adott kiterjesztésű állományokat „kivételnek” tekinti a program és nem tömöríti, illetve ha tömörített állományról van szó, akkor a legközelebbi módosításakor már nyitott állapotban teszi vissza a lemezre. Így például az OVL és BIN kiterjesztésű állományok tömörítését a NEWRES OVL BIN paranccsal tilthatjuk le.

A leggyakoribb (hiba)üzenetek

Unable to compress all eligible files in place.

A COMPRESS használatakor valamelyik nagyméretű állomány tömörítéséhez kevés a program rendelkezésére álló szabad lemezterület. Ilyenkor az a teendő, hogy átmenetileg egy másik lemezre visszük át az üzenet alatt felsorolt állományt és újra indítjuk a COMPRESS-t, majd visszavisszük az átmenetileg eltávolított nagyméretű állomány, amely ekkor már rögtön tömörítve kerül a helyére.

*/? not allowed in file specification with NEWSPACE RECOVER command.

A RECOVER parancs mellett nem használhatunk a névmegadáskor dzsóker karaktereket, hanem egyértelmű nevet kell beírunk. Path alkalmazása megengedett.

Answer by typing „Y” for YES, „N” for NO. Then press Enter.

Az INSTALL és UNINSTALL programoknál az igen/nem válaszokra Y, vagy N az elfogadható válasz.

Current version of file renamed xxxxxxxx.))).

A RECOVER használatakor az azonos néven létező fájl átnevezéséről tudósít az üzenet. Teendő: nevezzük át aminek akarjuk.

File already exists

A RECOVER parancs használatakor jelentkezhet ez az üzenet, ha a visszaállítandóval azonos néven már létezik egy állomány. A program ezt átnevezi, lásd az előző címszót.

File not found and cannot be recovered.

A program nem tudja teljesíteni a feladatát, mert

- rosszul adtuk meg a nevet, vagy
- a törölt fájl „kivétel” volt, vagy
- shareware változatnál nem a legutolsó volt, vagy
- regisztrált változatnál nem az utolsó 99 törölt fájl egyike volt, vagy
- nem volt elég hely a lemezen, hogy a NewSpace megőrizze a biztonsági másolatot az állományról.

File not found.

A program nem találja a fájlt, mert

- rosszul adtuk meg a fájlnevet, vagy
- töröltük a fájlt, és az már nem visszahozható.

File recovered

A sikeres visszaállítás után jelenik meg.

Invalid NEWRES parameter: xxxxxxxxxxxx

A program rezidens részét nem megfelelő paraméterekkel indítottuk el.

Invalid or missing function. For help, enter: newspace /?

A NewSpace programot nem a STATUS, DISK, RECOVER, COMPRESS, illetve /? parancsokkal indítottuk.

NewSpace commands will not be available from any subdirectory.

A Path parancs sora nem lehet hosszabb, mint 128 karakter, és a \NEWSPACE könyvtár már nem fért bele. Három megoldás is kínálkozik:

- A NewSpace programot vigyük át egy path-on lévő könyvtárba.
- Rövidítsük le a path parancssorát, hogy beleférjen a NewSpace is.
- 3.0 DOS-tól felfelé a NewSpace parancsok kiadásakor a \NEWSPACE könyvtárnevet is beírhatjuk.

No recoverable files have been found

A program nem talált visszahozható állományokat

Not enough disk space to install NewSpace.

Az INSTALL program futtatásakor a lemezen 225 K-nál kevesebb szabad terület volt. Ha ideiglenesen áthelyezünk néhány nagyobb állományt egy másik lemezre, és újra elindítjuk az INSTALL programot, már meg is oldottuk a problémát. Ha ezután az eltávolított állományt visszatöltjük a helyére, az már tömörítve kerül oda.

System statistics being gathered, please wait.

A rendszer ezzel az üzenettel csak arra figyelmeztet, hogy nem merevedett le, hanem dolgozik és a türelmünket kéri. A bővített területek adatainak összegyűjtéséhez kell néhány másodperc a programnak. Teendő: várni.

Unable to compress all eligible files in place.

Azt jelenti, hogy nagyon tele van a lemez, takarítani kellene.

A többi üzenet részletes leírása megtalálható a program lemezén lévő (angol nyelvű) dokumentációban.

Expanz! Plus — a kis drága

Amerikában 1991 közepén jelentették be ezt az új kártyát és annak szoftvereit, de tesztelni még nem volt alkalmunk. A gyártótól származó előzetes információk szerint kijavították az előző verzió (Expanz!) néhány bosszantó hibáját, és minimálisra csökkentették a különböző DOS verziókkal való inkompatibilitását.

A rendszer néhány fontosabb szolgáltatása

— A kártya gyors koprocesszorával egyidejű (real time) adattömörítést biztosít még a lassú XT gépeken is.

— Automatikus adattömörítést végez a kiválasztott meghajtókon.

— Egyaránt tömöríthető vele ramdrive, floppy, winchester, írható/olvasható optikai lemez és Bernoulli box. A lemezek tömörítése ki is kapcsolható.

— Felismeri a már tömörített állományokat és nem vesztegeti rájuk feleslegesen az idejét.

— Bootolható partíciót is lehet vele tömöríteni.

— A program árában benne van a „Super PCKwick memory manager” program.

— Együttműködik vele többek között a PC Tools és a Norton Utilities is.

— Egyszerű és fájdalommentes az installálása.

— A tömörített lemezen/partíción lévő szabad terület kijelzésekor az Expanz! Plus csak a tömörítés nélkül felhasználható területet jelzi ki, de rendszerint kellemes szolgáltatásként ennek a többszörösét vihetjük be. (A beírandó adatok jellegétől függően 1,8—15 között van a szorzó.)

— Mindíg csak az éppen használt tömörített lemeznek megfelelő mértékben foglal el helyet a RAM-ból.

Amiben a várakozásnak még nem felel meg

— Nagyobb partícióknál nő a rezidens rész mérete.

— A tömörítés határfoka nagyjából a hagyományos fájl-tömörítők szintjén marad, tehát ebben nincs minőségi ugrás.

— Kártya nélkül a szoftver fut ugyan, de rendkívül lelassulva. Csak a gyors 486-os gépeken használható kártya nélkül is lassulás nélkül a program.

— Be kell építeni, de egyes kisméretű táska-, illetve noteszgépekbe a kártya nem fér bele.

(A program és a kártya részletes ismertetésére a könyv tervezett folytatásában, illetve az Alaplapon térünk vissza.)

Archívkezelő utility-csomag

A tömörített állományok nemcsak megkönnyítik, hanem néha meg is keserítik az egyszerű felhasználók (vagy akár a programozók) életét. Mi minden fordulhat elő?

- Éppen a szükséges tömörítő/kibontó nincs kéznél.
- Nem tudjuk, melyik archívban van, amit keresünk.
- Nem tudjuk, hogy az adott archívon belül melyiket is keressük.
- Olyasvalakinek kell programokat/adatokat gyorsan, nagy mennyiségben átadni, aki most ül először gép elé.
- A címzett teljesen más tömörítőket használ.
- Egy vagy több pótolhatatlanul értékes tömörített állományunkról utólag derült ki, hogy vírusfertőzött.
- Több, rendszerint különféle archív állományból kell összeollóznuk valamit egy új archívba.
- Lemezhiba, vagy kollegáink gondatlansága miatt az archív néhány állománya megsérült.

Az ilyen és ezekhez hasonló gondok, feladatok megoldhatók szorgos aprómunkával is, de érdekesebb elővenni a megfelelő segédprogramot, és a robot nagy részét rájuk bízni. Dolgozzon a gép, hiszen jobban bírja, mint mi, ráadásul kevesebbet is hibázik.

Fájlkereső segédprogramok

Az archiválás velejárója, hogy amikor a tömörített állományokra csak bizonyos idő eltelté után van szükségünk, rendszerint nem emlékszünk, hová pakoltuk el. Ezen a gondon segítenek a különböző fájlkereső segédprogramok. A minta természetesen a Norton Utilities FF programja volt, „csak” ki kellett terjeszteni a keresést az archivált állományokra is. A megoldásban az alábbi öt kis utility érdemel figyelmet:

PKFIND	ARC fájlban és ARC-SFX fájlban is tud keresni.
SFIND	ARC fájlban és ARC-SFX fájlban is tud keresni.
PKZFIND	ZIP fájlban és ZIP-SFX fájlban is tud keresni.
PKFF	ARC fájlban, ZIP fájlban, ARC- és ZIP-SFX fájlban is tud keresni, egyszerre két megadott meghajtón.
FFIND	Minden ismertebb archívban és SFX fájlban tud keresni tetszőleges meghajtókon.
WHEREIS	Állománykereső, alkönyvtárakban és archívokban is.

A fenti sorrend a megjelenés időbeli rendjét és a használhatóság kiterjedését is tükrözi. Aki csak ARC és ZIP állományokkal találkozik, annak a PKFF tökéletesen megfelel, aki pedig egyéb típusú tömörítővel is foglalkozik, az előbb-utóbb úgyis átáll az FFINDR-ra.

(Egy ötödik segédprogram, az MTV is ide kívánczolt volna, de terjedelmi okokból az már csak a következő kötetben kaphat helyet.)

PKFIND.COM és PKZFIND.EXE

Mindkettő a PKWare terméke. Lehetővé teszi fájlok keresését ARC illetve ZIP kiterjesztésű állományokban is, nemcsak könyvtárakban és alkönyvtárakban. Egyszerre több fájlt is kereshet.

Szintaxisa a következő:

```
PK(Z)FIND [parancsok] [drive:][út] fájl [fájl2 ...] [>kimenet]
```

A lehetséges parancsok a következők:

- /c Folyamatosan listáz.
- /b Rövid listázás.
- /d A könyvtárneveket is kiírja.
- /a Csak az archív fájllokból listáz (PKFIND).
- /z Csak az archív fájllokból listáz (PKZFIND).
- /n Archív állományban nem keres.
- /h Kiírja saját szintaktikáját.

Ha minden paraméter nélkül hívjuk meg a PK(Z)FIND-ot, akkor a /h parancs az alapértelmezés. Ha meghajtót specifikáltunk, de parancsot nem választottunk, akkor mindenhol keres. Ha a program által nem értelmezhető parancsot adtunk meg, kiírja, mit nem ért.

Amennyiben nem adunk meg meghajtót és könyvtárat, akkor annak a lemeznek a gyökérkönyvtárából keres, amelyiken az aktuális könyvtár rajta van. Ha könyvtárat adunk, akkor abban és az abból nyíló alkönyvtárakban keres.

Nem keverhető a /z és /d illetve az /a és /d, valamint a /z és /n, illetve az /a és /n parancs.

A keresendő fájl megadásához a szabványos DOS névmegadási szabályok érvényesek, ? és * dzsóker karakterek használata megengedett.

A kimenet alapértelmezése a képernyő, ezt átirányíthatjuk nyomtatóra (pl. >LPT1:), vagy fájlba. Ilyenkor a fájlnak elérési utat is adhatunk a nevében (>C:\DOKSIK\KERES.DOC). Ha a fájlba történő átirányításkor >>-t írunk, akkor a már meglévő azonos nevű fájlt nem írjuk felül, hanem csak hozzáfűzzük a PK(Z)FIND kimenetéről érkező adatokat.

SFIND.COM

A PKFIND-hoz hasonló, de valamivel szerényebb képességekkel rendelkező program az SFIND.COM. Paraméterek nélküli meghívása esetén kiírja saját szintaxisát. A következőképp használható a program:

```
SFIND [drive:] fájlnev.ext [/n]/[p] [>kimenet]
```

Meghajtó azonosító nélkül az aktív könyvtár meghajtóját böngészi végig. A fájlnevében és kiterjesztésben használhatók a ? és a * dzsóker karakterek.

A /n parancs hatására az ARC archívokban nem keres. Gyorsíthatja a munkát, ha tudjuk, hogy a keresett állományunk nincs archiválva.

A /p parancs hatása, hogy a kijelzés megáll egy gombnyomásra várva, ha betelik egy képernyő.

A kimenet alapértelmezése a képernyő. Ezt átirányíthatjuk nyomtatóra (Pl.: >LPT1:), vagy fájlba. Ilyenkor a fájlnak elérési utat is adhatunk a nevében (>C:\DOKSIK\KERES.DOC). Ha a fájlba történő átirányításkor >> jelet írunk, akkor az esetleg meglévő azonos nevű fájlt nem írjuk felül, hanem csak hozzáfűzzük.

PKFF

A program Pál Miklós alkotása, tehát nem PKWare termék, de minősége alapján ők is nyugodtan bevehetnék a ZIP csomagba.

A program használatának leírása, sőt maga a C nyelvű forrás program is megtalálható az Alaplap mágneslemez mellékletén a programmal együtt megjelent PKFF.DOC illetve PKFF.C állományokban. Azoknak, akikhez ezek nélkül jutott el a program, vagy elvesztették a fenti állományokat, röviden összefoglalom a programról a főbb tudnivalókat.

A program használata:

```
PKFF [hol_keressen] [mit_keressen] [-parancsok] [>kimenet]
```

Természetesen a PKFF-nek is egy pathon lévő könyvtárban van a helye. Ha mégsem oda tettük, akkor legyen az aktuális könyvtárban, vagy írjuk elé az elérési útvonalat is.

Ha nem adunk meghajtót, akkor az aktuális lemezen keres. Egyszerre több is megadható, ezeket a beírás sorrendjében böngészi végig. Alapesetben a megadott meghajtó gyökérkönyvtárában kezdi a keresést, ha azonban egy (al)könyvtárat is megadunk, akkor onnan indul.

Az adható parancsokat egy bevezető – jel után írjuk. Ha nem adunk ki parancsot és a PKFF környezeti változót sem definiáltuk, akkor csak hagyományosan, a könyvtárak között keres, a tömörített állományokban nem. Az alábbi parancsokat fogadja el a program:

–a ARC fájlokban is keres.

–z ZIP fájlokban is keres.

- az ARC és ZIP fájlokban is keres.
- ae ARC és ARC-SFX fájlokban is keres.
- ze ZIP és ZIP-SFX fájlokban is keres.
- aze ARC, ZIP s a belőlük készített SFX fájlokban is keres.

A parancsok helye nem rögzített, a PKFF után bárhol állhatnak a parancssorban.

A program parancsainak alapértelmezései egy PKFF nevű környezetleíró változó segítségével módosíthatók. Ezt a változót célszerű az AUTOEXEC.BAT-ban definiálni, javasolt beállítása: set pkff=aze

Módosított alapértelmezés esetén a parancsok utáni – jel lekapcsolja az adott funkciót, például a fenti PKFF változó használatakor az –a–e– parancs hatására sem az ARC, sem az ARC-SFX állományokkal nem foglalkozik, csak a ZIP és a ZIP-SFX állományokkal egészíti ki a hagyományos fájlkereső funkciót.

Természetesen a megtalált állományok listája átirányítható nyomtatóra a DOS >prn vagy >LPT1: parancsaival, illetve fájlba a >, illetve >> parancsokkal. Az utóbbi esetben a már meglévő állomány végéhez fűződik majd az újabb találati lista.

Szükség esetén a program futása <CTRL>C-vel megszakítható.

FFIND 1.33

Az előzőleg tárgyalt fájl kereső programokhoz képest az FFINDD annyiban jelent előrelépést, hogy az ARC és ZIP állományokon kívül az LHARC, LARC és a PAK programok által készített archívokkal is foglalkozik.

A SolarSoft shareware programkönyvtár #432 lemezén megtalálható programot kétféle üzemmódban használhatjuk. Párbeszédéses módban (ha paraméterek nélkül hívjuk meg) a program először bekéri a keresendő fájl nevét (* és ? dzsóker karakterek is használhatók), majd a kereséshez szükséges parancsokat.

Paraméteres (normál) üzemmódban az alábbi módon használjuk:

ffind [parancsok] [drive:] [könyvtár] fájlnev [>kimenet]

Az FFINDD-ot célszerű egy path-on lévő könyvtárban elhelyezni.

A parancsok egy bevezető – karakter után következnek. Kivétel a h és ? (mindkettő Help) parancs, ezeknél bevezető – nélkül is a Help képernyő jelenik meg.

Az adható parancsok:

- s Az aktuális könyvtárban és annak alkönyvtáraiban keres.
- a Minden meghajtón keres (B:-n is, még akkor is, ha csak egy meghajtónk van), A:-tól az utolsóig.
- c Tömörebb kijelzés. Egy találat — egy sor.
- d Kijelez minden egyes könyvtárat, ahol keresett.
- h Help képernyő. Bevezető – jel nélkül is adható.
- ? Help képernyő. Bevezető – jel nélkül is adható.
- k Könyvtáranként kiírja a megtalált fájl neveit, de nem írja ki a méretet és a dátum adatokat.
- p A teljes nevet kiírja a megtalált fájl nevében. Ahol egy archív belsejében van a keresett fájl, ott először az archív nevét írja ki, kiterjesztéssel együtt, a következő sorban pedig a fájlnevet egy + jel után.
- q Letiltja az archívokban való keresést.
- r Kiírja a megtalált archívok archívkatalógusát, mint az egyszerű DIR parancs.
- x Kipakoló batch-fájl készítésére használható parancs. Az első bekért karaktersort a talált fájl neve elé, a másodikat mögé írja.

A `-p` paranccsal együtt és célszerűen a DOS `>`, vagy `>>` átirányító parancsával kombinálva ajánlatos használni.

WHEREIS

Keith Ledbetter 1989-es keltezésű programja első ránézésre nem egy nagy szám, mondhatni tizenkettő egy tucat. Mégis, a WHEREIS.EXE 3.1 olyan speciális szolgáltatásokat biztosít, amelyek kiemelik az átlagból.

A program telepítése nem igényel semmi előkészületet. Csupán a WHEREIS.EXE állományt kell egy path-on lévő könyvtárba bemásolnunk. A haladók ezen kívül elhelyezhetik egy környezeti változóban a program alapértelmezésként választott kapcsolóit (pl. `SET WHEREIS=-a -p`).

A program futását szükség esetén bármikor megszakíthatjuk (`<CTRL C>`).

A kapott listát a szokott módon a DOS operátoraival átirányíthatjuk nyomtatóra (`>LPT1:`, `>PRN`), lemezre vagy listaállományba (`>fajllist.doc`, illetve `>>lista.lst`).

A program a megadott állomány(oka)t keresheti alkönyvtárakban és archív állományokban is. A WHEREIS-t a következő archív típusok használatára készítették fel: ARC, LZH, PAK, PKA, ZIP, ZOO.

A WHEREIS program szintaxisa a következő:

```
WHEREIS [meghajtó(k):] [fájlmaszk] [kapcsoló] ... [kapcsoló] [>kimenet]
```

Az átböngészendő meghajtó nevét közvetlenül a programnév után kell megadni és egy `:` jellel lezárni. Ha több meghajtót akarunk átnézni egy paranccsal, akkor egyszerűen fel kell sorolni a megnézendő meghajtók neveit, mindenféle elválasztó karakter nélkül. Pl. `acd:`. Ha nem adunk meg drive-ot, akkor az aktuális meghajtón dolgozik a program.

A fájlmaszkban a DOS névmegadási szabályai érvényesek, vagyis lehet benne path információ is, valamint használhatjuk a megszokott dzsóker karaktereket (`*`, `?`). Ha nem adunk meg kiterjesztést a névhez, a WHEREIS automatikusan hozzáteszi a `.*` kiterjesztést.

A WHEREIS kapcsolói

A parancshoz tetszőleges számú kapcsolót adhatunk, ezek számát csak a DOS korlátozza. (Maximum 127 karakter a parancssorban.) A kapcsolókat mindig egy bevezető – karakter után kell írni. A kapcsolók nem vonhatók össze egyetlen közös bevezető – jel mögé. A kapcsolók használatakor a program nem tesz különbséget a kis és nagybetűk között.

A kapcsolókat feladataik szerint három nagy csoportra oszthatjuk. A felosztás meglehetősen önkényes, és jelentős átfedések vannak, mert egyes kapcsolókat több csoportba is be lehetne sorolni. Az első csoportba azok a kapcsolók kerülnek, amelyek befolyásolják, hogy hol keresse az adott fájlokat a program. A második csoportot azok alkotják, amelyek a keresés során különböző szűrő feltételeket állítanak fel. A harmadik csoport a találatok kijelzésére van hatással, az `-i` kapcsoló pedig egy egyetlen tagból álló csoportot képvisel.

Hol keressen

`-a`

Az `-a` (archív) kapcsoló azt biztosítja, hogy a WHEREIS az archívok belsejében is keressen, ne csak a megadott lemez(ek) (al)könyvtáraiban. A jelenleg támogatott archív típusok: ARC, PAK, PKA, ZIP, LZH, ZOO.

-ao

A program a megadott lemez(ek) (al)könyvtáraiban nem, csupán az archívok belsejében keres az -ao (archive only) kapcsoló hatására. A jelenleg támogatott archív típusok: ARC, PAK, PKA, ZIP, LZH, ZOO.

-arc, -lzh, -pak, -pka, -zip, -zoo

Csak a megadott kiterjesztésű archív fájlok belsejében keres, a többi archívot átugorja.

-c

Csak a path utasításban szereplő (al)könyvtárakban keres.

-d

A rendszer összes hozzáférhető merevlemezén keres.

Értelemszerűen a -c és a -a(o) kölcsönösen kizárják egymást. Ezeket a kapcsolókat nem lehet kombinálni.

Hogyan keressen

-e

A WHEREIS normálisan azt feltételezi, hogy az alkönyvtárak nevei nem tartalmazzak kiterjesztést. Ha vannak kiterjesztéssel ellátott nevű alkönyvtáraink is, akkor az -e (extension) kapcsoló arra utasítja a programot, hogy ezekben is keressen.

-g

Ez a kapcsoló (goto) hatására a program az első megtalált fájlt tartalmazó könyvtárba ugrik át és befejezi a keresést. Mielőtt könyvtárt váltana, előtte azért megkérdezi, hogy csinálhatja-e.

-h

A -h kapcsoló hatására a keresés nem a gyökérkönyvtárból, hanem az aktuális könyvtárból indul.

-v

A -v kapcsoló a program egyik különlegessége. Segítségével a megadott nevű állományokat letörölhetjük. Minden egyes fájl törlése előtt rákérdez, törölheti-e. Y-ra töröl.

-w

A -w (waste) kapcsoló hatására kiírja azt a lemezterületet, amit azért veszítünk el, mert a kis fájlok egy cluster-nál kisebb helyet foglalnak el.

-x<fájlnév.kit>

A -x kapcsolóval (exclude) kizárhatunk a keresésből állományokat. A parancssorban maximum 25 fájl-specifikációt adhatunk meg egyszerre.

-lt<nnn>

A -lt (less than) kapcsoló a megadott méretnél kisebb állományokat listázza ki.

-eq<nnn>

Az **-eq** (equal to) kapcsoló a megadott méretű állományokat listázza ki.

-gt<nnn>

A **-gt** (greater than) kapcsoló a megadottnál nagyobb méretű állományokat szűri ki a megtalált állományok közül.

A méret alapján válogató kapcsolókat kombinálhatjuk egy parancssoron belül, így például kiválaszthatjuk a 20000 és 30000 bájt közötti méretű állományokat (**-gt20000 -lt30000**).

-b<dátum>

A **-b** (before) kapcsolóval a megadott időpont előtt készített állományokat választhatjuk ki. A dátumot kétféleképp adhatjuk meg: teljes dátum megadás esetén (hh-nn-éé, pl. 7-31-91) az adott dátum előtt készült, részleges dátum megadás esetén az aktuális (rendszeróra szerinti) dátum előtt legalább x nappal készültek. Ha nem adunk meg dátumot, akkor az alapértelmezés 0, azaz az összes fájl, ami a tegnapi nappal bezárólag készült.

-s<dátum>

A **-s** (since) kapcsolóval a megadott dátum után készült állományokat választhatjuk ki. Itt is kétféleképpen adhatjuk meg a dátumot, teljes dátum megadása esetén (hh-nn-éé, pl. 7-31-91) az adott dátum után készült állományokat írja ki a program. Ha csak a napot adjuk meg, akkor azokat a fájlokat kapjuk meg, amelyeket a megadott x napnál nem régebben módosítottak. Ha nem adunk meg dátumot, akkor az alapértelmezés 0, azaz a mai napon készült állományokat sorolja fel a WHEREIS.

A **-b** és **-s** kapcsolókat természetesen együtt is nyugodtan használhatjuk, és így kijelölhetjük a megadott időintervallumban készült vagy módosított állományokat.

Hogyan írja ki

-f

Az **-f** kapcsoló egy olyan kiírást eredményez, amely csak a fájlneveket tartalmazza, minden egyéb információ nélkül. Rendszerint listafájlok készítéséhez használjuk, átirányítással (>LPT1:, >lista.lst). Ne kombináljuk a **-m** és **-r** kapcsolókkal.

-m

A találatok kilistázásakor a fájlnevek alapértelmezésként a lista bal oldalán vannak (név, méret, dátum, idő, path sorrendben). Ha az **-m** kapcsolót használjuk, akkor a név középre kerül, közvetlenül a path elé.

-r

Ha az **-r** kapcsolót használjuk, akkor a név középre kerül, közvetlenül a path elé (méret, dátum, idő, név, path sorrendben).

-p

A -p (pause) kapcsoló hatására a WHEREIS megáll és egy billentyű lenyomására vár, ha egy képernyőnyi információt már kiírt.

-y

A dátumot kiírásakor a program amerikai formátumban írja ki alapértelmezésként, hacsak a WHEREIS változóval nem módosítottuk ezt: hh-nn-ée (pl. 7-31-91). A -y kapcsoló hatására európai dátum formátumban írja ki a program a talált fájlok keltezését: nn-hh-ée (pl. 31-7-91).

Egyéb

-i

Amennyiben definiáltunk a WHEREIS meghívása előtt (pl. az AUTOEXEC.BAT-ban egy WHEREIS nevű változót, ami a program kapcsolóinak alapértelmezéseit tartalmazza, az -i kapcsoló ezt figyelmen kívül hagyja.

Archív kukkantók

Az archív listázó és kibontó segédprogramok rendeltetése az, hogy a megadott archív állományokba belekukkanthassunk. Az ide sorolható programok kiírják az archívok katalógusát, némelyikük pedig még plusz funkciókkal is rendelkezik. Kihagytam azokat a programokat, amelyek csak egyetlen archívot kezelnek, és a következő „jobb” programokat ismertetem: MAD, FV, LHDIR, ZIPVIEW, LIST.

MAD.EXE

Sokak gondja az, hogyan nézhetnek bele gyorsan és felesleges paraméterezések nélkül a tömörített állományok (hagyományos és SFX) tartalomjegyzékébe (az archívkatalógusba). Erre a feladatra kiválóan alkalmas Mike Sax programja, a MAD.EXE. (SolarSoft lemez, #412.)

A program paraméterezése átvette a DOS DIR parancsának megszokott kapcsolóit (/w és /p), így könnyebb megtanulni azokat.

Ha a programot minden kísérő paraméter nélkül hívjuk meg, akkor egy rövid Help képernyőn közli velünk, mire képes és hogyan. Ebből megtudhatjuk, hogy olvasni tudja az ARC, PAK, ZIP, ZOO, DWC és LZH kiterjesztésű archívokat, valamint ezek EXE, illetve COM kiterjesztésű SFX-eit. Az LHA segítségével készített LZH és LZH-SFX állományokat is kilistázza, nemcsak az LHARC és LHICE archívjait.

A program paraméteres hívása a következőképpen néz ki:

```
mad archív [/w] [/p] [/b] [/t] [>kimenet]
```

A MAD.EXE programot célszerű egy path-on lévő könyvtárban elhelyezni, így meghívásakor nem kell az elérési útvonal beírásával annyit vacakolni.

Az archív nevében elfogad ? és * (dzsóker) karaktereket is. A kiterjesztést általában elhagyhatjuk, de ha azonos néven több archívunk is van különféle kiterjesztéssel, akkor a MAD mindegyiket sorban kilistázza. Ha csak az egyik archívot akarjuk megvizsgálni, akkor egyértelműbb a névmegadás kiterjesztéssel együtt. Kiválóan alkalmas a program arra, hogy megállapítsuk egy COM vagy EXE állományról, hogy vajon futtatható állomány, vagy csak egy SFX archív. Az SFX-ről azt is megtudhatjuk, melyik tömörítővel készült.

A parancsok előtt bevezető / karaktert kell írunk. Ha nem adunk parancsot, akkor alapértelmezésként kiírja az archívban tárolt állományok neveit, méretét pakolás előtt és után, a tömörítési arányt, a használt tömörítési módszer jelét és a fájlok keltezését. Egyszerre több parancsot is adhatunk, ilyenkor a parancsok között szököz legyen. A MAD az alábbi parancsokat értelmezi:

/w Megegyezik a DOS DIR parancsának /w kapcsolójával, azaz soronként öt nevet ír ki kiterjesztéssel az archív nevét követő sortól kezdve. Egyéb információt nem ad.

/p Megegyezik a DOS DIR parancsának /p kapcsolójával, azaz billentyű leütésre várva megáll, ha betelt egy képernyő.

- /b Path nélkül írja ki az archív neveket
- /t Kiírja a CRC értékeket is.

Természetesen a program kimenete a DOS > operátorával átirányítható nyomtatóra vagy fájlba is.

FV.COM

Az FV.COM annak ellenére, hogy mintegy feleakkora méretű, mint a MAD.EXE, további extra szolgáltatásokkal áll rendelkezésünkre. (SolarSoft lemez, #432.) A szokásos ARC, PAK, ZIP, ZOO, LZH, és DWC állományokon kívül az LBR fájllokba is belenézhet. Sajnos az LHA által étrehozott SFX állományokba nem mindig tud belenézni, bár a vele készített LZH fájl katalógusát gond nélkül olvassa ki.

Használata:

fv archív [fájlok] [parancsok]

Az FV.COM is akkor hatásos igazán, ha path-on lévő könyvtárba tesszük.

Az archív nevébe beírhatjuk az elérési utat is, ha nem az aktuális könyvtárban van. A kiterjesztést akkor kell csak megadni, ha több azonos nevű, eltérő kiterjesztésű archívunk is van az adott könyvtárban, s közülük csak egyre vagyunk kíváncsiak. Az archív nevében dzsóker karaktereket is használhatunk.

Az archív neve után megadhatunk fájlmaszkokat, egy parancssorban egyszerre legfeljebb nyolcat, és a program csak az annak megfelelő állományokat jeleníti meg.

A program parancsai egy bevezető / jel után írhatók be a parancssorba. Az alábbi parancsokat értelmezi a program:

- /w Megegyezik a DOS DIR parancsának /w kapcsolójával, azaz soronként öt nevet ír ki kiterjesztéssel az archív nevét követő sortól kezdve. Egyéb információt nem ad.
- /p Megegyezik a DOS DIR parancsának /p kapcsolójával, azaz billentyű leütésre várva megáll, ha betelt egy képernyő.
- /b Path nélkül írja ki az archív neveket
- /r Átírja az archív dátumbejegyzését, és a bepakolt állományok közül a legfrissebbhez igazítja az archív dátumbejegyzéseit.
- /c A PKZIP-pel bepakolt állományok megjegyzéseit is megjeleníti.
- /x Leszedi az SFX állományokról az önkinyitó fejet és visszaalakítja normális archívvá a megfelelő formátumban és a megfelelő kiterjesztéssel az aktuális könyvtárba.

LHDIR.COM

Dennis Vallianos 1989-ben az LHARC 1.13-hoz készült programja (SolarSoft lemez, #432) igen hasznos kis apróság. Háromféle célra használható:

1. LZH archív kilistázó. Akár az LHarc, ez a program is belelát az LZH archívokba.

Pár példa a használatára:

```
LHDIR ABC.LZH
LHDIR ABC
LHDIR ABC *.TXT *.EXE
```

Belelát az önkinyitó LZH-SFX állományokba is. Ilyenkor azonban az SFX kiterjesztését is egyértelműen meg kell adni a parancssorban:

```
LHDIR ABC.EXE
```

LHDIR ABC.EXE *.TXT

2. LZH-SFX — LZH archívkonvertáló program. Ugyanis a fájl archívkatalógusának kiírása után a program megkérdezi, ha a fájl SFX volt, hogy átalakítsa-e normál LZH állománnyá. Amennyiben Y a válasz, az LHDIR létrehoz a COM vagy EXE kiterjesztésű SFX állományból egy normális LZH archívot, s a forrás SFX-et háborítatlanul a helyén hagyja.

3. Önkinnyitó LZH-SFX kereső segédprogramként is használható. Ha a programot LHDIR -S alakban hívjuk meg, az aktuális könyvtár EXE és COM állományait (maximum 300 COM és 300 EXE lehet egyszerre) végigbogarássza, hogy melyikük lehet LHarc SFX állomány.

ZIPVIEW.EXE

Ez a Norton Commanderhez illeszkedő ugribugri program megtalálható az Alaplap lemez mellékletén, valamint a SolarSoft #509-es lemezén. A Norton Commandert (NC), a számítógépek közelében mindenki ismeri. A programcsomag igen hasznos segítői a különböző betekintő programcskák, amelyekkel táblázatokba (123VIEW), dBASE, Reflex, R:base, Paradox 3 adatállományokba (DBVIEW, REFVIEW, RBVIEW, PARAVIEW), szövegekbe (WPVIEW) és grafikai (PCXVIEW) jellegű állományokba lehet betekinteni az adott programba való belépés nélkül.

Csokonait idézve „egy híját esmértem örömminek még...”. Ugyanis a tömörített állományokba való bekukucskálás lehetősége sokáig hiányzott a Norton Commander arzenáljából. Most itt van! Egy ügyes holland programozó F.A. Oldenhuis (Robert van Hoeven munkáit felhasználva) sok vetélytársát megelőzve megírta a ZIPVIEW programot, amely szervesen beilleszkedik az NC csomagba.

Mit tud ez az apró program?

— Felismeri és kilistázza a legelterjedtebb tömörítő programok által előállított archív állományok (ZIP, ARC, LZH, PAK, ZOO, DWC) belső tartalomjegyzékét (név, a fájl tömörített és eredeti mérete, a tömörítettség aránya, CRC). Sajnos a többszörösen pakolt állományok néha megzavarhatják.

— Ugyancsak felismeri a fenti archívokból készített önkinnyítő archívokat, és azoknak is kiírja a belső tartalomjegyzékét, ha meghíváskor az SFX fájl kiterjesztését is megadjuk.

— Ha az adott archív fájl kicsomagolója path-on van, akkor az <F3> lenyomásával a kurzorral kiválasztott fájlt kinyitja és kiírja a képernyőre. Megtekintés után az átmenetileg létrehozott fájlt törli. Ez csak akkor működik, ha sz NCMAIN.EXE állományt már módosítottuk, vagy lecseréltük. Addig ugyanis nem tekinti magáénak a ZIPVIEW-t.

— A kijelzett archív tartalomjegyzék állományai közül kedvünkre kijelölhetünk az Insert gomb, valamint a szürke + és – (a numerikus billentyűzeten) segítségével. A kiválasztott állományokat pedig a ZIPVIEW ki is nyitja a path-on lévő megfelelő kibontókkal. Nekünk csak azt kell még beírunk, hogy hová pakoljon ki. A kibontó program megfelelő meghívása már nem a mi dolgunk, ezt elvégzi helyettünk a ZIPVIEW.

— Az EGA és VGA monitorokon a Norton programokhoz hasonlóan támogatja a 43/50 soros megjelenítést, valamint egérrel is vezérelhető.

— Természetesen az <F1>-re kiírja saját tömör Help-jét is. Kilépni pedig az <F10>, vagy az <ESC> billentyűkkel lehet.

A ZIPVIEW program kipakoló utility-ként való használatához öt dolgot kell biztosítanunk:

— A megfelelő kibontó program(ok) a path-on legyen(ek), (PKUNPAK, PKUNZIP, PAK, ZOO, ARC, vagy ARCE, LHA, vagy az LHARC).

— A ZIPVIEW meghívása után ki kell jelölnünk a kibontandó állományokat (<Insert>, <szürke +>/<szürke -> a numerikus billentyűzeten).

— Meg kell adnunk, hová pakoljon ki, de ha nem adunk meg célkönyvtárat, akkor az aktuális könyvtárba nyit.

— A célkönyvtárban legyen elegendő hely a kikapolandó állományoknak.

A ZIPVIEW elindítása

a./ DOS promptról:

C:\ZIPPED>ZIPVIEW kakukk.ZIP

A jelenlegi, 1.0 verzió az archív nevében nem fogad el ? és * dzsóker karaktereket. Mind a nevet, mind a kiterjesztést be kell írni.

b./ A Norton Commanderben az indítás:

Az egyik ablakban legyen az a könyvtár, ahol a ZIPVIEW van. Álljunk rá, majd <CTRL j>, vagy <CTRL ENTER> hatására a kijelölt állomány, jelen esetben a ZIPVIEW.EXE neve a képernyő alján lévő parancssorba kerül. Menjünk át a másik ablakba, és álljunk rá a megnézendő archív nevére, és ismét a <CTRL j>, vagy <CTRL ENTER> majd <ENTER> leütése következik. Így anélkül is megnézhetjük például egy önkinyitó EXE vagy COM archív állomány archív katalógusát, hogy az SFX nevét egyáltalán leíránk. A kevésbé lusták legalább a ZIPVIEW nevét saját királyi kezükkel bepötyögthetik.

c./ A legkényelmesebb módszer:

— Először betesszük a ZIPVIEW-t az \NC (\NORTON, stb.) könyvtárba a Norton Commander mellé, hogy biztosan egymásra találjanak.

— A Command menüpont eXtension file edit parancsát választva (<F9>-C-X) beírjuk az alábbi pár sort:

```
ZIP: ZIPVIEW !!  
ARC: ZIPVIEW !!  
LZH: ZIPVIEW !!  
PAK: ZIPVIEW !!  
ZOO: ZIPVIEW !!  
DWC: ZIPVIEW !!
```

majd elmentjük a változtatást az NC-ben megszokott <F2>-vel, és kilépünk.

— Most már csak rá kell állnunk a megnézendő (nem SFX) archív nevére és az <Enter>-t lenyomni.

Természetesen az EXE és COM kiterjesztésű önkinyitó archívokat így nem tudjuk megnézni, azokhoz be kell pötyögnünk a DOS promptról a szükséges adatokat (lásd fent), vagy a lustábbaknak való fenti eljárást is követhetjük.

Amennyiben helytelenül paraméterezve hívjuk meg a ZIPVIEW programot, akkor a Help képernyő jelenik meg, és röviden leírja a program használatát.

Magyarország nem lenne az, ami, ha valaki át nem írta volna már az NCMAIN.EXE állományt úgy, hogy az NC végre teljesen a sajátjaként kezelje a ZIPVIEW programot. A tartalomjegyzékben egy archiv fájlra állva az <F3> gomb lenyomása azonnal meghívja a ZIPVIEW-t, amennyiben a már átfírt NCMAIN.EXE dolgozik.

Ez utóbbi beavatkozással kapcsolatban még két megjegyzés:

— Az NCMAIN.EXE átírása sajnos nem sikerült tökéletesen. A ZIPVIEW-t gond nélkül indítja, de a kedves kolléga sajnos éppen az 123VIEW helyébe tette be az archív betekintőt, ezért azután a Lotus táblázat-állományok, megtekintéséhez nem áll többé rendelkezésünkre az <F3>, csak a kiterjesztésekhez rendelt menü. Nem lenne rossz, ha valaki úgy tudná átírni az NCMAIN-t, hogy pl. a sokkal ritkábban használt PARAVIEW, vagy a REFVIEW, esetleg az RBVIEW helyére illesztené be a huncut kis ZIPVIEW-t.

— A színes kártyával és mono monitorral felszerelt PC/XT/AT gépek használóinak fontos: a ZIPVIEW jelenlegi verziója sem a Norton Commanderből, az <F3> billentyűvel történő meghívásakor, sem az NC eXtension menüjén keresztül indítva, sem a DOS parancssorból meghíva nem ad igazán jól olvasható színeket. Célszerű lehet a MODE MONO DOS paranccsal, vagy a Norton Utilities NCC programjával előzőleg beállítani a gépet a monokróm színekiosztáshoz. Mono kártyával ugyanis igen jól olvasható színekiosztást kapunk.

Az újabb verzió a szerző (F.A. Oldenhuis) leírása szerint egy ZIPVIEWB.ZIP nevű állományban indul majd útnak a BBS-ekben. Remélhetőleg az újabb verzióknak már nem holland, hanem angol nyelvű lesz a leírása, és az LHA, valamint az ARJ programokkal is hajlandó együttműködni.

LIST

Vernon D. Buerg LIST programjának különböző változatai meglehetősen elterjedtek. Előszeretettel használják a csak átmenetileg kibontott archív szövegállományok megjelenítésére. Érdemes a legújabb verziót beszerezni, mert az egérhasználat és számos plusz szolgáltatás kellemessé és könnyűvé teszi a program kezelését.

A LIST 7.5e program eredeti angol nyelvű kézikönyve mintegy 102 K méretben taglalja a program használatát. Itt most a mindennapi használatban legfontosabb információkat gyűjtöttem egybe. Ami kimaradt, az az eredeti programcsomag LIST.DOC és WHATSNEW állományaiban illetve a program megfelelő Help képernyőin megtalálható.

A SolarSoft #432 lemezen található LIST75E#.EXE csomagban a közkedvelt LIST programnak három verziója is van:

1. LISTS.COM

A legkisebb (small) verzió a legkevesebbet is tudja. Csak 30 K szabad RAM kell használatához, néhány extra szolgáltatásról azonban le kell mondanunk:

- Csak 600 K-nál kisebb állományok megnézésére alkalmazható. (Kinek nem elég?)
- Kimaradt az <ALT x> (képernyőmentés) funkció.
- Kimaradt az <ALT g> (goto DOS).
- Minimális a Help képernyő.

2. LISTR.COM

A normál (regular) változat már jóval többet tud (16 megabájtig kezeli a fájlokat), de hiányzik belőle az <ALT v> (file selection), az <ALT i> (insert filename), továbbá minimális az <ALT w> (windows) lehetősége, azaz legfeljebb két egyforma nagyságú ablak lehet, és nincs mód az ablakok méretének megváltoztatására.

3. LIST.COM

A Plus verzió a csomag legtöbbet tudó változata. A továbbiakban mi is ezzel foglalkozunk.

A LIST Plus szintaxisa parancssorból való meghíváskor:

list [fájlok] [/kapcsolók]

A megjelenítendő fájlok: Megadható több név és használhatunk dzsóker karaktereket is a fájlneveknél. Az egyes fájlok között a <CTRL PGUP> és a <CTRL PGDN> billentyűkkel válthatunk. Amennyiben nem adunk meg fájlnevet, akkor az aktuális könyvtár tartalomjegyzéke jelenik meg, ahonnan rámutatással választhatunk.

A LIST kapcsolói

- /S — Kapcsolt, vagy átirányított fájlt használ.
- /J — Junk filter bekapcsolva.
- /7 — 7-bit display.
- /8 — 8-bit display.
- /* — A * filter bekapcsolva.
- /W — A sortördelés (WRAP) be van kapcsolva.
- /H — Hex dump mode.
- /L — Pre-Loading on.
- /M — Egérhasználat.
- /K — Nincs egér.
- /E — A fájl végével jelentkezik be.
- /Fkeresett_szöveg — Azonnal megkeresi a megadott szöveget.
- /#nnnn — Az nnnn-edik sorban/rekordon jön be.

A LIST mint DOS SHELL (keret) program

A LIST-et fájlnev megadása nélkül hívja az aktuális könyvtár állományai jelennek meg. Ezek közül nemcsak választani tudunk, de a LIST lehetővé teszi az alapvető DOS funkciók elvégzését is, menüből (<F2>-re jön elő) és egybetűs parancsokkal is. Ezek a következők:

Copy — <C>

A normál DOS COPY parancsot helyettesíti, de annál intelligensebb, mert jelzi, ha a célkönyvtárban van már a másolandóval azonos nevű állomány és rákérdez, felülírhatja-e.

Delete — <D>

A normál DOS DEL/ERASE parancs. Itt is visszakérdez a törlés előtt.

Editor — <E>

Az adott állományhoz meghív egy szerkesztő programot, amit egy path-on lévő EDIT.BAT fájl segítségével indíthatunk. Ebben megfelelően paraméterezhetjük az adott programot. Pl.

```
PEII %1 /pc:\pe2.pro /q
```

Ha az AUTOEXEC.BAT-ban definiáltunk egy LIST változót, pl. a

```
SET LIST=d:\temp
```

formában, akkor az átmeneti állományok oda kerülnek.

Invoke — <I>

Végrehajtja az adott programot (EXE, COM, BAT) a megadott paraméterekkel.

Move — <M>

Az éppen kijelölt állományhoz bekér egy célkönyvtárat és a fájlt áthelyezi oda (COPY + DEL).

Path — <P>

Könyvtár vagy lemezegység váltását végezhetjük vele.

Quit — <Q> / <X> / <F10> / <ESC>

Kilépünk a katalógusból. Ha az <ALT V>-vel jutottunk ide, akkor visszajutunk a LIST-be, ha közvetlenül a DOS-ból, akkor visszakerülünk a DOS-ba, illetve a hívó batch-fájlba. Az X és az <F10> az új könyvtárba, ha munka közben könyvtárt változtattunk, az <ESC> a kiindulási könyvtárba visz.

Rename — <R>

Az adott állomány átnevezésére szolgál. Itt is rákérdez, mielőtt végrehajtaná.

Sort — <S>

A könyvtár állományait rendezhetjük át. Almenüje: Filename, Extension, Date, None, Size.

ViewArc — <V>

A szintén a SolarSoft #432 lemezen található FV program aktív közreműködésével kilistázza az adott archív tartalomjegyzékét egy FVFVVFV.FV\$ átmeneti állományba, és ezt mutatja meg. Használható a LZH, ARC, PAK, ZIP, ZOO, DWC és LBR állományok, valamint az LHARC által készített SFX-ek esetében.

<F1>

Meghívja a katalógus funkciókhoz tartozó Help-et.

<F2>

Meghívja a directory menüt. A menü parancsai: List, Copy, Del, Edit, Move, Path, Rename, Sort, ViewArc, F1-Help, F10-Exit.

Kilépés a LIST-ből

<F10> — Kilép a DOS-ba, az új könyvtárba. Ha munka közben könyvtárat váltottunk, a DOS prompt a legalsó sorban lesz, fölötte az utolsó kijelzett képernyőnyi szöveggel.

<ESC> — Kiléptet a DOS-ba, az induló könyvtárba, s otthagyja a képernyőt.

<X> — Kilép a DOS-ba, az új könyvtárba. Ha munka közben könyvtárat váltottunk, letörli a képernyőt és a prompt a legfelső sorban lesz.

<ALT x> — Úgy lép ki a DOS-ba, hogy a LIST használatba vétele előtti képernyőt állítja vissza.

<Q> — Úgy lép ki a DOS-ba, hogy a LIST használatba vétele előtti képernyőt állítja vissza.

A képernyős kijelzés

<ALT e> — EGA/VGA módokba kapcsol át.

<ALT h> — ASCII/Hex módok között kapcsol oda-vissza.

- <ALT r> — Ruler (felső sorvezető vonalzó) be és kikapcsolása.
- <F2> — A Find ablak háttérszínét állítja egy színnel arrébb.
- <F4> — A Find ablak betűszínét állítja egy színnel arrébb.
- <F5> — A szöveg alatti háttérszínét állítja be.
- <F6> — A szöveg színét állítja be.
- <F7> — A státusz- és parancssor háttérszínét állítja be.
- <F8> — A státusz- és parancssor betűinek színét állítja be.

A státusz-sor

Kétféleképpen nézhet ki. Az alapértelmezés a következő:

LIST llll nnnn +sss mm/dd/yy hh:mm – fájlnev

ahol:

- llll Az elsődleges kijelző ablak első sorában éppen kijelzett sor sorszáma.
- nnnn A fájl utolsó rekordjának a sorszáma, ha még nincs beolvasva, üresen marad.
- +sss A 80. karakteren túli oszlopok száma.

A konfiguráció mentése

<ALT C> — Kimenti a LIST.COM-ba az aktuális beállításokat. A rendszeres egerhasználatnál például kellemetlen és felesleges újra meg újra beírni az /m kapcsolót. Ehelyett csak az első alkalommal írjuk azt be, majd az <ALT C>-vel mentjük el a beállítást. A következő alkalommal a LIST már tud egerészni a kapcsoló nélkül is.

A szalámitaktika

Számtalan esetben fordul elő, hogy hiába tömörítettük állományainkat, mert a létrejött archív még mindig túl nagy ahhoz, hogy kitehessük floppyra. Így azután vagy újra pakoljuk a mamut méretű archívot több kisebb archívba, vagy felszeleteljük. Erre a feladatra jelenleg két program áll rendelkezésünkre, a ZIPDMP és a SLICE. Mindkettő magyar alkotás, érdemes lenne nyugaton is piacképes terméké fejleszteni. Mind a ZIPDMP, mind a SLICE mellé igen jól használható leírást mellékeltek.

ZIPDMP

PKZIP-pel készített túlsúlyos archívok és SFX állományok szeleteléséhez készítette Somorjai Ákos (Debrecen) a ZIPDMP.EXE programot. Segítségével a céllemezre még éppen ráférő ZIP vagy EXE archívokba tudjuk a böhönc fájlt átcsomagolni.

A SLICE programhoz képest kevesebb és több is ez a program. Kevesebb, mert kizárólag a PKZIP állományaival foglalkozik, továbbá terjedelmesebb és egyedül nem boldogul, csak ha maga a PKZIP és a PKUNZIP valahol path-on van. Nem mindig adja ki a 0 bytes free kijelzést, továbbá csak 400 fájl kezelésére képes és maximum 20 céllemezt tud előállítani. Aki ennél nagyobb állományokat hoz létre, az gyakoroljon önkritikát és használja inkább a SLICE/SPLICE programot, esetleg kézzel rámolja szét a góliátot. Többet tud viszont a ZIPDMP annyiban, hogy a szeletkék önálló életet is élhetnek, mivel a céllemezeken normális .ZIP archívok vagy ZIP-SFX-ek jönnek létre, s azokat már a szokott módon kezelhetjük.

A program akkor is bajban van, ha túlsúlyos kedvenceink még tömörítve is nagyobbak a rendelkezésre álló lemezkapacitásnál (pl. 360 K). Ilyen esetben használjuk inkább a SLICE programot, vagy rakjuk át az egészet az ARJ-vel egy többrészes archívba.

A program használata

```
zipdmp ZIP_fájl [kimeneti_fájl] [-parancsok]
```

Talán felesleges is megemlítenem, hogy a ZIPDMP.EXE egy path-on lévő könyvtárba helyezendő.

A ZIP fájl a szétszeletelendő ZIP fájl neve. Kiterjesztést csak akkor kell írni, ha EXE kiterjesztésű ZIP-SFX-ről van szó. A névben elérési útvonal is szerepelhet.

A kimeneti fájl megadása nem kötelező. Ha elhagyjuk, akkor a forrás ZIP fájl nevének első hat karakterét kapja, kiegészítve egy toldalékkal (_A, _B, ... stb.), ami a sorrendbe állításhoz szükséges. Ha a forrás fájl neve hat karakternél rövidebb, a kimeneti fájl neve is értelemszerűen rövidebb lesz. Ha a forrás fájl neve hét vagy nyolc karakteres volt, a ZIPDMP az első hatot egészíti ki a már említett két karakterrel.

Természetesen a fentieket figyelembe véve a kimeneti fájl megadásával magunk is ellenőrizhetjük a névadást. Amennyiben pedig az új fájl neve „a:” vagy „b:” karakterekkel

kezdődik, akkor a program minden floppy betétele előtt egy billentyű lenyomására vár. A név természetesen path-t is tartalmazhat.

A ZIPDMP hasznosítható parancsai

- d Alapértelmezés ez a parancs. Ilyenkor 360 K-s DSDD lemezekre dolgozik a program és maximum 354 K méretű lesz a lemezre szeletelt fájl.
- h A parancs hatására HD-s floppyra dolgozik a program.
- xxx Méretmegadással is dolgozhatunk, ha a fentiekől eltérő lemezekre dolgoznánk. Ne felejtjük, hogy a névlegesnél mindig egy kicsit kevesebb a ténylegesen rendelkezésünkre álló szabad hely.
- z A parancsot abban az esetben használjuk, ha a szétszedendő fájl egy EXE kiterjesztésű SFX. Használatakor a céllemezen továbbra is csak ZIP fájlok lesznek. Ha nem használjuk, akkor az SFX-ek szeletelése után a szeletek is EXE kiterjesztésűek lesznek.

SLICE/SPLICE

A szeletelő/összeragasztó csomag három állományból áll:

SLICE.COM — A szeletelő program.

SPLICE.COM — Az összeragasztó program.

5SLICE.FLP — Az eredeti programismertető.

A SLICE felszeleteli a kijelölt nagyméretű állományt akkora darabokra, hogy elférjen a céllemeze(ek)en. A program tetszőleges nagyméretű állományok szeletelésére alkalmas. Egy mamutból maximum 99 lemezméretnyi állományt hozhat létre, ami eddigi tapasztalataink szerint szinte minden óriás állományra elegendő. A SPLICE a szeletekből a céllemezen visszaállítja az eredeti állományt.

A szeletelő SLICE program használata:

slice [forrásdrive:][könyvtár]fájlnév célmeghajtó:

A SLICE.COM legyen egy path-on lévő könyvtárban, vagy írjuk be elé az elérési útvonalat is.

Ha a forrásfájl nem az aktuális meghajtón van, meg kell adni a meghajtó nevét is a szokott módon. A forráskönyvtárral ugyanez a helyzet.

A fájlnev tartalmazza a kiterjesztést is. A program szerzőjének közlése szerint egy apró, de többnyire észrevehetetlen hibácska maradt a programban: a hat karakternél hosszabb neveket a szeletelés előtt le kell rövidítenünk, mert ennél hosszabb nevekkal a programnak nehézségei támadnak. Hat karakter hosszú vagy ennél rövidebb nevek esetén a hiba nem jelentkezik. A fájlnevet minden esetben meg kell adni. A célmeghajtó megadása is kötelező.

A program az első céllemezre a szeletelt állomány mellé felírja az összeragasztó SPLICE programot is. A szeletek kiterjesztésének utolsó két karaktere értelemszerűen 01, 02, ..., 98, 99 lesz.

A SLICE folyamatosan figyeli a céllemez és azt csurig tölti úgy, hogy (az utolsó lemezt kivéve) 0 szabad bájt terület marad.

Az összeragasztó SPLICE program használata:

[splice_path]splice forrásmeghajtó: [célmeghajtó:][path]

A splice_path és a forrásmeghajtó legtöbbször azonos. A célkönyvtár elhagyható, ha onnan hívjuk meg az összerakó programot. A program egymás után bekéri a szeleteket tartalmazó lemezeket, csak arra kell ügyelni, hogy a megfelelő sorrendben adjuk be a lemezeket.

A program egyik legnagyobb előnye, hogy DOS verziótól teljesen függetlenül bármilyen kiterjesztésű állományt szét tud szeletelni és azt vissza is tudja állítani. Mivel a szeletelő a még szabad tárterületet figyeli, bármilyen floppyra (360 K, 720 K, 800 K, 1,2 MB) dolgozhat, még akkor is, ha azon már vannak egyéb állományok.

Konvertálók, SFX-készítők, kiollózók

E segédprogramok ismertetése előtt egy hasznos tipp: azokat az archívokat, amelyeket egy adott könyvtárban nem akarunk átalakítani, célszerű átmenetileg eldugni a konvertáló programok elől a NO.COM programmal.

REZIP

A REZIP.EXE program (SolarSoft #434) azzal a céllal készült, hogy a PKZIP régebbi verzióival készült ZIP fájlokat az új verzióval átcsomagoljuk. Használatához szükséges, hogy path-on legyen az új ZIP verzió PKZIP.EXE és PKUNZIP.EXE állománya és elegendő szabad lemezterület is kell a munkakönyvtárban, mert a REZIP először kipakolja az összes állományt a frissítendő ZIP fájlból és csak utána ereszti rá a PKZIP-et a Move paranccsal az ideiglenesen kibontott fájlokra. Mindenkit figyelmeztetek, ez nem egy villámgyors eljárás. Gyorsítható, ha a ramdrive-on van elegendő hely a munkához, vagy igen igen gyors winchestert használunk.

Hiba esetén abban a könyvtárban, ahol a REZIP.EXE maga is van, létrehoz egy PKERRORS.LOG állományt, amelybe becsülettel bevezeti, ha valami hiba történt. Hibás ZIP fájlt nem pakol át.

Használata:

```
rezip régifájl [újfájl]
```

(Nem kell a ZIP kiterjesztést beírni, a program hozzáteszi magától).

TOZIP

A TOZIP.EXE program (SolarSoft #434) a PKUNPAK és a PKZIP segítségével konvertál át ARC állományokat ZIP állományokká. Egyirányú konverziót tesz lehetővé. Hasonlóan működik, mint a fentebb említett REZIP, de az ARC fájl kibontásához értelemszerűen a PKUNPAK programot hívja meg. Hiba esetén a REZIP-hez hasonlóan írogat a PKERRORS.LOG fájlba.

Használata:

```
tozip ARC_fájl ZIP_fájl
```

2LZH

ARC, PAK és ZIP állományokat konvertál át LZH formátumra a PKXARC, PAK, PKZIP és az LHARC segítségével. (SolarSoft #412.) Nincs vele egyéb teendőnk, csak belépni abba a könyvtárba, amelyben a konvertálandó állományokat elhelyeztük, majd egyszerűen beírjuk: 2lzh, utána <ENTER>. Ez is csak egyirányú konverziót biztosít.

LHDIR

Az archív kilistázó programok között már ismertetett LHDIR program képes arra, hogy az LHARC 1.13 által előállított SFX archívokról levágja az SFX fejet és így létrehozson

egy normál LZH kiterjesztésű archívot. Ehhez egyszerűen el kell indítanunk a programot a következő módon:

```
LHDIR LZH-SFX.EXE
```

A program kiírja a fájl archívkatalógusát és megkérdezi, hogy csináljon-e belőle normális LZH archívot. Az LHA-SFX állományok (2.xx verzió) lefejezése az eltérő SFX fejméret miatt nem vezet a kívánt eredményre.

CHECKOUT

A CHECKOUT.EXE program eredendően a különböző archívokban való víruskeresést szolgálja, de használható arra is, hogy a megfelelő kibontó és bepakoló programok meghívásával konvertáljon akár különböző típusú archív állományok között. A program használata azért is előnyös, mert a program a felhasználó számára egyetlen lépésben

- elvégzi az archív épségének ellenőrzését,
- a SCAN programmal vírusellenőrzést végez,
- konvertál a megadott formátumra,
- egy megadott könyvtárba kigyűjti a sérült, illetve a fertőzött archívokat, állományokat és
- munkájáról naplót vezet.

A programnak és használatának részletes leírását lásd a Vírusháború című részben.

MAKESFX + PKSFX.PGM

A PKARC és a PKPAK csomagok része a MAKESFX.COM program, amely a PK36.EXE illetve a PK361.EXE (régebbi verzióknál értelemszerűen más a szám) önkinyitó archívból kimásolja az önkibontó programfejet PKSFX.PGM néven. Ehhez a fent említett SFX törzsállományok valamelyikének az aktuális könyvtárban kell lennie a MAKESFX.COM mellett. A program használata:

```
makesfx
```

Itt nem kell semmiféle paramétert beírunk. Ha a PKSFX.PGM állományt már létrehoztuk, célszerű a PKARC vagy PKPAK mellett tartani az archiváló könyvtárban. A MAKESFX programot pedig nyugodt szívvel törölhetjük, hiszen feladatát már elvégezte, és szükség esetén bármikor újra kibonthatjuk a PKARC, illetve PKPAK csomagot tartalmazó PK36.EXE, illetve PK361.EXE SFX állományokból.

A következő lépés a fenti módon előállított PKSFX.PGM és az SFX állománnyá átalakítandó ARC fájl összefűzése. Erre a következő szerkezet szolgál:

```
copy /b pksfx.pgm+ARC_fájl.ARC SFX-fájl.EXE
```

A copy parancs utáni /b paraméterre feltétlenül szükség van a hibátlan kapcsoláshoz. A PKSFX.PGM, az archív és a létrehozandó SFX egyaránt lehet egy távolabbi könyvtárban, akár másik meghajtón is, akkor azonban meg kell adni nevében az elérési útvonalat. A kiterjesztéseket a parancssorban szereplő mindhárom névhez pontosan és egyértelműen (* és ? nélkül) be kell írni. Az így előállított SFX mintegy 13.1 K-val nagyobb lesz, mint a forrás ARC fájl. Cserébe azonban külön kibontó segédprogram nélkül is rendelkezésre áll a PKXARC, illetve a PKUNPAK majdnem teljes parancskészlete.

Az PKARC/PKPAK-SFX állományok használatáról részletesebben lásd a PKARC/PKPAK csomagokról szóló fejezet önkibontó SFX állományokkal foglalkozó részét.

SEZ

A SEZ a ZOO programcsomag önkinyitó SFX-et készítő segédprogramja. (SolarSoft #428.) Viszonylag kicsi, 3 K-nál kisebb SFX fejet rak fel a ZOO fájl elejére, azonban a ZOO-SFX kizárólag önmaga teljes kibontására hivatott. Mivel a ZOO nem a tömörségéről híres, az SFX készítése előtt távolítsuk el az átalakítandó ZOO fájlból a felesleget, amivel a SEZ által létrehozott SFX úgysem tudna mit kezdeni. Ezek a következők: régebbi generációk, törölt állományok, megjegyzések, path információk stb.

Az így előkészített ZOO fájlból a SEZ egy olyan SFX-et kreál, amit bárki ki tud bontani, még ha hírét sem hallotta a ZOO-nak. A kiterjesztéseket nem kell beírni. A SEZ után elsőnek adott nevet a program ZOO fájlnek tekinti és ZOO kiterjesztést illeszt hozzá, a második megadott fájlnev pedig EXE kiterjesztéssel SFX lesz. Az SFX-fájl megadásakor ne használjunk path-t! Az SFX-et az aktuális könyvtárban tudja elkészíteni. Működtetése:

```
sez ZOO_fájl SFX_fájl
```

MKSARC

Az MKSARC.EXE programot, amely az ARC602.EXE állományból bontható ki, az ARC által készített tömörített állományok önkibontó archívva alakítására írták.

```
mksarc ARC_fájl[.ARC] [SFX_FILE]
```

A kiterjesztés alapértelmezése az archívnál ARC. Ha nem adunk meg SFX fájlnevet, akkor az archív nevét kapja EXE kiterjesztéssel. Az ARC fájl növekedése kb. 7,8 K. Az így előállított SFX archív kezelése ugyanaz mint a hagyományos ARC fájlké, csak a kiterjesztést is meg kell adni. (SolarSoft #431.) Az ARC-SFX saját parancsai megegyeznek a Vernon Bueg által írt ARCE kibontó program parancsaival.

ZIP2EXE

A PKZIP & PKUNZIP 1.10 csomag szerves része a ZIP2EXE.EXE program, amely fájdalommentesen elvégzi a megadott ZIP fájl kibővítését SFX állománnyá. Használatáról, paramétereiről, a vele készült ZIP-SFX állományok lehetőségeiről, tulajdonságairól részletes ismertetés található a PKZIP csomaggal foglalkozó fejezetben. Itt csak annyit, hogy a ZIP2EXE az 1.10 verzióban már nem igényli külön a PKSFX.PRГ fájl jelenlétét, és a mindössze 12,8 K-val hosszabb ZIP-SFX állomány szinte teljesen úgy kezelhető, mint egy szabályos ZIP fájl, viszont kibontó segédprogram nélkül is a PKZIP majdnem teljes parancskészlete rendelkezésünkre áll, a -v kivételével.

PKZIPFIX

A PKZIPFIX.EXE szintén a PKZIP csomag oszlopos tagja. A PKZIP ennek a kis segédprogramnak köszönheti híres és nagyfokú adatbiztonságát. Ha egy ZIP fájl megsérül, ami ugye előfordulhat lemezhiba vagy emberi gondatlanság miatt, nem kell az egészet úgy ahogy van kidobnunk, a PKZIPFIX programmal kiollózzhatjuk és egy PKFIXED ZIP fájlba pakolhatjuk át a még menthető részeket. Gyakran csak egy-két fájl sérül meg, főleg ha lemezhiba az ok, így a ZIP fájl többi állományát nem kell elveszítünk. Használata:

```
pkzipfix sérült[.ZIP]
```

A program által generált új ZIP fájlt ajánlatos minél hamarabb átnevezni, nehogy a legközelebbi kiollózásnál véletlenül felülírjuk egy másik kiollózott állománnyal.

REARJ

Az ARJ210.EXE programcsomag tagja a REARJ.EXE program. A megfelelő kibontó, tömörítő programok jelenlétében lehetővé teszi az átkonvertálást tetszés szerint az ARC, PAK, LZH, ZIP, ZOO, DWC, HYP, LZS állományok között. A célarchív alapértelmezése természetesen ARJ. Egyéb tömörítők használata is lehetséges, ha a REARJ.CFG állományt megfelelően módosítva felkészítjük erre a programot. (Lásd részletesebben a konfigurációs fájl ismertetésénél.)

A program egy átmeneti munkakönyvtárat hoz létre az aktuális könyvtárban, és munkája befejeztével el is takarítja azt. Segédállományként a REARJ.CFG segíti a program munkáját (lásd később bővebben).

A programmal elvileg egyszerre is átkonvertáltathatjuk az adott winchester összes archív állományát, de ez csak akkor biztonságos, ha elegendő szabad helyünk van még a winchesteren. A program használata:

```
REARJ [<kapcsolók>] <archív> [<archív> ...]
```

A REARJ.EXE és a REARJ.CFG állományoknak path-on kell lenniük, vagy az aktuális könyvtárban. A kapcsolók bevezető / jelek mögött állhatnak, a parancs után bárhol a parancssorban. Ha – bevezető katraktert használunk, akkor a Unix-szerű path megadást lefordítja (/ jelből \ lesz). Maximum 100 archív adható meg egy menetben. Dzsóker karakter és path is lehet az archív névben.

A REARJ kapcsolói

/a (= Archives within archives)

Az archívon belüli archívokat is konvertálja, például a ZIP-elt ZIP-et. Ha ezt a kapcsolót alkalmazzuk, akkor a REARJ.CFG-nek mindenképpen a path-on kell lennie, mert a REARJ rekurzív módon meghívja önmagát a beágyazott archív konvertálásához.

/d (= Delete original)

A forrásarchívot a sikeres konvertálás után törli. A readonly attribútumú archívokat nem törli.

/e (= don't return Error)

Nem ad vissza hibakódot, ha nem talál átalakítandó archívot. Beágyazott archívok átpakolásakor használatos.

/i (= Integrity check)

Ellenőrzi a REARJ.EXE program épségét. Csak a 3.0-nál korábbi DOS verzióknál kell megadni a REARJ.EXE teljes nevét is. (REARJ /ic:\uti\rearj.exe)

/l<logfile> (= Log file)

Naplózza a REARJ munkáját a kapcsolóval megadott állományba.

/o (= Overwrite)

A könyvtárban azonos néven lévő archívokat felülírja.

/q (= Query)

Minden egyes archív átalakítása előtt egyenként engedélyt kér a munkára.

/r (= Recurse)

Az aktuális könyvtár illetve a forráskönyvtár összes alkönyvtárában is keres. Segítségével egyetlen paranccsal a winchester gyökérkönyvtárából kiindulva az összes archívot átalakíthatjuk.

/s (= Skip verify)

Átugorja a fájl méret- és számellenőrzését.

/t<kit> (= Target)

A célarchív típusát adhatjuk meg ezzel a kapcsolóval. Az alapértelmezés ARJ.

/u[<kit>] (= Update with backup)

A forrásarchívról backup-ot hagy vissza. Ennek a kiterjesztése megadható a kapcsolóval, ha a BAK nem felelne meg.

/v (= Virus)

Végrehajtja az eredeti archívból a munkakönyvtárba átmenetileg kibontott állományokon a REARJ.CFG-be bejegyzett parancsot, ami a legtöbbször célszerűen a SCAN, megfelelő paraméterezéssel. A Parancs elé VIRUS-t kell írni. Ha a meghívott parancs nem 0 hibakóddal tér vissza, akkor a REARJ a kibontott állományokat nem pakolja be a célarchívba.

/x (= eXclude files)

A kapcsolóval megadott állományokat kizárja a folyamatból. A névben lehetnek dzsóker karakterek is.

/z

A konverziós folyamatot csak szimulálja, nem történik kibontás, felépítés, törlés.

A konfigurációs fájl

A programmal szállított REARJ konfigurációs fájl támogatja a ZIP, ARC, ARJ, PAK, LZH, LZS, DWC, HYP és ZOO állományok kezelését. Az ARC-hoz a PKPAK és PKUNPAK programokat használja. Az LZH-hoz az LHA programot keresi meg, a többi értelemszerű. A rendszer bővíthető újabb tömörítőkkal is, csak megfelelőenki kell tölteni az adott tömörítőhöz rendelt sorokat.

A fájl részei:

Az első sor (VIRUS SCAN /nomem *.*) akkor kerül végrehajtásra, ha a REARJ programot a /v kapcsolóval hívtuk meg.

Minden archívformátumhoz négy sor tartozik:

— Az első a formátum (= kiterjesztés).

— A második sor a bepakoláshoz, a harmadik sor a kikapcsoláshoz használt parancs-szerkezet. Ezekben az archív nevének a helyén %s áll. Amennyiben a DOS-szűrőket is használni kell, akkor a "COMMAND /C " szöveget kell eléjük írni.

— A negyedik sor tartalmazhat A, D, AD karaktereket, de üresen is maradhat. Azt jelzi a REARJ-nak, hogy az adott tömörítő kezeli-e a rejtett és rendszer jellegű állományokat (A), valamint támogatja-e a teljes könyvtárszerkezet tárolását az archívban (D).

Nem lehetnek feleslegesen üres sorok és megjegyzések a REARJ.CFG állományban, de bevezető szöközőket használhatunk a jobb tagolás érdekében. Ha újabb tömörítővel akarjuk bővíteni a rendszert, a fentiek figyelembevételével egy tiszta ASCII szövegszerkesztővel (pl. NE) tetszés szerint bővíthetjük, módosíthatjuk.

A REARJ naplófájl bejegyzései

Sikeres konvertáláskor a REARJ bejegyzi az egyes állományok főbb adatait: keltezés, célarchív típusa, eredeti archív mérete, új archív mérete, megspórolt bájtok, az eredeti archív neve.

Hibakódok a naplófájlban:

- 1 Nem találta a fájlt.
- 2 A fájl nem tartozik a konfigurált típusok közé.
- 3 A célarchív már létezik.
- 4 Nincs elég szabad lemezterület.
- 5 A felhasználó nem kérte.
- 6 Kipakolási hiba.
- 7 Bepakolási hiba.
- 8 A célarchív nem kezel könyvtárakat.
- 9 Hibás fájlszám.
- 10 Hibás összméret.
- 11 A beágyazott archívval volt probléma.
- 12 Nem tudta az archívot átnevezni.
- 13 A /v kapcsolóval meghívott program hibajelzést adott. Talán vírus?

Kiegészítő segédprogramok

Az alábbiakban tárgyalt segédprogramok nem magukkal az archív állományokkal foglalkoznak, de velük jobban ki tudjuk használni a különböző tömörítő és archívkezelő programokat.

NO.COM Utility DOS-hoz, tömörítőkhöz.

TREED.COM A Könyvtárszerkezetet grafikusán kiíró program.

NO.COM

A NO.COM egy olyan kisméretű (736 bájtos) segédprogram, amit a DOS parancsok illetve a rejtett és a rendszerfájlokat nem látó programok hatáskörének leszűkítésére használhatunk.

A NO átmenetileg, amíg a PC a vele egy sorban kiadott parancsokat végre nem hajtotta, a letiltandó fájlokat rejtett és rendszerfájlokká alakítja át, majd a parancs végrehajtása után visszaalakítja őket azzá, amik voltak.

Egy sorban több NO is kiadható. A letiltandó fájlok megadásakor a DOS névmegadási konvenciói érvényesek. Így használhatók vele a ? és * karakterek. Használata a következő:

NO fájlnev [NO fájlnev [...]] parancs paraméterek

A fájlnevbe be kell írni a NO-val kizárandó fájl kiterjesztését is, valamint, ha nem az aktuális könyvtárban van, akkor az elérési útvonalat is.

Ha a NO-t használjuk, ügyeljünk arra, hogy a végrehajtandó parancsot/programot ne szakítsuk meg <CTRL C>-vel vagy a reset gomb megnyomásával a parancs lezárása előtt, mert ilyenkor a NO nem állítja vissza a letiltott fájlok attribútum bitjeit. Ha ilyen eset előfordul, akkor a PC Tools, Norton Commander vagy egyéb segédprogramok

segítségével az elállítódott bitek visszaállíthatók. Ha a NO után semmit sem írtunk, akkor kiírja saját szintaxisát és közli, hogy elfelejtettünk fájlokat kijelölni.

A NO.COM igen jól használható az összes tömörítővel. A PKZIP és az LHA ugyan gond nélkül bepakolja a rejtett és rendszerállományokat is, de csak akkor, ha ezt a megfelelő parancs, kapcsoló vagy konfigurációs állománybeli változó használatával kifejezetten kérjük. A PKZIP rendelkezik egy kizáró (-x) paranccsal is, de azért jó, ha a NO is kéznél van.

TREED.COM

A TREED.COM egy olyan segédprogram, amellyel a kijelölt lemezegység könyvtárszerkezete jeleníthető meg grafikusán. Mivel kimenete átirányítható nyomtatóra vagy fájlba, igen hasznos szolgáltatás, amikor bonyolultabb könyvtárszerkezettel rendelkező programot akarunk átmenetileg elpakolni. Használata:

```
treed [meghajtó:][könyvtár] [>[>]kimenet]
```

A program utáni első paraméter a meghajtónév (A:, B:, stb.). Ezt meg kell adni, ha nem az aktuális meghajtó könyvtárszerkezetét akarjuk kiírni. Megadható könyvtárnév is, amire a TREED.COM csak az illető könyvtár alkönyvtárait rajzolja fel.

A második paraméter a kimenet. Alapértelmezés a képernyő. Ha ide >LPT1:-et vagy >PRN-t adunk, akkor a printeren rajzolja ki az adott könyvtár szerkezetét. Ha fájl a kimenet, akkor abba a könyvtárba írja alapértelmezésként, ahonnan meghívtuk a TREED.COM parancsot, kivéve, ha a kimeneti fájl nevében elérési útvonal is szerepel. A könyvtárszerkezetet leíró állományoknak célszerű .DOC vagy .DIR kiterjesztést adni. Ha a fájlba történő átirányításkor két > jelet írunk, akkor a már meglévő azonos nevű fájlt nem írjuk felül, hanem csak hozzáfűzzük. Példák:

```
TREED C: >LPT1:
```

```
TREED C:\SEGED >C:\DOKS\SEGEDDIR.DOC
```

```
TREED D: >C:\DOKS\DDIRLIST.DOC
```

```
TREED D: >>C:\DOKS\DDIRLIST.DOC
```

Ha nem adunk meg paramétert, akkor a program az aktuális meghajtó teljes könyvtárszerkezetét kirajzolja a képernyőn. Ha valamit elírtunk, és a program nem tudja értelmezni, akkor közli velünk, hogy nem érti, és leírja saját szintaxisát.

Keretjáték

A DOS-hoz képest nagy ugrást jelentettek a különféle shell (keret) programok. Ezek néhány jeles képviselőjével (Norton Commander, PathMinder, Xtree, PC Tools, Windows, stb.) bizonyára mindenki találkozott már.

Hasonlóképpen a tömörítő programok is egyre-másra bővülnek többféle archívkezelő funkcióval, és sorra jelennek meg a tömörítők használatát hihetetlenül leegyszerűsítő archív shell programok is. Ezek közül néhány keretprogram csak egy-egy tömörítő használatát támogatja (ArcMaster — SEA ARC, ZIPGUN — PKZIP), a jövő viszont várhatóan az integrált rendszereké. A NARC 4.0 és az IDCSHELL 4.0 a ARC és ZIP állományokat támogatja, a 4ZIP a DOS korlátain igyekszik átbújni, a külön fejezetben tárgyalt SHEZ 6.2 pedig esélyesen pályázhatna a tömörítők Norton Commandere cím elnyerésére.

ZIPGUN

A SolarSoft #434 lemezen található ZIPGUN 3.02 olyan keretprogram, amely szinte teljesen kiszolgálja az átlagos ZIP felhasználó mindennapi igényeit. Bár mindössze egyetlen archív típust használ, de azt meglehetősen jól. A ZIP csomag alig egy-két szolgáltatását felejtették ki a programból. A program a PKZIP 0.9x-es verziójához készült, de zökkenőmentesen használja az 1.10-et is. Használhatósága kitűnő, noha egerrel még jobb lenne. (Az 1990. májusi változatban ez legalábbis még hiányzik.)

A program működéséhez szükség van a path-on lévő PKZIP.EXE, PKUNZIP.EXE és LIST.COM programokra is, mivel a ZIPGUN nem maga tömörít, csak a fenti programok működését hangolja össze.

A bejelentkezés után egy tömör ismertető van a képen, valamint a program főmenüje. Az <F1> billentyű mindig Help-et hoz, az <F10> pedig képernyőt töröl és a főmenübe ugrik vissza bárhol. Kilépés a Quit Exit (qe) paranccsal.

Az adatbekérő ablakokban, ha már előzőleg volt valami adat benne, a legutolsó szöveg látható. A főmenü menüpontjai a következők: Add, Directory, Extract, Remove, Update, View, Quit. A menüből az <F10> után a megfelelő kezdőbetűk lenyomásával választhatunk, amire egy almenü gördül le villámgyorsan.

1. Add (bővítési) funkciók

Add — Bekéri a ZIP fájl nevét és a bepakolandó állományokat a menüpont választását követően megnyíló ablakban. A PKZIP-et -a paranccsal hívja meg.

Comment — Csak ZIPkommentet lehet adni a Comment menüpont választását követően megnyíló ablakban, fájlkommentet nem. A PKZIP-et -z paranccsal hívja meg.

Interchange — Valamiféle frissítést végez, de hogy ez az f, az a, az i, vagy az u paranccsal történik, még nem derült ki eddig.

Move — Mint az Add, csak nem -a, hanem -m paranccsal hívja meg a PKZIP-et, azaz bepakolás után törli a már bepakolt forrásállományt.

Az <F1>-re egy kis Help-et ad:

- Alfajnumerus karaktereket vár a névmegadásokhoz.
- Megadja a használható Wordstar-mozgató kódokat.
- Megadja a használható mozgatóbillentyűket.
- ^R (=CTRL-R), vagy ^U (=CTRL-U) visszaállítja a mező eredeti tartalmát.
- <ESC>-re egy szinttel visszalép.

2. Directory (könyvtár)

Change — Egy kis ablakban felkínálja az aktuális könyvtárat. Ha szerkeszteni akarjuk, akkor először valamelyik nyíl billentyűt nyomjuk meg, és utána módosítsunk, különben teljesen kézzel kell beírni az egész könyvtárnevet, a meghajtónévvel együtt. Ha elrontottuk, akkor a <CTRL r> illetve a <CTRL u> visszahozza a parancs kiadása után megjelenített könyvtárnevet.

Display — Kijelzi az aktuális könyvtár tartalmát.

Az <F1> billentyű lenyomásakor egy jól használható Help képernyő ugrik be, amely leírja az onnan használatba vehető funkciók kezelési módját.

A kijelölt állományok neve előtt egy ✓ jel van. A kijelölés a <SPACE>, <TAB>, illetve a <SHIFT TAB> gombokkal történik. A listában a <PgUp>, <PgDn>, <Home>, <End> és a nyíl billentyűkkel lehet közlekedni. Nem érdemes 5-6 állománynál többet egyszerre kiválasztani, mert úgysem fér bele a parancssorba, ahova a ZIPGUN írja.

Ha a betűbillentyűk közül lenyomunk egyet, akkor a kurzor a lenyomott betűvel kezdődő első állománynévre ugrik.

Ha egy állomány nevére állva megnyomjuk az <ENTER>-t, a program a LIST segítségével megmutatja nekünk a fájl tartalmát.

Bár az egyes menüpontokhoz tartozó funkciók az adott menüből is elérhetők, célszerű a Directory Display menüpontból az <ALT> billentyűvel indítani a megfelelő eljárást, mert akkor az itt kijelölt állományokkal dolgozik a program és kevesebb nevet kell nekünk beírni.

<ALT a> — (Add) — A kijelölt állományok mellé bekér egy ZIP fájl nevet, és a PKZIP-pel oda bepakol. Csak akkor működik, ha legalább egy bepakolandó állományt már kijelöltünk.

<ALT d> — (Delete) — A kijelölt állományokat simán kitörli az adott könyvtárból. Csak akkor működik, ha legalább egy törlendő állományt előzőleg már kijelöltünk, nem elég az, ha ráállunk egy fájlnevre.

<ALT e> — (Extract) — A kijelölt ZIP fájlokat kibontja az aktuális könyvtárba. Csak kijelölés után használható.

<ALT u> — (Update) — A kijelölt állományokkal felfrissíti a képernyő tetején kinyíló ablakban megadható ZIP fájlt.

<ALT v> — (View) — Kijelzi a kijelölt ZIP fájl archív katalógusát (tartalomjegyzékét). A PKZIP normál -v parancsát használja erre a célra.

<ALT t> — (Tag) — Egy speciális kis ablak nyílik, amelynek segítségével

— egyszerre kijelölhetjük a könyvtár összes állományát (a),

— egyszerre törölhetjük az összes kijelölést (n),

— átkapcsolhatunk: ami kijelölt volt, az jelöletlen lesz, és viszont (c).

<ESC>-re egy szinttel visszalép.

3. Extract (kibont)

Freshen — A menüpont választásakor megnyíló ablakban bekéri a ZIP fájl nevét és a frissítendő állományokat. A PKUNZIP-et -n paranccsal hívja meg.

Overwrite — Ezt a menüpontot akkor válasszuk, ha mindenképp felül akarjuk írni a kibontandó állományokkal a célkönyvtárban már meglévő azonos nevű fájlokat.

4. Remove (eltávolít)

Delete — A menüpont választása után egy ablakban bekéri a ZIP fájl és a belőle kitörölendő állományok neveit. A PKZIP -d parancsát használja.

5. Update (aktualizál)

Freshen — A menüpont választása után bekéri egy ablakban a ZIP fájl és a frissítendő állományok neveit. A PKZIP -f parancsát használja.

Move — A menüpont választása után egy ablakban bekéri a ZIP fájl és a bepakolandó állományok neveit. A PKZIP -m parancsát használja.

Update — A menüpont választása után egy ablakban bekéri a ZIP fájl és a frissítendő állományok neveit. A PKZIP -u parancsát használja.

6. View (archív katalógust klír)

Normal — Először is be kell adnunk a kilistázandó ZIP fájl nevét (a ZIP kiterjesztés elhagyható). Ha ez megtörtént, akkor a PKZIP -v parancsával kilistázza a megadott archívot.

Technical — A View Technical menüpont választása után a program először bekéri a kilistázandó ZIP fájl nevét (a ZIP kiterjesztés elhagyható), majd a PKZIP -vt parancsával listázza.

7. Quit (kilép)

Exit — A program befejezi működését és visszalépünk a DOS-ba.

FreeWare+ — Egy tájékoztató képernyőt ad ki a program.

Menüprogram az LHARC-hoz

Az LHMENU-t az LHARC program használatának megkönnyítésére készítette Rick Hicks. Szükség van hozzá Haruyasu Yoshizaki LHARC.EXE programjára is a path-on. Ha az LHARC program éppen más nevet viselne (a 2.xx verziók már az LHA nevet kapták), nevezzük át nyugodtan LHARC.EXE-nek, legalább a SHEZ 5.x is látja. Ez a program a PKARCM mintájára készült s így azok akik már használták a PKARCM programot, ezt is könnyedén kezelik majd.

Az LHMENU állományai

LHMENU.EXE	Maga az LHMENU program.
LHMENU.DOC	Az angol nyelvű eredeti programismertető.
LHAMENU.DOC	A magyar nyelvű leírás.

A program indítására írjuk egyszerűen be: LHMENU <archív>. Az archív neve tartalmazhat elérési utat is, valamint dzsóker karaktereket, de akkor csak listázni, megnézni, vagy kibontani tudjuk az archívot, módosítani nem. Ha az EXE kiterjesztést is beírjuk az LHMENU meghívásakor az archív neve után, akkor az LZH-SFX állományokkal is dolgozik a program.

A menüben a nyíl gombokkal mozgunk, majd a választott menüponton állva <ENTER>-t ütünk. Jó az is, ha a választott menüpont nagybetűjét nyomjuk le, akkor nem kell az <ENTER>.

Menüpontok

Az archívkatalógus kiírása: List Contents of File — L — Kiírja a képernyőre a megadott LZH fájl, illetve LZH-SFX archív katalógusát.

Egy választott állomány megnézése: Browse a file — B — Ha ezt választjuk, a program megkérdezi, hogy a bepakolt állományok közül melyiket akarjuk megnézni. Dzsóker karaktereket is elfogad a névben, és egy-egy képernyőnyi anyag megjelenítése után egy billentyű lenyomására vár. Ha több állomány is megfelel a maszknak, amit adtunk, akkor egymás után mindet megnézhetjük. <CTRL C>-re megszakad a kiírás és visszajutunk a menübe. Ha nem adunk meg nevet, az archív összes állományát előhozza.

Kibontás: Extracting files — E — Ezzel a menüponttal az aktuális lemez aktuális könyvtárába bonthatjuk ki a választott állományokat. Több név is megadható, valamint használhatók a dzsóker karakterek is. Ha nem adunk meg fájlnevet, mindent kipakol.

Bepakolás és aktualizálás: Update/Add to the file — U — Ez a menüpont ad lehetőséget arra, hogy az archívot új állományokkal bővítsük, vagy frissítsük. Be kell adni a bepakolandó állományok neveit. A dzsóker karakterek itt is használhatók. Ha nem adunk meg fájlnevet, mindent bepakol. Ha új archívot akarunk létrehozni, arra is alkalmas hogy létrehozza az új LZH állományt.

Áthelyezés az archívba: Move files to a LHARC file — M — Ez a menüpont arra való, hogy a forrásállományokat úgy pakolja be az archívba, hogy utána, ha sikeres volt a pakolás, ki is törli a forráskönyvtárból a már bepakolt állományokat. Először visszakérdez a program, hogy valóban törölni akarjuk-e a forrásállományokat, majd bekéri az áthelyezendő állományok neveit. Természetesen a dzsóker karakterek itt is alkalmazhatók.

File törlése archívból: Delete a file from the LZH file — D — Kitakarítja a megadott állományokat az archívból. A dzsóker karakterek itt is alkalmazhatók.

Épségellenőrzés: Testing LHARC file integrity — T — A jelen verzióban még nem működik. Ez az LHMENU olyan LHARC program verzióhoz készült eredetileg, amely még nem támogatta az épségellenőrzést.

Kilépés a DOS-ba: Quit to DOS — Q — A parancs hatására az LHMENU befejezi működését.

IDCSHELL és NARC

Ami tetszik

— E két program harmonikusan kiegészíti egymást. Jól használhatók, ha elsősorban DOS keretprogramot akarunk, alkalomszerűen kisebb tömörítési feladatokkal.

— A programok parancskészletét logikusan, egyszerűen és áttekinthetően állították össze. Az IDCSHELL kiadható parancsai mindig a képernyőn vannak.

— Lehet vele egerészni is.

— Archív készítésekor, bővítésekor a forrásállományokat egyszerre több könyvtárból is kiválaszthatjuk.

— Tömörítő segédprogram nélkül pakol be és bont ki.

Ami nem tetszik:

— Főleg a bepakolás nagyon lassú.

— Tömörítésben jócskán elmarad a PKZIP 1.10-től.

— Nincs Move a bepakolásakor.

— Nincs mód jelszavas titkosításra.

— Nem adhatók megjegyzések az egyes bepakolt állományokhoz, csak az egész ZIP fájlhoz.

— A ZIPkommentben nem fogad el ékezetes betűket.

— Nem tud SFX-et csinálni.

— Az 1.0 PKZIP formátumot használja csak.

Teszt:

Magát az IDCSHELL és a NARC állományait tömörítettem egy 10 MHz-es AT gépen (20 MB winchester). A 17 fájl eredetileg 310 658 bájt helyet foglalt el.

Program	Opció/parancs	Méret (bájt)	Csökkenés	Idő (perc:sec)
ARJ 2.10	a	127 668	58,90%	1:02
LHA 2.12	a	129 234	58,40%	1:12
PKZIP 1.10	-aex	133 980	56,87%	0:45
IDCSHELL	Kompresszióra	154 238	50,35%	11:12
PKZIP 1.10	-aes	162 194	47,79%	0:20
IDSCHELL	Sebességre	163 606	47,33%	0:57

Az IDCSHELL használata

Az IDCSHELL parancsai a képernyő jobb felén állandóan láthatók, hacsak egy kinyíló ablak (input, help, vagy almenü) átmenetileg el nem takarja. Ezen kívül a leggyakrabban alkalmazott parancsok külön parancssorban is megtalálhatók a képernyő alján.

Az állományok, könyvtárak, illetve parancsok közötti mozgás az egérrel, vagy a szokásos mozgatóbillentyűkkel történik (PgDn, PgUp, nyílbillentyűk, Home, End, stb.).

Egy parancs kiadható rámutatással, vagy a parancs első betűjének a lenyomásával. Az egér bal gombja az <ENTER>, a jobb gomb pedig az <ESC> megfelelője. Az <ESC> mindig kivisz a parancsból, az F1 a Help képernyőt hozza be.

A fájlokkal végzett műveleteknél az IDCSHELL megvizsgálja, hogy előzőleg kijelöltünk-e állományokat. Ha nem, akkor azzal dolgozik, amelyiken a kijelölő áll. Ha vannak kijelölt állományok, akkor a program megkérdezi, hogy a kijelölt állományokkal dolgozzon-e (Tagged), vagy azzal, amelyiken a kijelölő áll.

A program igen hasznos szolgáltatása a gyors keresés az <ALT-betű> kombinációkkal.

Az IDCSHELL parancsai

- A — ZIP file(s). — ZIP fájl létrehozása, bővítése.
- C — Copy. — Fájl másolása (DOS Copy), lemezek között is lehet.
- D — Directories. — Könyvtárfáról választhatjuk meg az aktuális könyvtárat.
- E — Erase. — Fájl törlése az aktuális könyvtárból.
- F — Find global. — Adott fájl keresése az aktuális meghajtón.
- L — Log drive. — Új meghajtó kijelölése.
- M — Move. — Fájl áthelyezése könyvtárak között, egy lemezen belül.
- N — NARC. — A NARC program meghívása.
- P — Print. — Fájl nyomtatása.
- Q — Quit. — Kilépés az IDCSHELL-ből.
- R — Rename. — DOS fájl átnevezése.
- S — Disk Space. — Szabad lemezterületek lekérdezése.
- T — Tag és <SPACE>. — Fájl kijelölése, illetve a kijelölés törlése.

- V — View file. — ASCII illetve hexa formában belekukkanthatunk az állományokba.
 W — WildTag. — Kijelölés fájlmaszk megadásával.
 Y — File attr. — Fájl attribútumok (hidden, system, readonly, archive) módosítása.
 -, + (G, H). — Könyvtárak közötti mozgás.
 F1 — Help. — Tömör segédinformációk.
 F2 — Select ZIP. — ZIP fájl kiválasztása.
 F3 — ANSI View. — ANSI szekvenciákkal rendelkező szövegek megjelenítése. Csak az első képernyőnyi anyaggal dolgozik.
 F4 — Print Dir. — Könyvtár kinyomtatása.
 F5 — Shell DOS. — DOS-ba lép ki átmenetileg, az EXIT parancs kiadásáig.
 F6 — Tag ALL. — Az könyvtár összes állományát kijelöli.
 F7 — Invert Tags. — Jelölést fordít.
 F8 — Sort Order. — Rendez.
 F9 — Edit File. — Az adott fájlt szerkeszthetjük.
 F10 — Clr ZIP Tags. — ZIP-hez a jelöléseket törli.
 ALT-F1 — File Mask. — Fájlmaszk megadása.
 ALT-F2 — EXECUTE. — Végrehajtja az adott programot.
 ALT-F3 — Stats. — Fájlstatisztika kiírása.
 ALT-F4 — 25/43. — EGA monitoron a 25/43 soros üzemmódok váltása.
 ALT-F9 — Editor. — Új editor megadása.

NARC

A NARC 4.0 program egy Assembly-ben megírt archívkibontó programcsomag, shell (keret) funkciókkal bőven ellátva. Ennek párja a tömörítés és az archívkarbantartás támogatására készült IDCSHELL. A NARC 4.0 verziója már nem csak az ARC és ARK kiterjesztésű archív állományokkal dolgozhat, hanem a ZIP fájlokkal is.

Meg kell jegyezni, hogy a NARC az egyetlen olyan ARC és ZIP kibontó programcsomag, amelybe szervesen bele van építve a kibontáshoz szükséges minden rutin, és ezért nem szorul kisegítő programok segítségére. Ugyanígy az IDCSHELL is egy olyan keretprogramnak tekinthető, amelybe beleépítették a tömörítő rutinokat.

A ZIP formátum előnyeit, lehetőségeit meglehetősen jól ismerik a program alkotói, hiszen a ZIP formátumot, mint az ARC szerkezet utódát az IDC közreműködésével alkotta meg Phil Katz, a PKWare mestere.

A két program azoknak ajánlható, akik egyszerű integrált programcsomaggal akarják elintézni tömörítési-kibontási igényeiket.

A programokat kétféle módon lehet konfigurálni:

- Megvesszük a regisztrált változatokat a szükséges konfiguráló segédprogramokkal.
- Környezetleíró változók segítségével konfiguráljuk.

Mivel Magyarországon szinte kizárólag az utóbbi fordul elő, ezt a konfigurálást ismertetem.

Az IDCSHELL és a NARC környezetleíró változói

NARCPATH=	Hol van a NARC.EXE, teljes névvel megadva.—
NARCHELP=	Hol van a NARCHELP.HLP állomány, teljes névvel megadva.
IDCED=	A használt szövegszerkesztő teljes neve.
IDCTEMP=	A használandó átmeneti munkakönyvtár.
IDCARCPATH=	Hol van az IDCSHELL.EXE, teljes névvel megadva.
IDCHELP=	Hol van az IDCHELP.HLP, teljes névvel megadva.

A szokott módon a DOS SET... parancsával adhatunk értéket nekik, pl. az AUTO-EXEC.BAT-ban: SET NARCPATH=c:\zipp\NARC.EXE

A NARC használata

A NARC bejelentkezésakor kiírja az aktuális könyvtárban lévő tömörített állományok neveit. Ha ezek közül rámutatással választottunk (azaz ráállítottuk a kurzort, majd <ENTER>-t nyomtunk, akkor megjelenik az adott archív tartalomjegyzéke, a képernyő alján pedig a parancssor, és legalul a parancsot részletező megjegyzés sor.

A NARC parancsai

Extract

Kibontja a megadott állományokat. Kibontás előtt megkérdezi, hogy mindegyik kijelölt (tagged) állományt kérjük, vagy csak azt, amelyiken a kijelölő áll. Ezután bekéri a kibontási útvonalat. Egy sima <ENTER>-re az lesz az aktuális könyvtár.

View

Képernyőre bontja ki a megjelölt állományt. Mind ASCII, mind hexa módban használható. Az F kapcsolja be/ki a szűrőt, amivel a különböző szerkesztő-programokkal készült szövegeket nézhetjük meg. Az M és a H váltogatja az ASCII és a hexa kiírást. Az S (Search) paranccsal kereshetünk egy megadott szövegrészt. Az <ESC>, a Q, az X, az F1 és az <ENTER> visszavisz a NARC főmenübe.

Print

Kinyomtatja a kijelölt állományokat. Először is a választott betűtípust kell megadnunk. (0 = A printer betűkészlete, átkapcsolás nélkül. 1 = Pica. 2 = Elite. 3 = Compressed.) Ezután arra kell válaszolnunk, hogy ömlesztve, vagy oldalszámozással, oldalakra formattálva akarjuk-e kinyomtatni az adott állomány(oka)t,

ZIP-wind

Lehetővé teszi másik ZIP fájl választását az aktuális könyvtárban. Az O parancs itt rendezi a kiírást, az S paranccsal pedig egy speciális kiterjesztéssel rendelkező (pl. .EXE ZIP-SFX) állományt is választhatunk.

Chg drive

Másik meghajtót választhatunk aktuálisnak.

DirTree

Grafikusan kiírja az adott lemez könyvtárszerkezetét és innen másik könyvtárt is választhatunk.

Quit

Kilép a NARC-ból, de előtte megkérdezi, hogy az eredeti, vagy az aktuális könyvtárba lépjen-e ki.

A felsoroltakon kívül a K (Kill) paranccsal törölhetjük a kijelölt állomány(oka)t, az F paranccsal pedig kereshetünk.

Összegzésként

Az IDCHELL keretprogramként kiváló, főleg, ha a NARC is mellette van, tömörítőként viszont használhatatlan. A NARC egy igen jól használható program, csak akkor kerül bajba, ha ismeretlen tömörítési módszerrel találkozik.

4ZIP és 4UNZIP

A különféle DOS bővítések rendszerint csak megtűrik a tömörítő programok használatát. Alig néhány van, amelyik kifejezetten támogatja a tömörített adattárolást. (Pl. a PC Tools 6.0 és 7.0 verziói az archív katalógus kiolvasásával, az elterjedtebb tömörítők beépíthetőségével, ezek ismertetése viszont meghaladja e könyv kereteit.)

Érdekes színfolt a DOS bővítések palettáján a 4DOS programcsomag (SolarSoft #383), amely számtalan új funkcióval, paranccsal bővíti a DOS lehetőségeit. A 4DOS csomag ismertetése ugyancsak nem tartozik e könyv témakörébe, de a kiegészítésként hozzá illeszthető 4ZIP és 4UNZIP programok (SolarSoft #511) már igen.

Maga a 4ZIP és a 4UNZIP nem képes a kívánt archiválási feladatokat elvégezni, ehhez természetesen szüksége van a path-on lévő PKZIP, illetve PKUNZIP programokra. A programok megírásának kettős célja volt. Először is, fel akarták ruházni a 4DOS programcsomagot a ZIP fájlok kezelésének lehetőségével. A második bevallott cél pedig az volt, hogy a 4DOS-ban létrehozott katalógus- és fájlkomentek valahogy maradjanak meg a tömörítési folyamatban.

A 4ZIP használata

A 4ZIP kifejezetten a 4DOS bővítés támogatására készült, és normál DOS alatt nem működik kielégítően. Képernyője hasonlít a 4DOS DIR és SELECT parancsaival kapott képernyőkhöz.

Kijelölés:

+	Az adott fájlt kijelöli.
-	Az adott fájl kijelöltségét törli.
<SPACE>	Az adott fájl jelöltségi állapotát átkapcsolja.
*	Az adott könyvtár összes állományát kijelöli.

Mozgás a katalógusban:

<Fel>/<Le>	Egy sorral feljebb/lejjebb visz.
<PgUp>/<PgDn>	Egy képernyővel fel/le görget.
<Home>/<End>	A katalógus elejére/végére visz.
<Enter>	A már kijelölt állományokkal elindítja a PKZIP-et.
<Esc>	Kilép a 4ZIP-ből.

Szintaxis:

4zip ZIPfájl [/am?] [d:][path][fájlspec] [+fájlpec2...]

A 4DOS alatt a dzsóker karakterek használata bővült a normál DOS-hoz képest. A *mos*.doc például az összes DOC kiterjesztésű fájlt kijelöli, melyeknek a nevében valahol szerepel a MOS.

A kapcsolók

Ha nem adunk meg kapcsolót, akkor interaktív módon, a katalógusban való rámutatással választhatjuk ki a nekünk megfelelő állományokat.

Az /a kapcsoló hatására a program rögtön induláskor az összes fájlt kijelöltnek teszi. Az /m kapcsoló a Move lehetőséget adja meg a programnak, azaz a már sikeresen bepakolt állományokat törli a forráskönyvtárból.

A /? kapcsoló a HELP képernyőt hozza elő, amely angolul ismerteti a 4ZIP szintaxisát. A forráskönyvtár megadása elhagyható, ilyenkor annak az alapértelmezése az aktuális könyvtár. Egyszerre csak egyetlen forráskönyvtár definiálható, ami azután az összes felsorolt forrásfájltra érvényes.

A fájlok felsorolásakor a DOS COPY parancsában már megszokott + jeleket kell használnunk a második és a többi fájlnev megadásakor.

A 4UNZIP használata

A 4UNZIP használata nagyon hasonló a 4ZIP-hez. Szintaxisa:

```
4unzip ZIPfájl [/adm?] [cél] [d:][path][fájl] [+fájl2 ...]
```

A kapcsolók:

Ha nem adunk meg kapcsolót, akkor interaktív módon, a katalógusban való rámutatással választhatjuk ki a nekünk megfelelő állományokat.

/a

Az adott ZIPfájlban minden kijelölve.

/m

A kijelölt fájlokat a kibontás után törli is, ez egy kifelé működő áthelyező (move) parancsnak felel meg.

/d

A kijelölt állományokat törli a ZIP fájlból.

Az /a és az /m kombinálható (/am), az /md azonban egyenértékű az /m kapcsolóval.

A /? kapcsoló a Help képernyőt hozza elő, amely angolul ismerteti a 4UNZIP szintaxisát.

Ha nem adunk meg célkönyvtárat, akkor az aktuális könyvtárba pakoltat ki a PKUNZIP-pel.

SHEZ — a tömörítők Norton Commandere

A keretprogramok leginkább kényelemszeretűnknek és lustaságunknak köszönhetik karrierjüket. Feladatuk elsősorban az, hogy ne kelljen fejben tartani a különböző programok használatának, paraméterezésének néha teljesen logikátlan szabályait, s hogy minél kevesebb fájlnevet és elérési útvonalat kelljen kézzel bepötyögni, ami köztudottan nemcsak fárasztó, de igen gyakori hibaforrás is.

A SHEZ ezért válhatott kedvelt segítőtársunkká. Mi is az, ami rávesz bennünket arra, hogy majdnem 200 K értékes területet áldozzunk winchesterünk kapacitásából egy segédprogramra? Íme néhány tény:

— ARC, PAK, ARJ, ZIP, ZOO, LZH állományokat felépít, bővít, módosít, kibont, kommentez.

— SFX-et készít LZH, ARJ, ZIP, PAK és ARC állományokból.

— A fenti archív és SFX állományok között tetszőlegesen tud konvertálni.

— Futtatni is tudja a fenti archív és SFX állományokban tárolt programokat.

— Épségellenőrzést végezhet és vírusfelderítést a McAfee féle SCAN programmal.

— Egyik archívból a másikba át tud vinni tömörített állományokat.

— Belekukkanthatunk vele a bepakolt állományokba, sőt módosíthatjuk is azokat a konfiguráláskor megadott szövegszerkesztő programmal.

— S mindehhez nem kell megtanulni a különféle tömörítők használatát, mert a paraméterezést elvégzi helyettünk a SHEZ.

— A program Help-je sem rossz, ha elakadnánk.

— Egérrel is használhatjuk.

Rögtön le kell azonban szögezni, hogy a SHEZ nem csodaszer, csak egy keretként szolgáló, meglehetősen intelligens segédprogram. Akkor tud igazán segíteni, ha megfelelően ellátjuk kiegészítő programokkal és információval, s főleg ha tudjuk, mit is akarunk tenni vele. Ehhez konfigurálni kell a SHEZCFG.EXE programmal, amelyet a SHEZ-zel egy csomagban terjesztenek. Képességeinek életrekeltségéhez szükség van az alábbi programokhoz való, gyakran intenzív hozzáférésre:

LHA.EXE, vagy LHARC.EXE az LZH állományokhoz (2.12-ig). (SHEZ 5.9-ig LHARC, 6.1-től LHA).

PAK.EXE a PAK állományokhoz (2.50-2.51).

ZOO.EXE a ZOO állományokhoz (2.01).

ARJ.EXE az ARJ állományokhoz (2.10-ig). (A SHEZ 6.1 verziótól kezdve).

PKARC/PKPAK, PKXARC/PKUNPAK, PKSFX.PGM az ARC állományokhoz (3.60-3.61).

PKZIP, PKUNZIP, ZIP2EXE a ZIP állományokhoz (1.10-ig).

SHEZ.EXE a vírusellenőrzéshez (McAfee).

LIST.COM a beleolvasáshoz (7.2 fölött).

NE, PE vagy egyéb szövegszerkesztő a szerkesztéshez.

A SHEZ az alábbi állományokban terjed:

SHEZ.EXE Maga a program.

SHEZ.DOC Az eredeti angol nyelvű programleírás.

SHEZ.HLP A SHEZ Help szövegeit tartalmazza.

SHEZ.REG Regisztrációs formaszöveg ASCII fájlban.

SHEZ.ICO Windows ikon.

SHEZ2.ICO Windows ikon.

SHEZ.REV A SHEZ egyes verzióinak változásai.

SHEZ.REF Táblázat a SHEZ billentyűkombinációiról.

• OMBUDSMN.ASP Tájékoztató.

SHEZCFG.EXE Konfiguráló segédprogram.

• README.NOW Friss információ az új verzióról.

SWAPER.DOC A SHEZ swap lehetőségeiről néhány sor.

A működéséhez szükséges információkat a SHEZ több helyen is tárolhatja. Magában a SHEZ.EXE állományban, aminek aktualizálását a SHEZCFG.EXE programmal végeztük el, illetve a SHEZ.CFG állományban. Ez utóbbit a SHEZ-ből munka közben, a pillanatnyi konfigurációt rögzítve lehet létrehozni, illetve módosítani az <ALT w> billentyűkombinációval. A fentiekén kívül még a környezeti változókból is vehet konfigurációs adatokat.

A SHEZ konfigurálása

Annak is érdemes elolvasni ezt a részt, aki már konfigurálta a SHEZ-t, mert valami apró dolog mindig elkerülheti a figyelmünket.

A program konfigurálása időben is elválasztható négy külön fázisból áll:

— A SHEZ állományainak a bemásolása egy path-on lévő könyvtárba.

— A SHEZ.EXE konfigurálása a SHEZCFG.EXE programmal.

— A SHEZ aktuális paramétereinek rögzítése egy SHEZ.CFG állományban az <ALT w> billentyűkombinációval. (A 6.2 verziótól ezt már maga a SHEZCFG program is elvégezheti.)

— A megfelelő környezeti változók beállításának beépítése az AUTOEXEC.BAT-ba, vagy egyéb batch-fájlba.

SHEZCFG.EXE

Aki a SHEZ-t igazából használni akarja, annak legalább egyszer le kell futtatnia a SHEZCFG programot. Enélkül a SHEZ nem lesz tisztában azzal, hogy milyen monitorra dolgozzon, és hol találja meg a szükséges tömörítő, kibontó és egyéb segédprogramokat, valamint a gyári alapbeállításokkal sem lesz tisztában.

A program indítása után legelőször azt kérdezi meg, hogy hol keresse a konfigurálandó SHEZ.EXE állományt. Itt be kell írni a meghajtót, az elérési útvonalat és a teljes nevet is, az .EXE kiterjesztéssel együtt, a DOS-ban megszokott módon (pl: C:\ZIP\SHEZ.EXE).

A SHEZ CONFIG MENU az alábbi választási lehetőségeket kínálja fel:

Color setup = Színbeállítás.

Programs setup = Segédprogramok adatait módosíthatjuk.

Additional setup information = A SHEZ-t paraméterezi tovább.

Save Configuration info. = Kiírja a konfigurációt és kilép.

Integrity check of shez program = Teszteli a SHEZ épségét.

Registration number installation = Regisztrációs szám megadása.

A választott menüpont inverz betűjét kell lenyomni, illetve kilépéshez (mentés nélkül) az <ESC> gombot.

Színbeállítás

Default color set	= Gyári beállítás.
Custom color set	= A színek külön-külön is beállíthatók.
Black & White color set	= Fekete-fehér monitorhoz.
Monochrome set	= Mono kártyához.

A „Default”, a „Monochrome”, vagy a „Black & White set”-et választva a program udvariasan közli, hogy további színmódosításokhoz rendelkezésünkre áll a „Custom color set”.

A „Custom color set” választásakor a program egyenként bekéri a SHEZ által használt ablakokhoz használandó színeket. Nem kell a színek nevével vagy kódjával vesződni, a háttérszín a balra-jobbra nyilakkal, a betű- és keretszín a fel-le nyilakkal állíthatjuk. Közben mindig látható egy teszt ablak, amit az éppen választott színekkel rajzol ki a program. Ha egy ablak kész, akkor az <ENTER> visz a soron következő ablakra. Az <ESC> billentyűvel változtatás nélkül kiléphetünk innen. Az alábbi ablakokat lehet „kiszínezni”:

Main window header	A főablak fejléce.
Highlite bar	Kijelölő sáv.
Function key line	Alsó funkciógombok sora.
Function key highlight	Kiválasztott funkciógombok.
Menu bar and window	Menü sor és ablak.
Menu highlight	Kiválasztott menüpont.
Status window	Státusz ablak.
Filelist window1	Az 1. fájllista ablak.
Filelist window2	Az 2. fájllista ablak.
Data input field	Adatbeviteli mező.
Help window	Help ablak.

Tippek:

— Színes kártyával és mono monitorral dolgozók figyelmébe: először a Black & White, majd a Custom color set választását javaslom.

— Minden színbeállításnál a Custom color set segítségével ellenőrizzük, hogy minden ablakban jól olvasható-e a SHEZ képernyője.

Segédprogramok beállítása

A Program setup választásával beadhatjuk a hozzáférhető tömörítő és egyéb segédprogramok leőhelyeit és verziószámait.

Először a ZIP állományokhoz kér információt a program:

- Akarunk-e ZIP állománnyal dolgozni (Y/N).
- Hogyan találja meg a PKZIP-et (pl. C:\ZIP\PKZIP.EXE).
- A PKZIP verziószáma (pl. 110).
- Hogyan találja meg a PKUNZIP-et (pl. C:\ZIP\PKUNZIP.EXE).
- Akarunk-e ARC állománnyal dolgozni (Y/N).
- Az ARC kipakoló programunk leőhelyét kell megadnunk. Jelen verzió a PKXARC és a PKUNPAK programokat támogatja (pl. C:\ZIP\PKUNPAK.EXE)

— Az ARC bepakoló programunk lelőhelyét kell megadnunk. Jelen verzió a PKARC és a PKPAK programokat támogatja (pl. C:\ZIP\PKPAK.EXE).

— Akarunk-e LZH, ARJ, ZOO, PAK állománnyal dolgozni (Y/N).

— Hogyan találja meg pl. az LHARC/LHA aktuális verzióját? (Pl. C:\ZIP\LHA.EXE)

— Mi a program verziószáma (Pl. 212).

— Az ideiglenesen kikapolt állományok módosítására, szerkesztésére melyik editort választja. Javasolom mindenkinek, hogy Norton Editort, vagy hasonló kisméretű programot használjon. (C:\NORTONNE.COM).

— Az editornak induláskor átadandó paraméterek.

— Hol van az a program, amelyikkel végleges kikapolás nélkül belekukkanthatunk a tömörített állomány szövegfájljaiba (pl. C:\UTIL\LIST.COM). A Vernon D. Buerg féle LIST.COM a legjobb erre a célra. A LIST.COM 7.5e verzió megtalálható többek között a SolarSoft programkönyvtár #432 számú lemezén.

— Igénybe akarjuk-e venni a program víruskereső lehetőségeit (Y/N).

— Hol van a McAfee féle SCAN.EXE program. Teljes név megadása szükséges (pl. D:\AV\SCAN.EXE).

Ezután visszajutunk a SHEZ CONFIG MENU-be.

Belső paraméterezés

Az „Additional program setup” választása után az egyes adatbeviteli ablakok kitöltésével sorban közölhetjük a programmal a működési paramétereket:

— Kibontáskor mi a célkönyvtár alapértelmezése.

— A View (megnézés) és a Run (program futtatása archívból) funkciók használatakor átmenetileg melyik lemezre nyissa ki a választott állományt. Az aktuális lemezt az X beírásával választhatjuk, de célszerűbb a ramdrive (általában D), vagy a winchester minél nagyobb szabad területtel rendelkező partíciójának megadása.

— A kilistázott katalógust ne rendezze (X), illetve név (N), kiterjesztés (E) vagy dátum (D) szerint rendezze.

— Az archívok tartalomjegyzékét ne rendezze (X), illetve név (N), kiterjesztés (E) vagy dátum (D) szerint rendezze.

— Ha paraméter nélkül indítjuk a SHEZ-t, mit listázzon. Megadhatunk teljes nevet (drive:\path\ fájl.kit), illetve használhatunk dzsóker karaktereket. A SHEZ különleges dzsóker karaktere a \$. A \$. \$ az összes archív állományt (lásd alább) kilistázza. Ha nem adunk meg path-t, akkor az aktuális könyvtárból listáz mindent (*.*).

— Mit jelentsen a fent említett \$. \$. Az alapértelmezés szerint a SHEZ a ZIP, ARC, LZH, PAK, ARJ, ZOO állományokat tekinti archívoknak. Ha ehhez hozzáírjuk még az EXE és COM kiterjesztéseket, akkor a program az SFX-eket is látja. A kiterjesztés előtt a pontot is ki kell tenni, a kiterjesztéseket egymástól pedig vesszővel választjuk el (pl. .ZIP,.ARC,.LZH,.ZOO,.PAK,.SDN,.ARJ,.EXE,.COM).

— Hány soros legyen a kijelzés: 25 (= CGA, Hercules), 43 (= EGA), 50 (= VGA). Az utóbbi két számhoz EGA, illetve VGA kártya kell. Az alapértelmezés 25.

— Hány állományt kezelhet a SHEZ egy tömörített állományban, illetve egy könyvtárban. A kérdés mellett a lehetséges maximális érték is fel van tüntetve. 500 már jól használható.

— Gyorsan kinyíló ablakot akarunk-e (Y/N)? Ez egyes lassú, főleg CGA jellegű kártyáknál érdekes.

— BIOS hívásokkal írjon-e a képernyőre (Y/N)? Ez egyes nem teljesen kompatibilis gépeken fontos. Az én gépemén gyorsabb a SHEZ képernyőkezelése, ha ide N kerül.

— Mono monitort használunk-e (Y/N)? Ha igen, célszerű itt jelezni, mert olvashatóbb lesz a kép.

— Ki akarjuk-e használni azt a PKZIP 1.10 használata esetén rendelkezésünkre álló alternatívát, hogy a sebesség, vagy a tömörség fontosabb számunkra. Esetleg az imploding technikát akarjuk az SFXJR állományok miatt. A válasz itt is Y vagy N.

— A PKZIP használata esetén (ha az 1.10 verzió van a gépben) mi legyen az extra kompressziós faktor alapértelmezése? Az S a sebességet (Speed), a Z a tömörséget (siZe) optimalizálja, az I pedig az imploding technika használatát erőlteti. (s/z/i)

— Ha nem hozzáférhető az EMS memória, vagy nem elég, vagy nincs is (sajnos mindhárom gyakori dolog), akkor hová írja ki átmenetileg a memória tartalmát. Könyvtárnevet kell megadnunk (pl. C:\TEMP).

— Használhatja-e a program az EMS memóriát a Swap funkcióra? A válasz Y vagy N lehet.

— Használhatja-e a program a beépített memórialapozásos technikát (Y/N).

— A segédprogramok futtatásakor használja-e a fent említett memórialapozásos technikát (Y/N).

— Rákérdezz-e a célkönyvtárban már létező fájl felülírására (Y/N).

— Használunk-e egeret (Y/N).

— Engedélyezzük-e a DOS-ba való közvetlen kilépést (Y/N).

— Kiírja-e a célkönyvtárat a képernyőre (Y/N).

— Melyik legyen a tömörítő programok munkalemeze. Az aktuális könyvtár lemeze = X. Ajánlatos a ramdrive, vagy a winchester (legtöbbször C) megadása. Kettőspont nem kell.

— Kilépéskor a program visszaállítsa-e a SHEZ behívása előtti képernyőt (Y/N).

— Az egér érzékenysége állítható 0 és 200 között. Az általában használatos érték 50.

— A SHEZ.CFG állományt is egyből frissítsük-e (Y/N).

A konfigurációs adatok kiírása a SHEZ.EXE állományba:

A SHEZ CONFIG MENU „Save configuration info” pontját választva a SHEZCFG kiírja a beállított értékeket a SHEZ főállományába, majd befejezi működését, és kilép a DOS-ba.

A SHEZ.EXE épségének ellenőrzését végzi az „Integrity check of shez program” menüpont. Ha ezt választjuk, akkor a program a CRC értékek segítségével teszteli a SHEZ.EXE épségét. Tesztelés közben a képernyő bal felső részén a felirat alatt egy kis jel pörög, ez jelzi a program működését.

A „Registration number installation” pontot csak kevesen használják Magyarországon, mivel nálunk közismert okok miatt rendkívül kevés példány regisztrálására jut pénz. Ezt a menüpontot választva beírhatjuk regisztrációs számunkat, és utána indításkor a program már nem vár billentyűleütésre. Ha azt akarjuk, hogy regisztrált legyen a program, hívjuk meg ezt a menüpontot és töltsük fel a kinyíló ablakot egyesekkel. Ez segít! A SHEZ 6.x regisztrált állapotban nem áll meg egy billentyűleütésre várva az indítás után. A többi előnyt csak a profi SHEZ alkalmazók észlelik csak.

Ha mindent végigvettünk, akkor két módon szállhatunk ki a konfigurációs programból: a „Save configuration info” menüpontot választva a beállított paraméterek mentésével, illetve az <ESC> billentyűvel mentés nélkül. Ez utóbbi esetben még egyszer rákérdez a program, nem akarjuk-e mégiscsak elmenteni az éppen beállított paramétereket.

A SHEZ 6.2 környezeti változói

A SHEZ a működéséhez szükséges adatokat, amelyeket alapértelmezésnek vesz, több helyről is veheti:

— Először is megkeresi a SHEZ.CFG állományt, és abból olvas. Ez azért is fontos, mert hálózati alkalmazásnál minden felhasználónak saját SHEZ.CFG-je lehet.

— Ha azt nem találja, vagy nincs benne a szükséges adat, akkor az alább ismertetendő környezeti változókat (environmental variables) keresi meg.

— Ha nem találja a környezeti változókat, akkor a SHEZ.EXE-ben tárolt információt használja.

A környezeti változók közül a SHEZCFG-nek érdemes az AUTOEXEC.BAT-ban értéket adni, a többit pedig a SHEZ-t indító batch-fájlban helyezzük el. Persze mind betehető az AUTOEXEC.BAT-ba.

BIOS — Ha a változónak adunk értéket (SET BIOS=y), akkor a program a BIOS-on keresztül ír a képernyőre.

MONO — Ha a változónak adtunk értéket (SET MONO=y), akkor a program monokróm módban fut.

SHEZEX — A tömörített állományok kibontása az itt is megadható lemezre és könyvtárba történik, hacsak a programból meg nem változtatjuk.

SHEZWORK — A SHEZ által használt átmeneti munkakönyvtár megadására szolgál ez a változó. Kilépéskor a SHEZ teljesen kitakarítja a könyvtárat maga után, a rejtett és rendszerállományokat is törli belőle, de magát a könyvtárat nem. Hálózaton hasznos, mert nem szemetel maga alá.

SHEZCMT — Azt az alapértelmezésként használt fájlt adhatjuk meg, amelyikben a kommenteket tároljuk. A kommentek bővítése, frissítése esetén a SHEZ-en keresztül az adott tömörítő program innen veszi a kommentet (pl. a PAK, a ZOO vagy az ARJ).

SHEZCFG — A SHEZ.CFG konfigurációs állomány teljes nevét kell beleírni (pl. SET SHEZCFG=D:\ZIP\SHEZ.CFG).

SHEZEV — A SHEZ-ben megvan az a lehetőség, hogy nemcsak a konfiguráláskor megadott fájl-betekintőt használhatja, hanem az adott állományra állva a <CTRL ENTER>-rel az itt megadható kukkantó programot hívja meg. Maga a SHEZEV az általános kukkantó-pótló, a SHEZEV.XXX változóknál pedig a megadott kiterjesztésű állományokhoz használandó segédprogramot definiálhatjuk a megfelelő paraméterezéssel. Például:

```
SHEZEV=C:\UTIL\FVIEW $F /43
```

```
SHEZEV.GIF=VGIF $F /VGA
```

```
SHEZEV.PIC=VPIC $F /P
```

A \$F a SHEZ által átadott fájlnev helyén áll. A kiértékelés sorrendje a <CTRL ENTER> lenyomása után: SHEZEV.XXX > SHEZEV > SHEZ.CFG alapján > SHEZ.EXE konfigurált alapértelmezés.

Egerészés a SHEZ-ben

A SHEZ-nek nincs külön egérkurzora. Az egér mozgatása megfelel a nyílbillentyűknek.

Az egér bal szeme az <ENTER>, jobb szeme az <ESC> billentyűnek felel meg. Az egermenüt az egér baloldali gombjával kétszer rákoppintva (háromgombos egereknél a középső gombbal) hozhatjuk elő. A SHEZ-nek két egermenüje van, annak megfelelően, hogy a DOS ablakban a DOS állományokat, vagy pedig egy tömörített állomány belsejét nézzük. A két egermenünek közös menüpontjai is vannak:

SHOW MENU — Mindkét esetben a / billentyű lenyomásának felel meg, azaz a képernyő tetején aktivizálódik a menüsor.

HELP — Mindkét esetben az F1 billentyű lenyomásának felel meg, azaz a képernyő közepén kinyílik a Help ablak.

EXIT SHEZ — Kilépés a SHEZ-ből az aktuális könyvtárba. Megfelel az <ALT q> lenyomásának.

+SENSITIVITY — Az egér érzékenységet növeli (gyorsítja az egeret).

-SENSITIVITY — Az egér érzékenységet csökkenti (lassítja az egeret).

Menüpontok a DOS ablakból

View file — A kiválasztott archív archívkatalógusát hozza be.

Convert file — Átkonvertálhatjuk az adott archívot egy más típusúra. Ha csak frissíteni akarjuk, az is megengedett, ilyenkor átcsomagolja az adott tömörítő régebbi verziójával készült állományokat. A célarchív típusát egy hirtelen előbukkanó almenüből választhatjuk ki.

Make SFX — Az adott állományból a megfelelő segédprogram(ok) felhasználásával önkibontó SFX-et készít. A ZOO-nál ezt nem támogatja. Ha PAK állományból készítünk SFX-et, akkor a forrás PAK fájl törlődik. Ez a PAK program miatt van.

Copy file — Az adott állományt másolja át. Másolás előtt egy ablakba be kell írni a célkönyvtár nevét (és az új állomány nevét, ha az aktuális könyvtárba akarjuk másolni).

Move file — Mint a fenti Copy file, csak a sikeres átmásolás után törli a forrásállományt.

Rename file — A kijelölt fájl átnevezésére szolgál.

Delete file — A kijelölt állományt törli.

Convert tagged — A kijelölt állományok csoportját átalakítja a menüpont megadásakor kinyíló almenüből választható formátumúra.

Make tagged SFX — A kijelölt állományok csoportjából SFX-eket készít.

Copy tagged — A kijelölt állományok csoportját átmásolja.

Move tagged — A kijelölt állományok csoportját áthelyezi.

Rename tagged — A kijelölt állományok csoportját átnevezi.

Delete tagged — A kijelölt állományok csoportját törli.

Menüpontok az archívkatalógus ablakából

View file — A tömörített fájl megnézése a konfiguráláskor megadott kukkantó programmal (LIST.COM).

Extract file — A kijelölt állományt kibontja az éppen érvényes kibontási útra.

Delete file — Törli a kijelölt állományt az archívból.

Add file — Bővíti az archívot.

Edit file — A kijelölt állományt átmenetileg kinyitja a pillanatnyilag érvényes munkakönyvtárba, majd meghívja hozzá a konfiguráláskor megadott szerkesztő programot. A szerkesztés befejezte után az aktualizált állománnyal felfrissíti az archívot is.

Execute file — A kijelölt állományt átmenetileg kinyitja a pillanatnyilag érvényes munkakönyvtárba, majd elindítja. A program befejezése után törli az átmeneti könyvtárból.

Convert from SFX — SFX-ből normál archívvá konvertál.

Extract All — Az archív összes állományát kibontja az éppen érvényes kibontási útra.

Extract tagged — A kijelölt állományok csoportját kibontja az éppen érvényes kibontási útra.

Delete tagged — Törli a kijelölt állományok csoportját az archívból.

Spec. extract path — Kibontási útvonalat adhatunk meg. Újabb módosításig a SHEZ ezt használja.

Az egeret állományok kijelölésére is használhatjuk: a két (szélső) gomb együttes lenyomásával jelölhetünk ki állományokat. Ha a gombokat lenyomva tartjuk és közben az egeret mozgatjuk, meggyorsíthatjuk a kijelölést.

Amikor a SHEZ Yes, vagy No választ vár, akkor az egér bal szeme a Yes, a jobb szeme a No választ adja.

A LIST.COM egeresítését lásd a LIST ismertetésénél.

A SHEZ használatának módja

A SHEZ-t többféleképpen is meghívhatjuk:

— Paraméter nélkül (SHEZ) A bejelentkezéskor aktuális könyvtár alkönyvtárait és a benne lévő, a SHEZ konfigurációs adatainak megfelelő archívokat láthatjuk a file list ablakban.

— Könyvtármegadással (SHEZ c:\zipped\) Az adott könyvtár tömörített állományait láthatjuk.

— Archív nevének megadásával (SHEZ c:\zipped\munka.zip) A megadott állományt láthatjuk belülről, azaz az archív katalógust mutatja a SHEZ. (ARJ és ZOO SFX-et nem !!)

— Dzsóker karakterekkel (SHEZ *.* Z?H*.\$, SHEZ c:\zipped\$. \$)

A 6.x verziótól kezdve a nem regisztrált példányok egy billentyű lenyomására várnak. A regisztrált változatok nem állnak meg itt.

A SHEZ, mint a legtöbb program, nemcsak menükkel és egérrel vezérelhető, de funkciói nagy részét megadott billentyűkhöz, billentyűkombinációkhoz rendelték. Ezek természetesen függenek attól, hogy mit mutat meg éppen a SHEZ: a normális DOS könyvtárat az állományokkal és alkönyvtárakkal, vagy egy tömörített állomány belsejét. A két esetben — amint az eltérő feladatokból is levezethető —, más-más parancsok vannak mind a menükben, almenükben, mind pedig az egyes billentyűkhöz rendelve.

A SHEZ használata menükkel

A DOS ablak menürendszere

A DOS ablak tetejének bal szélén egy betű jelzi, ha a katalógus valamiféle rendezéssel került kiírásra. A rendezés az F6 billentyűvel, vagy a Misc menüből kérhető.

A menü a / lenyomására ugrik elő, az <ESC> visz vissza belőle. A DOS ablak menüjének öt pontja van: View, Convert, Misc, File és Quit.

View — Arra szolgál, hogy belenézhesünk az éppen kiválasztott állományba. Almenüje két választási lehetőséget ad:

— Megnézhetjük a szokott módon, a LIST.COM segítségével.

— Megnézhetjük alternatív kukkantó programmal. Ennek adatait a SHEZ a megfelelő környezeti változóknak keresi.

Convert — A kijelölt archív, illetve archív csoportok átkonvertálását kérhetjük itt. Ha nem SFX-et akarunk, akkor meg kell adni a célarchív típusát is egy almenüből. Alkalmazható arra is, hogy ugyanolyan archív formátumban újratömörítsünk állományokat.

Convert highlighted file

= A kijelölt állományt átpakolja.

Convert all tagged files

= Az összes kijelölt állományt átpakolja.

CVT highlighted to SFX = A kijelölt archívból SFX-et készít.
CVT all tagged to SFX = A választott archívokból SFX-eket készít.

Misc — Ez a menüpont a vegyesfelvágott. Innen adhatunk meg kibontási utat, kérhetünk másik könyvtárat, fájlstatistikát a kijelölt állományról. Átrendezhetünk, új archívot hozhatunk létre, kérhetjük a könyvtár újraolvasását, kijelölhetünk, törölhetjük a kijelölést. Innen léphetünk ki átmenetileg, az EXIT parancs kiadásáig a DOS-ba. A konfigurációs fájl innen is frissíthető. Az archívokhoz kommentet innen adhatunk és a vírusvadászat is innen indul.

File — Ez alatt a menüpont alatt kaptak helyet a DOS Shell funkciókat támogató parancsok: másolás, áthelyezés, átnevezés, törlés, szerkesztés. Az egyes parancsok mellett látható a megfelelő billentyűkombináció is.

Quit — Kilépés. Kiléphetünk a menüből, de elhagyhatjuk a SHEZ programot is, visszatérve akár az aktuális könyvtárba, akár a kibontási célkönyvtárba.

Az archív katalógusablak menürendszere *Archív Eszköz +/*

Extract — A kibontással kapcsolatos parancsok kerültek ide.

Update — Az archív bővítése és frissítése esetén választhatunk, hogy listából, vagy promptozva akarunk dolgozni. Ez utóbbi esetben sem kell beírni a fájlneveket, hanem a listából egyszerűen kijelölhetjük azokat.

Delete — Ennek a menüpontnak a segítségével törölhetünk állományokat az archívból. A törölt állományok csak akkor tűnnek el a listából, ha az <ALT R> billentyűvel újraolvasztatjuk a könyvtárat.

Misc — Ez a vegyesfelvágott menü. Innen lehet tesztelni, megnézni, szerkeszteni, futtatni, kommentezni, kommenteket megnézni, keresni, maszkot adni, vírust ellenőrizni, jelszózni, stb. Az egyes parancsok mellett fel van tüntetve a megfelelő billentyűkód is. Részletesen lásd ott.

Other — Innen is létrehozhatunk új archívot, választhatunk egy másik aktuális könyvtárat, illetve a PKZIP használata esetén választhatunk a tömörítési technikák között.

Print — A PKZIP nyomtatási lehetőségeit használja ki ez a menüpont. Innen megválaszthatjuk, hogy nyomtatáskor melyik portra (COM1, COM2, LPT1, LPT2, ...) és milyen üzemmódban akarunk nyomtatni (text/grafikus).

Convert — SFX-ből normál archívot csinál PAK és ARJ SFX esetén.

Tag — Kijelölés.

Quit — Kilépés. Kiléphetünk csupán a menüből, vagy a SHEZ-ből is, akár az aktuális könyvtárba, akár a kibontási célkönyvtárba.

A SHEZ billentyűre kiosztott funkciók

Kiválasztás, kijelölés

+ — Az adott állományt kiválasztottnak jelöli.

— — Az adott fájl kijelöltségét megszünteti.

<SPACE> — Az adott fájl kijelöltségét átkapcsolja.

<CTRL SPACE> — A könyvtár, illetve az archív összes állományát kijelöli.

<CTRL U> — Az összes kijelölt állomány kijelöltségét megszünteti.

Archív állomány bővítése

A — A kijelölt állomány(oka)t hozzáadja az archívhoz.

<ALT C> — A kijelölt állományhoz fájlkommentet ad, illetve aktualizálja azt.

S — A kijelölt állomány(oka)t kommentezi, illetve aktualizálja a kommenteket.

<ALT S> — Az összes állományt kommentezi.

C, F7 — Új archív létrehozását végzi. Bekéri az archív típusát (menüvel), nevét, forrásállományait.

<CTRL Y> — Az aktuális archív kijelölt állományait egy másik archívba másolja át.

M — A kijelölt állomány(oka)t bepakolja egy archívba, majd a sikeres bepakolás után törli a forrásállományokat.

<CTRL F3> — A kijelölt állományt áthelyezi egy másik könyvtárba.

F — Az aktuális archív állományait frissíti.

U — Az aktuális archív állományait aktualizálja.

<CTRL T> — Frissíti az archív állományainak dátum és időpont bejegyzéseit.

G — Jelszót adhatunk meg az archív kezeléséhez.

<ALT A> — Az archívba bepakolandó állományok listáját készíti el.

<ALT F> — A frissíteni kívánt archívok listáját készíti el.

<ALT M> — Az áthelyezni kívánt állományok listáját készíti el.

<ALT U> — A frissíteni, bevinni tervezett állományok listáját készíti el.

Kibontás archívból

E — Az éppen kijelölt állományt kibontja az archívból.

<ALT X> — Mindent kipakol az archívból.

<CTRL E> — Az összes kiválasztott állományt kibontja az archívból.

<ALT N> — Minden állományt kibont, ami frissebb, mint a célkönyvtárban lévő azonos nevű állomány.

T — Ellenőrzi az adott archív épségét a CRC értékek alapján.

O — Felülírás be/kikapcsolása.

V, <F3>, <ENTER> — Az éppen kijelölt fájl megnézése.

<ALT V> — Filekommentek megtekintése.

<F8> — A választott fájl szerkesztéséhez kell ezt a billentyűt lenyomnunk. Ilyenkor a SHEZ egy átmeneti munkakönyvtárban kinyitja a megadott állományt, majd azt a konfiguráláskor megadott szerkesztő programmal tetszésünk szerint módosíthatjuk. A munka befejeztével a SHEZ felfrissíti az adott állományt, majd törli az átmeneti fájlt.

<CTRL ENTER> — Az adott állományt az alternatív kukkantó programmal mutatja meg.

R — A választott programot futtatja. Csak EXE, COM és BAT állományok! A gyakorlatban ez úgy néz ki, hogy átmenetileg kibontja, elindítja, majd a futás után törli a futtatásra kijelölt programot.

Konvertálás

<ALT V> — A kijelölt archívot átcsomagolja. A célarchív típusa menüből választható. A forrás— és a célarchív lehet azonos típusú is.

<CTRL V> — A kijelölt archívokat átcsomagolja. A célarchív típusa menüből választható. A forrás- és a célarchív lehet azonos típusú is.

<F8> — A kijelölt archívból SFX-et csinál. A PAK forrásarchív mindenképpen törlődik, a többinél választhatunk.

<SHIFT F8> — A kijelölt archívokból SFX-et csinál. A PAK forrásarchívok mindenképpen törlődnek, a többinél választhatunk.

Az archív állomány karbantartása

D — Törli az archívból azt az állományt, amelyiken a kurzor áll.

<ALT D> — Kitöröl minden kibontott állományt.

<CTRL D> — Törli az archívból a kijelölt állományokat.

Z — Archívfájl maszkot adhatunk meg vele.

I — Az adott archív statisztikai adatait írja ki.

<F9> — Rendezi az archívkatalógus kilistázását.

<ALT S> — Gyorskeresés.

<F2> — Másik archív választása.

P — A path-t akarjuk-e tárolni, illetve figyelembe akarjuk-e venni kibontáskor az eltárolt path-t.

DOS Shell funkciók

<CTRL F2> — Az éppen kijelölt fájl másolására szolgál.

<SHIFT F2> — Előzetesen kijelölt állományok másolására szolgál.

<CTRL F3> — A kijelölt állományt áthelyezi egy másik könyvtárba.

<SHIFT F3> — A kijelölt állományokat áthelyezi egy másik könyvtárba.

<F4> — A kijelölt állományt törli.

<SHIFT F4> — A kijelölt állományokat törli.

<F5> — Fájlstatisztikát ír ki.

<F6> — Rendezve írja ki a fájllistát.

<CTRL F6> — A kijelölt fájl átnevezésére való.

<SHIFT F6> — Az előzetesen kijelölt állományokat nevezhetjük át.

<ALT D> — A fájlok dátumadatait írja ki a DOS ablak fájllistájában.

<ALT S> — A fájlok méretadatait írja ki a DOS ablak fájllistájában.

Egyéb funkciók

<F1> — Bárhonnan Help-et ad.

<ALT Q> — Azonnali kilépés a SHEZ-ből.

<CTRL X> — Azonnali kilépés a SHEZ-ből a célkönyvtárba.

<ALT R> — A parancs hatására újraolvassa az aktuális könyvtárat.

<ALT W> — Felfrissíti a konfigurációs fájlt az éppen aktuális beállításokkal.

<ALT 1> — A ZIP archívok használata esetén a PKZIP-et -! paranccsal hívja meg.

<ALT P> — A kibontás célkönyvtárát adhatjuk meg.

/ — Meghívja a menüt.

<F10> — A DOS-ba lép ki, de csak átmenetileg. EXIT-tel visszajutunk ugyanoda, ahol voltunk.

A tömörítés és a vírusok

Mint tudjuk, a fájlokba beépülő vírusok többsége az EXE és COM állományokat támadja meg. (Néha a SYS, BIN, OV? kiterjesztésűeket is) Ez ellen legtöbbször hatásos védekezési mód a tömörített archív állományokban való programtárolás.

Ez a védekezési megoldás arra épít, hogy a vírusok zöme olyan állományokba másolja be saját kódját, amelyek alkalmanként átveszik a gép vezérlését. Ez a védekezés sem tökéletes azonban, hiszen a vírusok (sajnos) szintén fejlődnek. Az Alaplap 1991. májusi száma jelezte, hogy feltűntek az első olyan vírusok, amelyek adatállományokba is beleírnak, így például DBF és ARC (!) állományokba.

Megelőzés

A tömörítőprogramok mindegyike rendelkezik a Test (= épségellenőrzés) paranccsal, ez azonban már csak utólagos, nem pedig megelőző jellegű lépés, amire igen nagy szükség lenne. Az esetleg megtámadott archív állományokból az érintetlennek bizonyult részeket kibonthatjuk, ha az archív katalógusa még épen maradt. A ZOO esetén az archívkatalógus elvesztése sem jelent visszahozhatatlan veszteséget, mivel a FIZ segédprogram nem onnan indítja az ellenőrzést.

A PKZIP regisztrált változatának használói előnyben vannak az egyéb tömörítő programokkal dolgozókkal szemben, mivel a ! parancs alkalmazásával bepakolt állományokat a PKUNZIP automatikusan teszteli, és a bepakolás óta megváltozott tartalmú állományoknál figyelmeztet a módosulás tényére. A PKZIP fenti funkciójának életre keltéséhez a következőt kell megtennünk:

— Regisztráltatni kell az akárhonnan megszerzett PKZIP-et, a csomagban megtalálható megrendelő nyomtatványt kitöltve és elküldve az ott megadott címre (a regisztrálási díj 25 USD, illetve 47 USD a következő verzió megküldésével együtt), majd

— A PKWare cégtől megkapott azonosító kódunkat a csomagban található PUTAV.EXE programmal beépítjük a PKZIP.EXE-be.

— A fentiek elvégzése után az így feljavított PKZIP a ! parancsot a leírás szerint használja (lásd a PKZIP csomag ismertetésénél).

Az ARJ csomag a PKZIP-en is tútesz: Ha regisztráltatjuk és kifizetjük a kísérő leírásban közölt összeget, akkor egy olyan segédprogramot kapunk, amellyel magunk is egyszerűen létrehozhatunk biztonsági csomagolásban (security envelope) lévő SFX-et. Ez egyrészt kibontáskor automatikusan teszteli magát, másrészt az ARJ shareware verziójával semmi módon nem módosítható. Az ARJ egyik segédprogramja, a REARJ pedig az archívok átpakolásakor automatikus vírusellenőrzést is végezhet a McAfee féle SCAN programmal.

A vírusok az archív állományok közül — kiterjesztésük és funkciójuk miatt — leginkább az SFX (önkinyitó) archívokat veszélyeztetik. A COM illetve EXE kiterjesztés miatt a vírusok közönséges programállománynak tekintik és fertőzik őket. Ezért ajánlatos

az SFX archívok másolását, terjesztését, használatát csak a víruskereső és vírusirtó programok használata után végezni.

Veszélyek, ellenőrzés

A tömörített állományok súlyos veszélyeket is rejthetnek magukban, ha a bepakolt állományok már a bepakolást megelőzően is vírussal fertőzöttek voltak. Szerencsére azonban van mód a tömörítve tárolt állományok ellenőrzésére is. A SHEZ programba például beépítették a vírusellenőrzést, és a program a különféle archívokat a McAfee féle SCAN programmal képes tesztelni. Ez gyakorlatilag úgy zajlik, hogy a SHEZ a megfelelő kibontó programmal egy átmeneti munkakönyvtárba ideiglenesen kibontja a kijelölt archív bepakolt állományait, majd ráereszti a SCAN-t.

A biztonságos és jó hatásfokú vírusszűréshez célszerű a SCAN legfrissebb verzióját használni.

Az archív állományok vírusellenőrzéséhez a SHEZ-en kívül a CHECKOUT program is rendelkezésre áll, amely a SHEZ-hez hasonlóan a SCAN-t ereszti rá az ideiglenesen kibontott állományokra.

A fenti ellenőrzést természetesen manuálisan is megoldhatja bárki, aki nem szeret feleslegesen keret-, illetve segédprogramokat alkalmazni (például a rendelkezésre álló RAM szűkös volta miatt). Ez tanácsos akkor is, ha olyan vírusokra is akarunk ellenőrzést végezni, amelyeket a SCAN nem ismer fel, vagy ha egy másik víruskeresővel akarunk dolgozni: egy előzetes RAM ellenőrzést követően egyszerűen bontsuk ki egy üres átmeneti könyvtárba az archív bepakolt állományait, majd ereszük rá saját víruskereső, vírusirtó programunkat.

Az önkipakoló állományok egy részét szintén ellenőrizhetjük a SHEZ, CHECKOUT és SCAN programok segítségével, gyakran azonban csak az előbb leírt módon, manuálisan tudjuk az ellenőrzést elvégezni.

Az LHA által létrehozott Large model SFX (lásd részletesen ott) állományok a !/ paranccsal meghívva automatikusan el is indítják a szintén az SFX-ből frissiben kibontott !.BAT batch-fájlt. Ha a bepakolt állományok közül akár csak egyetlen is fertőzött volt, annak beláthatatlan következményei lehetnek.

Az újabban megjelent ARJ programot ez úton is mindenkinek a figyelmébe ajánlom. Egyrészt a program újabb „egzotikus” kiterjesztést használ, amit a korosabb vírusok még nem ismernek, másrészt a program az LHA-hoz hasonlóan önmaga épségét is ellenőrizni tudja. Az LHA a szabványos t (= teszt) paranccsal ellenőrzi önmagát, az ARJ pedig az i (= integrity check) parancsot használja erre a célra.

CHECKOUT.EXE

A programot mindenkinek ajánlom, aki védekezni akar a vírusszűrés ellen. A program tetszőleges mélységben egymásba pakolt archívokkal is boldogul. Használható interaktív és parancs üzemmódban, illetve a kettőt kombinálva. Kezeli az LHARC, PAK, PKZIP, ZOO és PKPAK által létrehozott archívokat. Az LHA sem okoz neki gondot, ha azt előzőleg átnevezzük LHARC.EXE-nek.

A program különlegessége, hogy nemcsak az archívokat kezeli, hanem a megadott EXE, COM, OVL, BIN, PIF és SYS állományokat is a megadott könyvtárban. Sajnos a program jelenlegi verziója az SFX állományokat (éppúgy, mint a SCAN) nem ellenőrzi megfelelően. Ezeket külön kell ellenőriztetni. Az SFX-eket ellenőrző rutinok elkészültek ugyan, de a szerzők még biztonságosabbá akarták azokat tenni, ezért csak a következő verzióba épülnek be szervesen.

Ha paraméterek nélkül hívjuk meg a CHECKOUT programot, akkor a képernyő bal felén a menüpontok láthatók, a jobb oldali ablakban pedig az éppen aktuális alapértelmezések:

Menüablak	Alapértelmezés ablak
Subdirectory to process – s	K:\UPLOADS (Munkakönyvtár)
Logging active – L	yes (Naplózzon)
Log subdirectory– o	K:\ (Naplókönyvtár)
Bad Files subdirectory – B	K:\BADFILES (Selejtkönyvtár)
Move bad Archives – A	yes (Sérült archívot vigye át)
Move Infected files – I	yes (Beteg fájlokat vigye át)
Re-archive files – R	no (Konvertáljon)
Use Time stamp code – T	:: (Keltezés átírása)
PCB DIRnn Files – P	no (PCB DIRnn fájlok is)
VirusScan active – V	yes (A SCAN aktív)
Move Good files – G	no (Jó állományok áthelyezése)
Write Batch File W	(Batch-fájlt ír a fentiekből)
Do It now D	(Végrehajtja az ellenőrzést)

A menüablakban ki kell választani a megfelelő pontot, majd az <ENTER> lenyomása után a kinyíló input ablakba be kell írni a megfelelő adatokat, illetve ki kell választani a megfelelő opciót. Az elérési útvonalak beírásakor az első karaktert a program a meghajtó kijelöléseként értelmezi és automatikusan kiteszi utána a :\ jeleket. A yes/no kapcsolók automatikusan átváltanak, ha a menüben kiválasztjuk a kétállású opciót. A konvertáláshoz a program felkínálja a lehetséges célformátumokat (LZH, ZIP, ARC, PAK, ZOO, NO).

A Write Batch File menüpont hatására a program megír egy CKOT.BAT batch-fájlt a menüben kiválasztott paraméterekkel. Ezt a batch-fájlt lefuttatva már megfelelően paraméterezve indíthatjuk a CHECKOUT programot.

A CHECKOUT alkalmazható paraméterei

/s<forrásdrive:\forráskönyvtár\archív fájl>

Ebben a formában kell megadnunk azt a könyvtárat amelyiket ellenőrizni akarjuk, ha nem az aktuális könyvtárat ellenőrizzük. Ha nem adunk meg fájlazonosítót, akkor a megadott könyvtár összes EXE, COM, BIN, SYS, OVL és PIF állományát ellenőrizteti a SCAN-nel. Amennyiben fájlazonosítót is megadunk, akkor csak az annak megfelelő állományokat ellenőrzi. Ha egy archív belsejében további archívok vannak, a CHECKOUT azokat is ellenőrzi.

/l

A naplózást engedélyezhetjük ezzel a paraméterrel. A naplóállományok a megadott naplózó könyvtárba kerülnek. A CHECKOUT.LOG állomány tartalmazza a tesztelt állományok neveit és a tesztelés eredményeként beírt üzeneteket.

/o<meghajtó:\>

A naplófájlt tartalmazó könyvtárat adhatjuk meg ezzel a paraméterrel.

/b<meghajtó:\selejtkönyvtár>

A sérült, illetve fertőzött állományokat gyűjtő selejtkönyvtárat ezzel a paraméterrel adhatjuk meg.

/a

Amennyiben ezt a paramétert is megadjuk, a program a sérült archívokat átviszi a /b paranccsal megadott selejtkönyvtárba.

/i

Amennyiben ezt a paramétert is megadjuk, a program a fertőzött állományokat átviszi a /b paranccsal megadott selejtkönyvtárba.

/r

Ez a paraméter a CHECKOUT program egyik különleges szolgáltatása: használatával át lehet alakítani, természetesen a megfelelő kibontó és tömörítő programok (PKPAK, PKUNPAK, PAK, LHARC, ZOO, PKZIP, PKUNZIP) segítségével a megadott könyvtárban lévő archív állományokat. A sima /r letiltja a konvertálást, a /rz ZIP, az /ra ARC, az /rl LZH, az /rp PAK és az /ro ZOO kiterjesztésű archívokat generál.

/t

A már ellenőrzött állományoknak új keltezészt adhatunk ezzel a paranccsal. Ha a CHECKOUT a megadott dátumú állománnyal találkozik, akkor átugorja, nem ellenőrzi. Ezáltal lehetséges, hogy a program a munkanap elején vagy végén csak az utolsó tesztelés óta létrehozott és módosított állományokat tesztelje.

/p

Ez a paraméter a PCBoard BBS-ekkel kapcsolatos feladatokhoz kell. Mivel egyelőre elenyésző azok száma, akik Magyarországon ezt használják, itt eltekintek ismertetésétől. Akit mégis érdekel, az megtalálja az angol nyelvű leírást a CKOT11.TXT állományban, amit a CHECKOUT-tel együtt terjesztenek.

/v

A kapcsoló használata azt közli a programmal, hogy a SCAN aktív.

/g<használható könyvtár>

E kapcsolóval megadható, hogy ellenőrzés után a program az itt megadott könyvtárba helyezze át az épnek és fertőzésmentesnek bizonyult állományokat.

/j

Megegyezik a fent leírt /g paraméterrel, batch-fájlokban használható.

A CHECKOUT segédállományai

A program menüjő rendszere segítségével elkészíthetjük a CKOT.BAT állományt, ami a CHECKOUT paraméterezett meghívására szolgál. További két BAT állományt használ még a program: a CKOT\$1.BAT-ot az újrapakolás előtt keresi meg, a CKOT\$2.BAT-ot pedig a time stamp beállítása előtt. Ha nem találja ezeket, akkor a szokott módon fut le.

Hasznos tanácsok

— Idegen forrásból származó programokat előzetes tesztelés nélkül ne futtassunk. Ugyanez érvényes arra az esetre is, ha egy másik gépről, amely nem a mi felügyeletünk alatt áll, lemezen hozunk át állományokat.

— Az archív állományokat is ajánlatos tesztelni. Egyrészt a megfelelő tömörítő Test parancsával, majd akár manuálisan, akár a SHEZ vagy a CHECKOUT és a SCAN program segítségével.

— Ha a PKZIP-et használjuk, és módunkban áll regisztráltatni a programot, ne habozzunk mihamarabb megtenni azt. Érdemes! Az ARJ-nál is!

— Az önkinyitó (SFX) archív állományokat se felejtjük el tesztelni, mielőtt tartalmukat rászabadítanánk gépünkre.

— Ha egy adott víruskeresővel nem találunk fertőzést a vizsgált állományok között, az sajnos még nem garancia arra, hogy a vizsgált állományok 100%-os biztonsággal vírusmentesek. Nem árt a vírusellenőrzést többféle víruskereső programmal is elvégezni. Több szem többet lát!

— A programozók mindig gondoljanak arra, hogy nagyobb egy program iránt a bizalom, ha a róla készített leírásban és ismertetőben szerepelnek a VALIDATE programmal kapott ellenőrző összegek és a program veszélyeztetett állományainak pontos méretei.

— Magát a tömörítő programot sem árt néha ellenőrizni. Erre közvetlen lehetőség van az LHA (LHA t path\LHA.EXE), illetve az ARJ program (ARJ i) esetében, míg a többi archiváló programnál megfelelő víruskeresőre van szükség.

Az önkinyitó SFX-ek

A könyv megfelelő fejezeteiben az adott tömörítő programok leírásaiban mindenhol szerepel az SFX állományok készítése és használata, de ez a téma megérdemli, hogy külön összefoglaló fejezetet is szánjunk neki. Ez a rész azoknak szól, akik majdnem profi módon tudják már használni megszokott tömörítő programjaikat és szeretnék kihasználni az SFX állományok nyújtotta lehetőségeket is.

A tömörítő programok használata során előfordul, hogy nem tudunk sem megnézni, sem kibontani egy tömörített állományt. Vagy azért, mert nincs kéznél a megfelelő kibontó állomány, vagy nem ismerjük a kibontó program helyes használatát.

Az első esetre megoldást jelent, ha összegyűjtjük a legfontosabb tömörítő programcsomagokat, s azt egy path-on lévő közös könyvtárba tesszük. Igazán fontos a PKZIP, a PKPAK, a PAK, az LHA, az ARJ és a PKLITE, míg a DWC, a ZOO, az ARC, az LZEXE és a KK inkább különleges csemege.

A fent említett programok teljesértékű kihasználásáig eljutni nem kis feladat. Senkitől sem várható el, hogy bemagoljon programonként 50-60 oldalnyi leírást, ráadásul többnyire angolul, csupán azért, hogy napi rendszeres (vagy ráadásul olykor kevésbé rendszeres) be-, illetve kikapolásait elvégezze.

Átmeneti megoldás a SHEZ, illetve a ZIPVIEW program használata. Az előbbi előnye, hogy a hagyományos és az SFX archívok szinte minden lehetőségét ki tudja aknázni, viszont meg kell ismerni újabb 40-50 oldalnyi angol nyelvű leírást. A ZIPVIEW zoknak ideális, akik csak bele akarnak kukkantani a tömörített állományokba, beleolvasnának egy-egy bepakolt szöveges fájlba, vagy időnként kibontanának egy-egy állományt. A fent említett két program elsősorban a másoktól kapott archívok használatba vételét könnyíti meg.

Az SFX állományok lehetőségei

- Csak 1-14 K-val nagyobbak az eredeti archívnál.
- Kinyitásukhoz nem kell más, csak elegendő üres lemezterület a céllemezen.
- A legtöbb SFX ugyanúgy, vagy majdnem ugyanúgy kezelhető, mint a hagyományos archívok.
- Egyes programcsomagok SFX állományai különféle extra szolgáltatásokkal is rendelkeznek.
- Tesztelni lehet épségüket.

Ajánlatos az SFX állomány mellé legalább egy, de inkább két kísérő állományt tenni: egy TXT, DOC, vagy README állományt, amelyben leírjuk, mi van az SFX-ben és hogyan ajánlatos kinyitni, valamint egy kikapoló BAT állományt, hogy könnyebbé tegyük az SFX fogadását.

Ha az LHA-val dolgozunk, használjuk ki a ! és !.BAT állományok adta lehetőségeket a külső leírások és batch-fájlok mellett, esetleg helyett.

Melyikkel hogyan

A PAK-, ZIP- és ARJ-SFX-ek is elláthatók az LHa-hoz hasonló Telop-pal. (Részletesen lásd az adott program ismertetésénél!)

1) PKARC 3.60 és PKPAK 3.61

Kell hozzá:

MAKESFX.COM (A PK36.EXE, illetve a PK361.EXE tartalmazza.)

PKSFX.PGM (A MAKESFX hozza létre a PK36.EXE, illetve a PK361.EXE állományokból.)

ARCHIV.ARC (Előzőleg létrehozott archív állomány.)

Az SFX előállítás (a DOS promptról):

COPY /B PKSFX.PGM + ARCHIV.ARC ARC-SFX.EXE

A /B paraméter fontos! Enélkül hibásan hoznánk létre az SFX állományt. Növekedés: kb. 13,1 K.

Az SFX archív frissítése: Kibontás, aktualizálás, utána pakolás, majd SFX képzés.

Észrevételem: Mivel a programok parancskészlete és szintaktikája csaknem teljesen megegyezik, próbáljanak áttérni a PKZIP 1.10 használatára. A ZIP gyorsabb, tömörebb, biztonságosabb és az SFX készítés is sokkal egyszerűbb. A meglévő ARC állományok könnyedén átkonvertálhatók a TOZIP, a PAK vagy a Rearj programmal.

Választhatják a NoGate PAK programját is, és akkor segédprogram nélkül tetszés szerint tudnak majd ARC, PAK és ZIP állományok, illetve belőlük készített SFX-ek között konvertálni.

2) PKZIP 0.9x-től PKZIP 1.02-ig

Kell hozzá:

MAKESFX.COM (A PKZXXX.EXE tartalmazza.)

PKSFX.PRG (A MAKESFX hozza létre a PKZXXX.EXE állományból.)

ZIP2EXE.EXE (A PKZXXX.EXE tartalmazza.)

ZIP_fájl.ZIP (Előzőleg összepakolt archív állomány.)

Az SFX előállítás (a DOS promptról):

A PKSFX.PRG-nek az aktuális könyvtárban, vagy path-on kell lennie az SFX készítésekor.

ZIP2EXE ZIP_fájl [SFX_fájl]

Ez az SFX a PKUNZIP majdnem minden parancsát elfogadja és végrehajtja (kivéve a -v parancsokat). Növekedés: kb. 15,5 K.

SFX archív felfrissítése: Azonos a ZIP_fájl kezelésével, csak a kiterjesztést is be kell írunk.

Észrevételem: Szerezzék be a PKZIP 1.10 verziót. Minden tekintetben kompatibilis a régivel, tehát nem vesz el semmi adat a már meglévő ZIP fájlokból. A PKZIP új verziója tömörebb, gyorsabb lett és új szolgáltatásokkal is bővült. A REZIP segédprogram, vagy a SHEZ pedig felfrissíti a régi verziókkal készített ZIP állományokat.

3) PKZIP 1.10

Kell hozzá:

ZIP2EXE.EXE (A PKZ110.EXE tartalmazza.)

ZIP_fájl.ZIP (Előzőleg összepakolt archív állomány.)

Az SFX előállítása (a DOS promptról):

Az 1.10 verziótól kezdve nincs szükség többé a PKSFX.PRG-re.

a) ZIP2EXE ZIP_fájl [SFX_fájl]

Növekedés: kb. 12,8 K.

b) ZIP2EXE -j ZIP_fájl [SFX_fájl]

A -j hatására kisebb méretű, de butább SFX jön létre, ha csak imploding technikával pakoltuk össze a ZIP_fájlt. Növekedés: kb. 2,9 K.

c) ZIP2EXE -f ZIP_fájl [SFX_fájl]

Az -f opcióval OS/2 alatt az ún. family model-t hozza létre.

SFX archív frissítése: Azonos a ZIP fájl kezelésével, csak a kiterjesztést is be kell írunk.

Észrevételem: Jól választottak. Az 1.10-től a ZIP-SFX is a jelszavas titkosítást! (Nem kell beugratni!)

4) LHARC 1.13 és LHICE (1.14, ICE)

Kell hozzá:

LHARC.EXE vagy LHICE.EXE (Maga a tömörítő program.)

LZH_fájl.LZH vagy LZS_fájl.LZS (Előzőleg bepakolva.)

Ajánlott a Telop (! fájl az LZH fájl elején) használata AUTOLARK.BAT (El is induló SFX-ekhez, az LZH fájlban.)

Az SFX előállítása (a DOS promptról):

a) Small model (COM, vagy EXE, 64K alatt, de butább SFX.)

LHARC s LZH_fájl

Növekedés: kb. 1,3 K.

b) Large model (EXE kiterjesztés, 64 felett.)

LHARC s /xkjelszó LZH_fájl

Növekedés: kb. 1,9 K.

SFX archív frissítése: Azonos az LZH_fájl kezelésével, csak a kiterjesztést is be kell írunk.

Észrevételem: Térjenek át minél előbb az LHA 2.1x változatára. Alig változott a szintaxis, a parancs- és kapcsolókészlet, viszont még tömörebb lett és gyorsult is a program. Az önindító SFX-ek lehetősége ragyogó!

5) LHA.EXE 2.XX (jelenleg 2.12-ig)

Kell hozzá:

LHA.EXE (Maga a tömörítő program.)

LZH_fájl.LZH vagy LZS_fájl.LZS (Előzőleg bepakolva.)

Ajánlott a Telop (! fájl az LZH fájl elején) használata.

!.BAT (El is induló SFX-ekhez, az LZH fájlban.)

Az SFX előállítása (a DOS promptról):

a) Small model (COM, vagy EXE, 64 K alatt, de butább SFX)

LHA s LZH_fájl

Növekedés: kb. 1,5 K.

b) Large model (EXE kiterjesztés, 64 K felett.)

LHA s /x LZH_fájl

Növekedés: kb 1,9 K.

SFX archív frissítése: Azonos az LZH fájl kezelésével, csak a kiterjesztést is be kell írunk.

Észrevételem:

Jól választottak. Nem árt persze a mintegy havonta megjelenő újabb verziókkal felfrissíteni program-gyűjteményünket.

6) ARJ 2.10

Kell hozzá:

ARJ.EXE

Forrás ARJ fájl, vagy a még nyitott állományok.

Az SFX előállítás (a DOS promptról):

a) Normál ARJ-SFX egyetlen meglévő archívból:

ARJ f -je ARJ_fájl[.ARJ] ...

vagy

ARJ j -je ARJ-SFX ARJ_fájl[.ARJ]

Normál ARJ-SFX több meglévő archívból:

ARJ j -je ARJ-SFX ARJ_fájl1[.ARJ] ARJ_fájl1[.ARJ]

Normál ARJ-SFX közvetlenül a még nyitott forrásállományokból:

ARJ a -je ARJ_fájl[.ARJ] [bázis\] [maszk]

A kiterjesztés elhagyható, alapértelmezése ARJ. Az így létrehozott SFX saját parancsokkal, kapcsolókkal és Help képernyővel is rendelkezik. Növekedés: kb 14 K.

b) „Buta” SFX (SFXJR) előállítás. Megegyezik a fenti parancsokkal, csak -je helyett -jel a kapcsoló. Növekedés: kb 6 K.

Észrevételem:

Az ARJ nagyon tömör és sok szolgáltatással rendelkezik. Csak akkor érdemes más tömörítőre áttérni, ha a nagyobb sebesség az elsőrendű szempont.

7) ARC 6.02

Kell hozzá:

MKSARC.EXE (Benne van az ARC602.EXE állományban.)

ARCHIV.ARC

Az SFX előállítás (a DOS promptról):

MKSARC ARCHIV[.ARC] [SFX_fájl]

A kiterjesztés alapértelmezése az archívnál ARC. Ha nem adunk meg SFX fájlnevet, akkor az archív nevét kapja EXE kiterjesztéssel. Növekedés: kb 7,8 K.

SFX archív frissítése: Mint a hagyományos ARC fájloké, csak a kiterjesztést is meg kell adni.

Észrevételem: Sürgősen térjenek át valami jobb programra. Javaslom a PKZIP 1.10, a PAK 2.50 és az LHA 2.xx programokat.

8) PAK 2.50

Kell hozzá:

PAK.EXE (Maga a tömörítő program.)

ARCHIV.ARC, PAK_fájl.PAK, vagy ZIP_fájl.ZIP (előre bepakolva), vagy helyettük az SFX-be bepakolandó, még nyitott állományok.

Az SFX előállítás (a DOS promptról):

a) Már meglévő archívból:

PAK c archív /exe

Így SFX állományokká lehet alakítani a PAK, az ARC és a ZIP kiterjesztésű archívokat is. A PAK a forrásállományt törli az önkinyitóvá alakítás után. Növekedés: kb. 6,9 K.

b) Nyitott állományokból:

PAK a PAK-SFX /exe

A többi bepakoló parancs (u, m) mellett is SFX állományt eredményez az /exe kapcsoló. SFX archív frissítése:

PAK c SFX_fájl.EXE

Leszedi az önkibontó fejléct az archívról. Az archívot tetszés szerint lehet frissíteni, utána pedig visszaalakítani SFX állománnyá.

Észrevételem: Jól választottak.

9) ZOO 2.01

Kell hozzá:

SEZ.EXE (A ZOO csomag része.)

ZOO_fájl.ZOO (ZOO-val készített archív.)

Az SFX előállítás (a DOS promptról):

SEZ ZOO_fájl EXE_fájl

A kimeneti EXE fájl neve nem tartalmazhat sem könyvtár-, sem meghajtónevet. A ZOO fájl kiterjesztésének alapértelmezése ZOO, az EXE fájl kiterjesztésének az alapértelmezése pedig EXE. Növekedés: kb. 2,9 K.

SFX archív frissítése: Csak a ZOO állományt lehet vele frissíteni.

Észrevételem: Gondolják át újra, milyen céllal kezdték a ZOO programot alkalmazni. Ha csak a helytakarékoság volt a cél, akkor sürgősen térjenek át egy jobb programra. Javaslom a PKZIP 1.10, a PAK 2.50 és az LHA 2.xx, illetve az ARJ programokat. Ha a ZOO speciális szolgáltatásai miatt döntöttek a program mellett, (generációk, hosszú kommentek, visszahozás, adatbiztonság) akkor használják, de ne feledkezzenek meg a többi tömörítő programról, már csak az adatscere miatt is.

A következő táblázat a különböző SFX-ek szolgáltatásait hasonlítja össze. A fejlécben szereplő számok a tárgyalt tömörítőtípust, a betűjelzések (abc) az SFX altípustokat jelölik.

Az SFX archívok különleges szolgáltatásainak táblázata

	1	2	3a	3b	3c	4a	4b	5a	5b	6	7	8	9
Közvetlen frissítés	-	+	+	+	+	+	+	+	+	+	-	+	-
Célkönyvtár adható	+	+	+	-	-	-	+	-	+	+	+	+	-
Fájlmászk adható	+	+	+	-	-	-	+	-	+	+	+	+	-
Archívkatalógus	+	-	-	-	-	-	+	-	+	+	-	-	-
Könyvtár visszaáll	+	+	+	-	-	-	+	-	+	+	+	+	-
Nyomtatás	+	+	+	-	-	-	-	-	-	-	+	+	-
Jelszavas védelem	-	-	+	-	-	-	-	-	-	+	+	-	-
Listafájl lehet	-	-	+	-	-	-	-	-	-	+	-	-	-
Telop	-	-	-	-	-	+	+	+	+	+	-	+	-
Kipakoló indítás	-	-	-	-	-	-	+	-	+	-	-	-	-
Jogosultságellenőrzés	-	-	+	+	+	-	-	-	-	+	+	-	-
Help	+	+	+	+	+	-	-	-	-	+	+	-	-
Generációkezelés	-	-	-	-	-	-	-	-	-	-	-	-	+

Általános tippek, tanácsok

Bár a legtöbb tömörítő tudja aktualizálni az SFX archívokat is, néha biztonságosabb újraszervezni a kibontott állományokat és utána új SFX-et felépíteni.

Kibontás előtt nem árt megnézni, mekkora szabad lemezterület kell a kibontáshoz. Ajánlott programok: MAD, FV, ZIPVIEW, SHEZ, MTV. Ezek az összes archív formátumot ismerik. Ha tudjuk, melyik SFX típussal van dolgunk, akkor az adott kibontó programnak az archívkatalógust kiíró (l vagy v) parancsával is megnézhetjük. Nemcsak a helyigény miatt érdemes megnézni az SFX katalógust, hanem azért is, hogy lássuk, a könyvtárszerkezetet is elmentették-e. Ha igen, ajánlatos a könyvtár visszaépítő paranccsal kinyitni az SFX-et.

Az LHARC és LHA programok SFX-jébe tegyünk bele egy telop-ot, amiben tanácsokat adunk a kibontáshoz. A telop mindig a legelsőnek bepakolt állomány legyen.

Írhatunk az SFX mellé egy kibontó batch-állományt is, amely a winchesteren létrehozza a szükséges könyvtárat és abba kipakolja az SFX tartalmát, anélkül, hogy magát az SFX-et rávinné a winchesterre.

Ha csak DOS-ból bontunk, célszerű a célkönyvtárra állni és onnan az A:SFXfájl paranccsal kinyitni az SFX archívot.

Néha nem árt, főleg méretesebb archívok esetén felszabadítani a RAM-ot a felesleges rezidens programoktól, mert egyébként a gépet kiakadhat.

A mamut méretű PKZIP eredetű SFX-eket érdemes a ZIPDMP-vel kisebb darabokra szabdálni. Így változatlanul SFX-ek maradhatnak a ZIPfájlok, de kezelhetőbbek és szállíthatóbbak lesznek.

A SHEZ a SCAN programmal az archívokból és SFX-archívokból is ki tudja pécézni vírusvendégeinket. A CHECKOUT az ismertebb archívokkal ugyanezt neg tudja csinálni, csupán az SFX állományok kezelése maradt még ki belőle.

Ha aktualizáljuk SFX, vagy hagyományos archív állományainkat, lehetőleg a winchesteren tegyük. Amellett, hogy gyorsabb, mint a floppy, általában van elég szabad

lemezterület az aktualizáláskor keletkező átmeneti állományok kezelésére. Nem árt mindig a legnagyobb szabad területű partíción dolgozni.

Melyiket mire?

LHA-SFX (LZH_fájl -- SFX_fájl)

Az LHA segítségével előállított SFX állományok mind közül a második legtömörebbek. A telop egyedülálló lehetőséget ad, hogy a felhasználó kibontás előtt tájékozódhasson, egyáltalán kell-e neki a csomag. Ha nem felejtettük el az SFX előállításakor az s parancs mellett az /x kapcsolót is megadni, akkor a !.bat indító állomány a !/ parancs hatására közvetlenül a kikapolás után végrehajtásra kerül. Az SFX-nek megadhatunk egy célkönyvtárat is (szintén a Large modellel), ahova kikapolja magát. Nem utolsó az a szempont sem, hogy az SFX elkészítéséhez a forrásállományokon kívül mindössze egyetlen, 34 K méretű LHA.EXE állomány kell.

ARJ-SFX (ARJ_fájl -- SFX_fájl, ill. forrás -- SFX_fájl)

Az ARJ az LHA előtt áll a tömörítés határfokát nézve. Extra szolgáltatásai is igen előkelő helyre sorolják. Az SFX előállítása egyszerű, nem kell hozzá semmiféle segédállomány. Vissza is lehet alakítani az ARJ-SFX-et egyszerűen sima ARJ állománnyá. Telop-ja az LHA-hoz hasonló. A jelszavas védelmet is támogatja.

PKZIP-SFX (ZIP_fájl -- SFX_fájl)

Hacsak nem a -j paranccsal készítettük az SFX-et, az önkibontó állomány a PKUNZIP minden parancsát ismeri (a -v parancsok kivételével). Így azután a ZIP-SFX-ből akár nyomtathatunk is, sőt az 1.10 verziótól a jelszavas titkosítás és a listafájl használata is megengedett. Egyenlőre egyedülálló szolgáltatás (csak a ZOO-nak van hasonló), hogy a sérült archívokból az ép részeket ki lehet ollózni. A ZIPDMP segédprogrammal a kellenél nagyobbra sikeredett ZIP-SFX-eket pedig kisebbekre lehet szeletelni.

A ZIP-SFX-ben is lehet Telop, de az előre megszerkesztett ZIP kommentet, amelyben egy CTRL-S karaktert elhelyeztünk nem közvetlenül a PKZIP-pel, hanem a ZIPCOMNT segédprogrammal vagy a PAK programmal adjuk hozzá.

PAK-SFX (PAK_fájl, ARC_fájl, ZIP_fájl -- SFX_fájl)

A PAK programmal a SEA-ARC, a PKARC/PKPAK a régebbi verziójú PAK által előállított *.ARC állományokat könnyedén konvertálhatjuk egy PAK-SFX fájlba, csakúgy, mint a PKZIP-pel előállított ZIP fájlkat. A PAK-SFX előállítása igen egyszerű, mivel a forrásállományokon és a PAK.EXE programon kívül semmiféle egyéb segédprogramra nincs szükségünk. A PAK-SFX tömör, gyorsan kinyílik.

ARC-SFX (ARC_fájl -- SFX_fájl)

Az SFX ugyanúgy kezelhető, mint az eredeti *.ARC fájl az ARC programmal. Az előállított SFX által értelmezett parancsok megegyeznek Vernon Buerg ARCE programjának parancsaival.

ZOO-SFX (ZOO_fájl -- SFX_fájl)

Csak *.ZOO formában aktualizálható. Túlzottan sok előnye nincs.

PKPAK-SFX (ARC_fájl -- SFX_fájl)

A PKARC és a PKPAK programokkal előállított *.ARC fájlon kívül a PKSFX.PGM állományra is szükségünk van a PKPAK-SFX előállításához. Ez utóbbit a PK361.EXE-ből a belőle kibontott MAKESFX.COM hozza létre. Az SFX-et a DOS-ból, vagy a SHEZ hatékony közreműködésével állíthatjuk elő. Az egyik legnagyobb SFX fej rakódik rá az .ARC állományra, de cserébe az SFX a PKUNPAK/PKXARC minden parancsát értelmezni tudja.

A DWC és a HYPER SFX ritka, leírásukat lásd a DWC, illetve a HYPER program ismertetésénél.

A könyvben említett külföldi nevekhez kapcsolódó címek

**PKARC, PKXARC, PKSFX, PKPAK, PKUNPAK, PKFIND, PKZIP, PKUNZIP,
ZIP2EXE, ZIPFIX, PKLITE, CHK4LITE, HDROPT**

**PKWare, Inc.
7545 N Port Washington Rd.
Glendale, WI 53217, USA**

**PAK, PAKINST
NoGate Consulting
P.O.Box 88115
Grand Rapids, MI 49518, USA**

**LIST, ARCE, FFIND, FV
Vernon D. Buerg
139 White Oak Circle
Petaluma, CA 94952, USA
BBS: (707) 778-8841**

**SCAN
McAfee Associates
4423 Cheeney St.
Santa Clara, CA 950514, USA
Tel: (408) 988-3832
BBS: (408) 988-4004**

**ARJ, REARJ, ARJSORT, UNARJ
Robert K. Jung
2606 Village Road West
Norwood, MA 02062, USA**

**SHEZ, SHEZCFG
California Software Design
James Derr
P.O.Box 15248
Santa Rosa, CA 95402, USA**

**STACKER
Stac Electronics
5993 Avenida Encinas
Carlsbad, CA 92008, USA
Tel: (800) 522-5335**

**NEWSPACE
Isogon Corporation
330 Seventh Avenue
New York, NY 10001, USA
Tel: (212) 967-2424**

EXPANZ! PLUS
InfoChips Systems
2840 San Thomas Expressway
Santa Clara, CA 95051
Tel: (408) 727-0514
Fax: (408) 727-4190

LZEXE, LZSHELL, COMTOEXE, LZESHELL, UNPACKEXE, LZ
Fabrice Bellard
451 Chemin du mas de Matour
34790 Grabels, France

Magyarországi címek:

SolarSoft Programkönyvtár — Floppyland szaküzlet
1251 Budapest V., Váci u. 84.
Telefon/Fax: 118-2651

Cédrus Informatikai Részvénytársaság
Cédrus Kiadó — Alaplap szerkesztőség
1251 Budapest XI., Karolina út 17.
Telefon: 166-2111
Fax: 185-2221

Galax Kereskedelmi Kft.
1113 Budapest, Bocskai út 54.
Telefon/Fax: 166-7557

PannonSoft
1399 Budapest, II., Vérhalom tér 10.
Telefon/Fax: 135-9755

TallSoft LopWare Studio & Library — Nagy Gábor
1116 Budapest, XI., Kisköre u. 16.
Telefon: 166-1444 (este, lakás)
157-5311/160 (napközben, munkahely)

Tartalom

Bevezetés	7
Alapfogalmak, értelmezések	9
Általános fájl-tömörítők	13
Tömörítők tesztje	17
A legsikeresebb — a PKZIP	21
Menüvel bontogatni — PKZMENU	39
Kommentezés egy iZIPicit elegánsabban	47
Az adatsűrités éLHArCosa	51
Ahogy én PAKolok...	63
Új trónkövetelő: az ARJ	79
A jó öreg PKARC és PKPAK	99
Az ősi ARC	111
„Temetni jöttem...” — DWC	121
Bevezetés a ZOO-lógiába	127
HYPER — a mélynövésű óriás	139
EXE és COM fájlok tömörítői	141
Egy kellemes meglepetés — PKLITE	143
Ezerarcú LZEXE	149
A besorolhatatlan KK	151
Fújd fel a winchestered!	153
Stacker, te csalfa!	155
Új (NewSpace) a Nap alatt	163
Expanz! Plus — a kis drága	169
Archívkezelő utility-csomag	171
Fájlkereső segédprogramok	173
(PKFIND, SFIND, PKZFIND, PKFF, FFIND, WHEREIS)	

Archív kukkantók	179
(MAD, FV, LHDIR, ZIPVIEW, LIST)	
A szalámitaktika	189
(ZIPDMP, SLICE)	
Konvertálók, SFX-készítők, kiollózók	193
(REZIP, TOZIP, 2LZH, LHDIR, CHECKOUT, MAKESFX, SEZ, MKSARC, ZIP2EXE, PKZIPFIX, REARJ, NO, TREED)	
Keretjáték	201
(ZIPGUN, LHMENU, IDCHELL, NARC, 4ZIP)	
SHEZ — a tömörítők Norton Commandere	211
A tömörítők és a vírusok	223
(CHECKOUT)	
Az önkinyitó SFX-ek	229
Nevek, címek	237

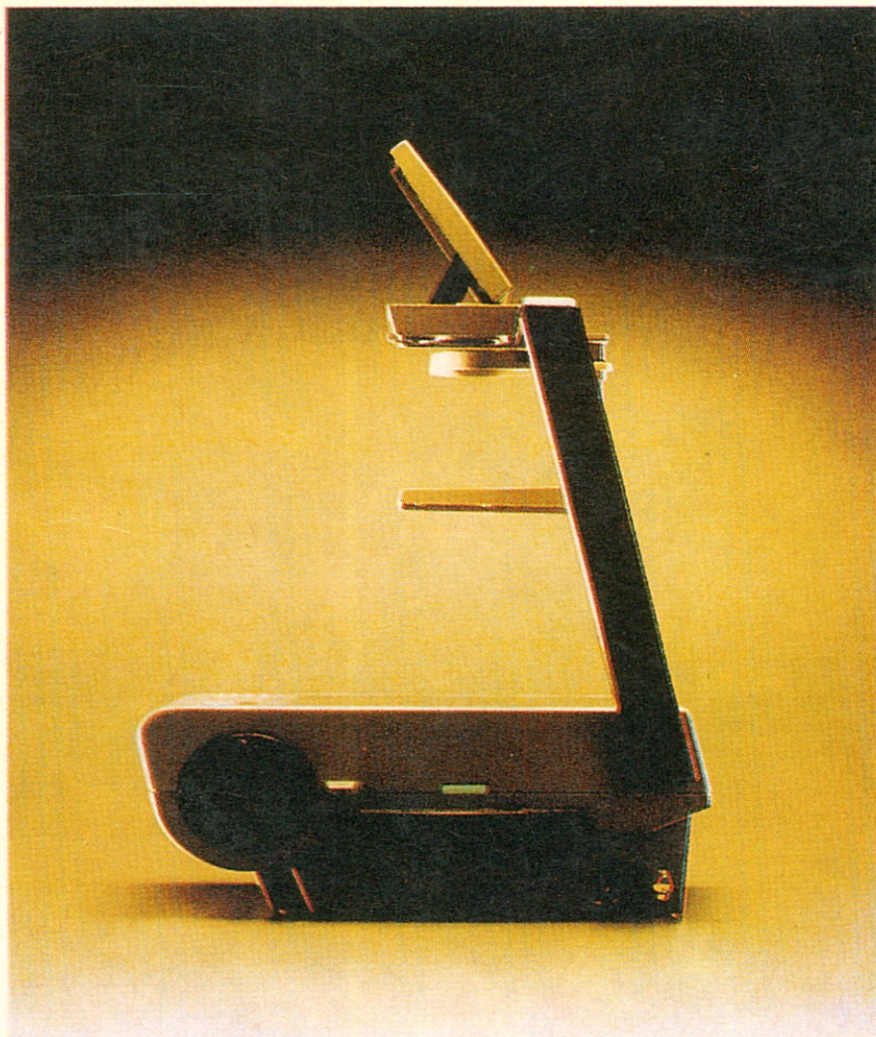
256,- FT

Ez a helyes írásvetítő!

Nappali fény mellett is kitűnő képet varázsolnak a vászonra a Polaroid írásvetítők.

A könnyen hordozható legkisebb modell mindössze 4,5 kg.

Előadások látványossá tételéhez ideális útitárs!



Többféle változatban kaphatók a

FLOPPYLAND

számítástechnikai szaküzletben
(Budapest V., Váci utca 84. Telefon/Fax: 118-2651)
és országszerte a Cédrus Rt. viszonteladójánál.

Polaroid