

UNd64

COLLABORATORS

	<i>TITLE :</i> UNd64		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 6, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	UNd64	1
1.1	UNd64 Guide	1
1.2	UNd64 Guide	1
1.3	Dementi	2
1.4	Programbeskrivning	3
1.5	Systemkrav	3
1.6	Användning	3
1.7	I början fanns UNd64...	4
1.8	.d64/.x64 Support	5
1.9	Vad är en .d64 diskimage?	5
1.10	.x64-anmärkningar	5
1.11	.d64-anmärkningar	6
1.12	.d64 / .x64 utdata	6
1.13	Stöd för .t64	7
1.14	Vad är en .t64 tapeimage?	8
1.15	.t64 utdata	8
1.16	.t64-anmärkningar	8
1.17	Stöd för .p00	9
1.18	Stöd för .lnx	9
1.19	Stöd för .lbr	10
1.20	Stöd för .sfx	10
1.21	Felmeddelanden	11
1.22	Kompatibilitet	12
1.23	Programmets utveckling	12
1.24	Kvar att göra	17
1.25	Upphovsmannen	17
1.26	Också ett citat	18
1.27	Den före detta 'Amiga 600 From Hell'	18
1.28	Tack till	18
1.29	Stöd för !zipkodad RLE-komprimerad split diskimage	19
1.30	Översättarnas kommentarer	20

Chapter 1

UNd64

1.1 UNd64 Guide

UNd64 v 38.15 (3.7.97)

© Copyright 1994-1997 Jess Sosnoski

Detta program är numera emailware. Om du vill smaka på hur det känns att vara registrerad är det bara att skicka ett mail.

Om du omöjligen kan skicka mail är vykort eller liknande medel godtagbara - pluspoäng ges för röksignaler.

Registrering syftar även till att ge mig en överblick över programmets målgrupp.

OBS! Till min stora bestörtning har jag sett att några människor refererar till programmet som version 0.xxx. Detta är helt förkastligt då det är version 38 revision 15.

Vidare

1.2 UNd64 Guide

UNd64 v 38.15 (3.7.97)

© Copyright 1994-1997 Jess Sosnoski

Översättarnas kommentarer

Dementi

Programbeskrivning

Systemkrav

Användning

Felmeddelanden

Stöd för .d64/.x64

Stöd för !zipkodade filer

Stöd för .t64

Stöd för .p00

Stöd för .lnx

Stöd för .lbr

Stöd för .sfx

Kompatibilitet

Programmets utveckling

Kvar att göra

Om upphovsmannen

Ånyo ett citat

I början fanns UNd64...

Tack till

1.3 Dementi

Dementi

Detta program används på egen risk. Jag tar inget ansvar överhuvudtaget för eventuella problem eller skador som inträffar vid användning, modifikationer av eller annat som kan härröra från eller eventuellt kopplas samman med denna produkt eller dess beståndsdelar, vilken form de än må vara i.

Detta program är copyright Jess Sosnoski och finns enkom tillgängligt som emailware. Det får spridas förutsatt att följande är uppfyllt:

1. Programmet och dess dokumentation får inte ändras på något sätt.
2. Programmet och dess dokumentation måste spridas tillsammans.
3. UNd64 får ej användas i kommersiellt syfte utan mitt godkännande.
4. UNd64 får spridas på CD-ROM, t.ex. Aminet och Fred Fish.
5. UNd64 får spridas via coverdiskar med den restriktionen att jag kontaktas först så att den spridda versionen verkligen är den senaste. Det skulle dessutom vara trevligt om jag fick ett exemplar av tidningen ifråga.

/* Tanken att bli odödliggjord på en skiva laseretsad aluminium
omhuldad av icke nedbrytbar plast tilltalar mig skarpt. */

1.4 Programbeskrivning

Programbeskrivning

Detta program syftar till att öka användbarheten av C64-emulatorer - och för den delen även C64:or - genom att ge dig möjlighet att tillgå filer sparade i olika typer av arkiv från diverse källor.

Arkiv som stöds i denna version:

- .d64 disk images, 35-trackiga med eller utan felinformation.
- .t64 band (tape images)
- .p00 program images från PC-kompatibla C64-emulatorer.
- .x64 disk images från Unixstödda C64-emulatorer.
- .lnx C64 LYNX-arkiv (alla versioner, både punter- och xmodemtyper)
- .lbr C64 library-arkiv
- .sfx C64 självextraherande arkiv
- !zipkodade RLE-komprimerade split diskimages

Du kan hitta ett överflöd av C64-program på Internet. Det räcker med att söka på "64 software" på någon lämplig sökmotor.

1.5 Systemkrav

Systemkrav

En Amiga med Kickstart 1.1+ (Jag har inte tillgång till 1.0, så jag har inte kunnat testa programmet på en sådan konfiguration), runt 200 Kb ledigt minne och förhoppningsvis en räckta filer att använda programmet på.

1.6 Användning

Användning

Från Shell:
und64 [kommando] {valmöjlighet} [filnamn] {sökväg}

Programmet kan inte startas från Workbench.

Om programmet startas utan att några parametrar är specificerade kommer en hjälptext att skrivas ut. Närsomhelst kan CTRL-C användas för att avbryta programmet.

Kommandon:

- l Visa filer i aktuellt arkiv.
- x Extrahera filer från arkiv.
- t Testa filer i arkiv. (Om detta kommando tillämpas på en .p00-fil fungerar det som "l".)

Valmöjligheter:

d Möjliggör handhavandet av DELETEDe filer (endast .d/x64).

WARNING! Handhavandet av DELETEDe filer kan leda till oönskade sidoeffekter. Om en ny fil delvis har skrivit över den borttagna kan den återskapade filen eventuellt innehålla delar av den förra.

a Bearbeta samtliga filer (endast .d/x64).

Möjliggör extrahering av s.k. splat-filer, dvs. felaktigt stängda filer. Se även varningen ovan. Splat-filer har prefixet *.

n Avstå från filnamnskonvertering. Detta förändrar inte filnamnen, vilket medför att de kan se en aning lustiga ut. Otillåtna tecken ersätts emellertid ändå.

OBS! Filnamnen konverteras till AmigaASCII, och alla tecken vars kod är <32 eller >127 ersätts med passande tecken; /"?:#* ersätts med "-".

Filnamn:

Namet på filen samt dess suffix om något sådant finns.

Sökväg:

Behöver endast anges om du vill att resultatfilerna skall hamna i ett annat bibliotek än det indatan hämtas ifrån.

Om din angivna sökväg inkluderar ett bibliotek måste detta efterföljas av en slash (/), t.ex. RAM:Oepir_Risti/ ; om slashen utelämnas extraheras filen till RAM:

OBS! Sökvägen får ej omfatta fler än 255 tecken om du inte tycker om oförutsägbara resultat.

Exempel:

und64 la FooBar.d64

eller

und64 la !!blah

OBS! Programmet känner själv av arkivets filtyp och agerar därefter.

1.7 I början fanns UNd64...

...och det finns fortfarande!

Det finns ett annat program i sinnevärlden som har mage att antyda att det får mitt program att verka gammalmodigt, men enligt min mening är mitt program avsevärt stabilare och snabbare. Därför bör du undvika detta andra program.

Jämför själv:

BENCHMARKS:

Extraktion från COMMODORE.D64 med utdata skickat till >NIL:

UNd64 6 sekunder

Brand-X 35 sekunder

En rättvis bedömning är följande: Detta andra program simulerar en 1541-drive bra, speciellt när det gäller läshastighet och det sätt på vilket det handhar cirkulära referenser (Jag känner medlidande med dem som har SCSI-II och en A4000T utrustad med '060 och en stor hårddisk. Programmet ifråga fyller snabbt upp den.).

Dessutom är mitt program mindre.

1.8 .d64/.x64 Support

UNd64 v 38.15 (3.7.97)

© Copyright 1994-1997 Jess Sosnoski

Vad är en .d64 diskimage?

Utdata

.d64-anmärkningar

.x64-anmärkningar

1.9 Vad är en .d64 diskimage?

Vad är en .d64 diskimage?

En .d64-fil är en fil som består av innehållet av en Commodore 1541-formatterad disk, sektor för sektor. Denna filtyp används ofta av PC C64-emulatorer för att spara C64-filer och för att överföra 1541-disketter till ett passande format för användande med emulatorer. (Jag har för mig att emulatorn heter C64S?)

Olyckligtvis kan få C64-emulatorer på Amiga dra fördel av detta. Det finns ett program på Aminet som möjliggör skapande av .d64-arkiv på en 5.25"-drive utrustad Amiga. Med programmet följer också ett program för den omvända operationen. Om jag inte missminner mig finns det att hitta i misc/emu/1541.lha.

Vad är en .x64 diskimage?

.x64 är ett format som används av just X64, en C64-emulator som finns under UNIX. Stöd finns för formaten 1541, 1570, 1571, och 1581.

1.10 .x64-anmärkningar

Vad är då .x64?

 En .x64 diskimage är mer eller mindre ekvivalent med en vanlig .d64 diskimage med en 64-bytes header före vilken beskriver vilken typ av image det är. UNd64 läser headern för att kontrollera att typen stämmer och behandlar sedan filen som vore den en .d64-fil. Se därför .d64 för mer information.

OBS! UNd64 stöder endast 1541/1570 35-track diskimages.

.x64-filer finns i ett otal former, vilka motsvarar 1571, 1581, and 40-track 1541 images. Eventuellt kommer jag att lägga till stöd för dessa i framtida versioner av detta program, men det beror på vilket mottagande jag får vad de nuvarande versionerna beträffar.

1.11 .d64-anmärkningar

.d64-anmärkningar

 Det finns ett flertal typer av .d64-diskimages, nämligen:
 35-track standard, 35-track w/ med extra felinformation (stöds)
 40 & 42 track standard, och 40 & 42 track w/ med felinformation
 (stöds för närvarande inte)

Du har kanske lagt märke till att en del .d64-filer endast innehåller en eller två filer vilka inte verkar kunna användas. Om detta är fallet är det troligen så att filerna är bootkod för ett trackladdande program. Detta betyder att programdata är lagrad på ett icke 1541-dos-kompatibelt sätt; med andra ord så kan du inte göra någonting med filerna.

Om du verkligen behöver dessa filer, kopiera .d64-filen tillbaka till en 1541-diskett varefter du använder 1541:an via A64:s interface. På Aminet finns ett program som kopierar .d64-filer till en 1541-drive: Easy1541.lha. Vidare finns ett antal relaterade program för manipulation av .d64-filer på Aminet i misc/emu.

Försök dock inte att extrahera d/x64 GEOS-disketter - de använder en annan metod för att representera data och UNd64 kommer att rapportera fel filstorlekar. Jag kan inte garantera att data extraheras rätt från dessa images.

1.12 .d64 / .x64 utdata

Programmets utdata:

UNd64 38.15 (3.7.97)
 © Copyright 1994-97 Jess Sosnoski

```
Disk Name:old games          00 2a
  Approx
Blks  Bytes  Type  Filename
-----
   2   508  *PRG  truck and field
  71 18034  *DEL<          track
```

```

97 24638 *PRG<          field
33  8382  PRG    spacedinvaders
33  8382  PRG    knight driver
33  8382  PRG    duvan
33  8382  PRG    wizzard of war
33  8382  PRG    kickinthehead
  1   254  PRG    amigaforever!!
33  8382  PRG    bewareilive
66 16764  PRG    oepirristi
  3   762  *DEL    attack
55 13970  *DEL    .attack
  4  1016  *DEL    puh
33  8382  PRG    ratrace
117 29718 PRG    jumping jacks j2
  0     0  *DEL    froggee ii

```

Förklaring av utdata

Blocks: Antal block filen använde på disketten; ett block = 256 bytes varav 254 används för data.

Approx Bytes: 254*(antal block)

Detta tal ger en approximativ filstorlek - den verkliga storleken är mindre eller lika med givet tal.

OBS! Om någon har ändrat på biblioteksstrukturen och manuellt ändrat antal block så kan ovanstående ge felaktig information.

Type: DEL, PRG, SEQ eller USR

Denna visas endast i informativt syfte och påverkar inte utdata. För de användare som inte använt C64 så är PRG-filer typiskt program (exekverbara filer), SEQ-filer = textfiler, USR-filer kan vara lite av varje och DEL-filer är borttagna filer. Du kan försöka extrahera dessa via valmöjligheten d (DEL-filer markeras alltid som "splat"). Det finns även REL-filer (RELative), men det finns f.n. inget stöd för dem; emellertid kommer programmet att visa att de finns men det är också det enda det gör. REL-filer är ett slags 3d-filer med alla möjliga sidosektorer, listor och annat, så jag valde att inte stödja dem (de kan troligen inte sparas på ett vettigt sätt i ett filsystem som inte stöder denna typ av filer).

Splat-filer har prefixet * och låsta filer har suffixet <. Splat-filer är felaktigt stängda eller borttagna filer, vilka är skriv- resp. delete-skyddade, precis som låsta filer. Dessa fil-attribut påverkar ej programmets utdata.

1.13 Stöd för .t64

UNd64 v 38.15 (3.7.97)

© Copyright 1994-1997 Jess Sosnoski

Vad är en .t64 tapeimage?

Utdata

Anmärkningar

1.14 Vad är en .t64 tapeimage?

Vad är en .t64 tapeimage?

.t64 är ett format utvecklat av Miha Petemel för C64-emulatorn C64s (PC).
Det är helt enkelt ytterligare ett format som emulatorn använder sig av
för att spara C64-filer.

1.15 .t64 utdata

UNd64 38.15 (3.7.97)

© Copyright 1994-97 Jess Sosnoski

Tape Name:imported files
Number of Entries: 30
Number Used : 1
Version : 1.0

Size	Type	Filename
37632	PRG	1a.C64

Förklaring av utdata

Tape Name:
Namnet på arkivet

Number of Entries:
Högsta antal filer som stöds i arkiv

Number Used:
Antal filer som är sparade i .t64-imagern. UNd64 har problem med
att korrekt visa antalet filer i vissa äldre .t64-arkiv, se

.t64-anmärkningar
)

Version:
Versionen på .t64-arkivet.

Bytes: Storleken på filen i bytes

Type: ---, PRG, eller IMG; --- betyder ett tomrum i biblioteket

1.16 .t64-anmärkningar

.t64-anmärkningar

Äldre .t64-arkiv kan medföra vissa problem: filstorleken som visas kan vara fel och vissa arkiv hävdar att det inte finns några filer i dem även om så inte är fallet. Ett annat problem är att de använder filtypen #68 för PRG-filer. Specifikationen för filtyper är att 1 är PRG och 2 och uppåt är IMG. UNd64 kontrollerar om det finns några filer även om arkivet säger att det inte finns det.

1.17 Stöd för .p00

UNd64 v 38.15 (3.7.97)
© Copyright 1994-1997 Jess Sosnoski

Vad är .p00 program images?

.p00 är ett format använt av C64-emulatorn C-64s (PC). De består av ett C64-program med en header innehållande en identifierare och filnamnet. Antagligen skapades detta format för att enklare spara C64-filer på PC då C64-filnamn är 16 tecken långa.

.p00 utdata

UNd64 38.15 (3.7.97)
© Copyright 1994-97 Jess Sosnoski

Size	Filename
33247	logo

Här finns inte mycket som kräver en förklaring.

1.18 Stöd för .lnx

UNd64 v 38.15 (3.7.97)
© Copyright 1994-1997 Jess Sosnoski

Vad är ett .lnx-arkiv?

LyNX-arkiv skapas av ett program för C64 skivet av Will Corley. De består av ett antal konkatenerade filer. UNd64 stöder samtliga .lnx-arkiv.

Observera att vissa tidiga versioner av LyNX kan ha en felaktig fillängd specificerad för den sista filen i arkivet - UNd64 försöker att rätta till detta.

.lnx utdata

UNd64 38.15 (3.7.97)
 © Copyright 1994-97 Jess Sosnoski

Archived With: lynx xv by will corley
 Number of Entries: 2

Size	Type	Filename
27317	SEQ	lynx xv.docs
15744	PRG	library 9.0

Archived With:
 Versionen av LyNX som arkivet är skapad med.

Type:
 Samma som de i .d64; UNd64 kommer emellertid att försöka extrahera REL-filer. Jag är osäker på vad som händer i detta läge, så att om någon ställs inför en liknande situation, rapportera vad som händer.

1.19 Stöd för .lbr

UNd64 v 38.15 (3.7.97)
 © Copyright 1994-1997 Jess Sosnoski

Vad är ett .lbr-arkiv?

LBR står för LiBRary och dessa arkiv är skapade med ett C64-program kallat just Library. De är i mångt och mycket ekvivalenta med LyNX-filer. UNd64 stöder den icke komprimerade versionen av dessa arkiv.

Utdata är som vid .lnx-arkiv.

1.20 Stöd för .sfx

UNd64 v 38.14 (3.7.97)
 © Copyright 1994-1997 Jess Sosnoski

Vad är ett .sfx-arkiv?

SFX-arkiv verkar existera på ett flertal olika datorer, C64 inkluderad. De är självextraherande arkiv; du kan starta dem i din C64-emulator och vänta på att de extraheras, eller så kan du extrahera dem snabbt och enkelt på din Amiga.

UNd64 konverterar DFX-arkiv till lzh/lha-arkiv; använd sedan lha för att göra vad du anser borde göras.

SFX-arkiv finns bl.a. att hitta på ftp.funet.fi.

Både l och d har samma funktion: de visar utdatafilens namn och storlek.

1.21 Felmeddelanden

Felmeddelanden

***Break

Du tryckte på CTRL-C och avslutade programmet.

* Not a recognized archive!!

a.) Filens längd var inte EXAKT 174818 eller 175531 bytes, vilka som bekant är de enda möjliga för en 35-tracks diskett (.d64).

b.) Filheadern innehöll inte tillräcklig information med avseende på arkividentifieringssträngarna (långt ord där ja).

* Circular Link Found, File Truncated

UNd64 hittade en cirkulär referens i ett .d64-arkiv och klippte därför utdatafilen. En cirkulär referens är då pekaren till nästa track och sektor pekar till ett track+sektor som använts tidigare i filen.

* Circular Link Found in Directory

Samma som ovan, förutom att det nu gäller biblioteksstrukturen.

UNd64 avbryter en librarianscan ("biblioteksgenomsökning" om du tycker bättre om det då) ifall en sådan länk uppdagas.

* File size error (n bytes)

I en .d64-fil orsakas detta av en cirkulär link, en felaktig specificering av filstorleken i biblioteksstrukturen eller en felaktig fil (eventuellt fel på disketten).

I en .t64-fil beror det på att en felaktig fillängd finns specificerad i biblioteksstrukturen, dvs. fillängd + filoffset > arkivstorleken. Talet inom parentes anger den riktiga filstorleken.

* Illegal Track or Sector

Tag det lugnt, ingenting är fel på din diskettstation; detta meddelande säger bara att en länk till track+sektor i en fil i .d64-arkivet är felaktig.

En vanlig 15x1:a skulle klaga på samma sätt om du försökte läsa filen ifråga från originaldisketten.

* Illegal Track or Sector in Directory

Samma som ovan förutom att det nu gäller ett bibliotek.

* File Exists! (R)ename (O)verwrite (S)kip (A)bort

Det finns redan en fil med samma namn i biblioteket.

R - Döp om filen som skall extraheras

O - Skriv över filen i biblioteket

S - Extrahera inte filen i arkivet

A - Avbryt programmet (motsvarar CTRL-C)

* Not a 35-track 1541/1570 .X64 image!

.x64-formatet ifråga stöds inte av und64.

* Files pointer points past EOF

Arkivet är inte komplett.

*** Cant Read Input File! (Kan ej läsa indatafil)

```

*** Cant Write Output File!   (Kan ej skriva utdatafil)
*** Couldn't Allocate Memory!  (För lite minne)
*** Error Reading Input File!  (Fel på indatafil)
*** Error Writing Output File! (Fel vid skrivning av utdatafil)
*** Corrupt Archive!          (Arkivet är ej komplett/felaktigt)

```

Ovanstående felmeddelanden är alla ödesdiga såtillvida att programmet genast avslutas. Oftast kommer en requester att uppenbara sig innan dessa meddelanden skrivs ut.

1.22 Kompatibilitet

Kompatibilitet

```

-----
Detta program är utvecklat på
    Den före detta 'Amiga 600 from hell'
    Programmet är testat på följande konfigurationer:
A600  2Chip      1.3/1.3
A600  2Chip 4Fast  2.05/2.1
A600  2Chip 4Fast  3.1/3.1
A600  2Chip 8Fast  3.1/3.1
(25 Mhz '020 & 882)
A1000 512K Chip  1.1,1.2,1.3
A3000 2Chip 8Fast 2 16BitFast 2.04/2.1, 3.1/3.1
(25 mhz '030 med massor av SCSI device)

```

(Översättarna: Det fungerar finfint på Puhs A1200 med 28 MHz B1220, 2+4, 3.1)

Tro det eller ej - programmet fungerade på samtliga! Bugrapporter tas dock tacksamt emot. (Jag är medveten om att det kräver ca 180 Kb ledigt minne för att starta.)

1.23 Programmets utveckling

Programmets utveckling

```

-----
37.1  Första officiella versionen. Skriven och kompilerad i
      PCQ Pascal 1.2d (PD-versionen), assemblerad med A68k 2.71,
      disassemblerad med IRA 1.02, återassembled och optimerad
      med PhxAss 3.30, hundraprocentigt länkad med PhxLnk 2.03
      och hunkmergad med Imploder 4.0 (ni vet den med allt ljud).
      Programmet kan utföra ungefär vad det säger sig kunna
      utföra.

```

Storlek: 9340 (19.4.94)

```

37.2  Lade till cirkulärlinkskontroll och fixade den äckliga bugg
      som gav sig tillkänna genom att den fråntog oss alla
      möjligheten att avbryta en extraktion med CTRL-C.
      C-l-kontrollen innebär emellertid att extraktionen slöas

```

ned en aning.
Fixade buggen med REL-filer; UNd64 försöker inte längre lista eller extrahera dessa små elakingar.
Förbättrade kontrollen för ogiltiga filnamn.
Skrev om delar av dokumentationen.

Storlek: 10080 (21.4.94)

37.3 Snabbade upp cirkulärlinks kontrollen i det att den numera tar lika lång (eller kort) tid på sig oavsett omständigheterna.

Lade till filnamnskollisionskontroll.

Förbättrade kontroller efter ogiltiga filnamn.

Skrev om delar av dokumentationen igen.

Storlek: 10684 (27.4.94)

37.4 Numera är UNd64 mer eller mindre helt assemblerad med SNMA 1.39 istället för A68K; på samma sätt åter-assemblerar och optimerar jag det med Phxass 3.71, vilket gör hanteringen lättare och programmet inte bara snabbare utan även kortare.
Fixade så att man skulle kunna handhava både deletade filer och splat-filer, och lade till möjlighet att estimerar filstorleken på vad som extraherades.

Storlek: 9944 (4.5.94)

37.5 Assemblerad med SNMA 1.45; lämnade Phxass bakom mig då detta bara orsakade GURUs, och sånt har man inte tid med när timmen är sen och humöret lite segt.
Lade till filkontroll samt stöd för låsta filer; fixade ett par buggar och skrev om koden lite.
Utdatan är numera mer givande.

Storlek: 11888 (10.6.94)

37.6 Assemblerad och optimerad med SNMA 1.97.
Jag skrev om läs- och skrivrutinerna för att få (och fick!) en KLAR förbättring vad avser hastighet. Skrev om delar av dokumentationen igen.

Storlek: 12168 (3.1.95)

37.7 Lade till arkivfilstorlekskontroll, vilket innebar att UNd64 nu kan erbjuda så gott som fullständig igenkänning och behandling av .d64-arkiv.

Storlek: 13000 (28.1.95)

38.0 Skrev om läs- och skrivrutinerna (och vi pratar skrev OM) i det att jag ville få dem AVSEVÄRT snabbare.
Skrev om och snyggade till koden, vilket innebar att jag samtidigt blev av med ett par buggar.
Gjorde en guide av informationen (och jag gjorde det med en texteditor, vilket inte var helt kul).
Tidigare versioner av UNd64 hade svårt att ange korrekt

fillängd, men det skall vara historia nu. Vidare hade det kommit till min kännedom att programmet ibland strulade till det med "space" i filnamn, och även denna oangelägenhet sanerades.

Nu går det bra att ändra mottagarbibliotek igen.

Fixade till dokumentationen.

Programmet skickar nu resultaten till >NIL:.

Förbättrad utdata. Kan nu lista (men inget mer) REL-filer.

Låsta filer markeras som just låsta och splat-filer som splat-filer. (filattribut stämmer nu alltså överens med dem på en riktig 1541)

Felkontroller här och var var lite slarvigt skrivna, men där gick jag fram med järnhand.

Lade till stöd för .t64 och .p00; de båda formaten känns nu av automagiskt.

Nästan helt och hållet assemblerat och optimerat med SNMA 1.97, länkat med PhxLnk 4.03, disassemblerat med IRA 1.02, återassemblerat och ännu mer optimerat med PhxAss 4.10.

Kompilerat med 3.1 PCQ-Includes.

Storlek: 16028 (14.3.95)

- 38.1 Fixade en bug relaterad till parsningen av kommandoraden, vilken krävde att man specificerade en annan valmöjlighet för att avstå från filnamnskonvertering.

Assemblerat med SNMA 1.99, återassemblerat med PhxAss 4.18 och länkat med PhxLnk 4.17.

Storlek: 15960 (5.7.95)

- 38.2 Skrev om funktionen som kontrollerar om en fil redan existerar, delar av filstorleksfunktionen, koden för extraktion, koden för "track parsing" av bibliotek och kod relaterad till track- respektive sektorlänkning. Parsing av filsektorer rapporterar numera felaktiga track och sektorer.

Parsing av bibliotek rapporterar numera felaktiga track och sektorer, och detekterar dessutom cirkulära länkar.

Lade till utdatasökväg (vilket ju kan visa sig rätt så användbart i sammanhang som dessa).

Lade till fler felmeddelanden.

Assemblerat med SNMA 2.03, återassemblerat med PhxAss 4.23.

Storlek: 19692 (24.10.95)

- 38.3 Filestimeringen felade ibland i det att den inte frigjorde ett visst filinfoblock i situationer då detta skulle göras. Fixade också en misstänkt bugg i filkollisionsfunktionen. Rensade bort lite uråldrig kod och gjorde lite pytteoptimering.

Storlek: 19600 (28.10.95)

- 38.4 Slängde lite överflödiga kod och genomgick ytteligare en batalj med filstorleksfunktionen (igen! :<); om den rätt osannolika situationen inträffade att programmet lyckades

läsa en fil, allokerade ett filinfoblock och sedan inte klarade av att undersöka det tillfredsställande, ja då frigjordes inte detta filinfoblock som det var tänkt...

Sök och du skola finna; fixat en gång för alla. Optimerade lite smått och gott här och var (översättarna vill påpeka att "biscuit" snarast betyder småkaka, men inte sjutton kan man optimera dem?)
Assemblerat med SNMA 2.04 och Phxass 4.25.
(koden är faktiskt mindre nu än i 38.2 och 38.3!)

Storlek: 19376 (26.11.95)

38.5 Lade till datumfunktion.

Assemblerat med Phxass 4.26 och länkat med Phxlnk 4.20. (Hoppas att jag) fixade alla mungwallhits och kraschar som uppstod när för långa sökvägs- eller filnamn användes. Nu accepteras upp till 255 tecken i ett filnamn. Det ni!

Storlek: 19372 (13.1.96)

38.6 Ändrade datumfelet i den föregående versionen av dokumentationen (svårt fel, det där, att inte komma ihåg vilket år man lever i).

Lade till stöd för .x64 disk images (1541/1570). Klarar nu att extrahera .t64 IMAge-filer.
(kontakta mig vid problem)

Storlek: 20164 (17.2.96)

38.7 Gjorde koden modulär genom att separera diverse arkivhanterare. Detta gjorde dessutom koden en aning snabbare och läsligare.

Läser nu in arkiven i ett stycke, varefter den bearbetar det hela.

Fixade två buggar i .t64-koden: om en fils offset var större än arkivets längd fungerade det inte särskilt bra tidigare (eller gjorde det det, vad vet jag?), och om en fil inte kunde skrivas tillfredsställande stängdes endast indatafilen. Fixade en bugg i filkollisionsfunktionen.
Assemblerat med PhxAss 4.30 och länkat med PhxLnk 4.23.

Storlek: 19940 (7.5.96)

38.8 ÄNTLIGEN fick jag tummen ur och lade till stöd för .lnx.

Några småoptimeringar gjorde det hela aningen snabbare. Assemblerat med Phxass 4.32.

Storlek: 22636 (27.5.96)

38.9 Karvade loss 300 bytes av överflödigt kod. Ha!

Fixade en RIKTIGT ELAK bugg (den kan knappast ha funnits där innan jag uploadade arkivet?) som kraschade datorn när man ville lista/testa en .d64/x64 disk image.

Storlek: 22344 (20.6.96)

38.10 Inga värre ändringar direkt. Stöd för .d64-filer med medföljande felinformation (vilken ignoreras). Dessutom ett par VERKLIGT små optimeringar.

Storlek: 22236 (17.9.96)

38.11 Ändrade det sätt på vilket .d64 inläses; listningen tar numera en sekund längre men är mycket mer pålitlig då hela imagen läses in. Lade dessutom till igenkänningskod för ytterligare ett .t64-ID.
Gjorde inläsning av arkiv (dock ej .d64, se ovan) säkrare genom att låta programmet kolla ID snarare är filstorlek först. Tog versionrestriktioner från .t64 och Lynx-moduler. De sistnämnda stöder nu punter-lynx-arkiv. Adderade stöd för .lbr.
UNd64 kollar nu .t64/.p00 efter fel.
Några mindre optimeringar.
Fixade dokumentationen; avlägsnade gammal information.
Ytterligare en bugg i filkollisionsrutinen (denna gång vid namnändring) fixades.

Storlek: 23800 (10.10.96)

38.12 Lade till stöd för .sfx-files, konvertering till lzh/lha-kompatibla arkiv kan nu ske.
Phxass 4.33 assemblerade.

Storlek: 24516 (19.10.96)

38.13 Omstrukturerade guiden.
Skrev om en hel del av programmets huvuddel och .d64-modul, vilket gjorde att vi nu kan slänga den där 180 K+ BSS-hunken åt... Nu kommer programmet inte längre allokeras 180 K varje gång du startar det; det avsätter endast minne efter behov! :)
Detta gör inläsningen av .d64 ett par millisekunder snabbare p.g.a. det sätt på vilket minnet nu används.
Fixade en bugg då pekaren till nästa track/sektor pekade på 0,0, vilket medförde att programmet rapporterade skrivfel.
Avlägsnade detektering av cirkulära länkar i biblioteksstrukturen.
Fixade en otrevlig bugg i .p00-modulen då ett arkiv undersöktes - programmet rapporterade ibland att en fil redan existerade även om ingen extraktion försiggick.
Använde PhxAss 4.34 och PhxLnk 4.30 i skapandets process.

Storlek: 24601 (10.2.97)

38.14 Omstrukturerade källkoden
Viss optimering.
Fixade bugg relaterad till CBMASCII->ASCII-konvertering, vilken inte konverterade alla tecken till ASCII.
Lade till ett andra pass för att ta bort överskjutande mellanslagstecken om konverteringen av otillåtna tecken genererade dessa.

Lade till stöd för RLE-komprimerade l!zipkodade filer!
Lade till fixp00 för .p00-rutinen.
Handhavandet av .X64-arkiv omfattar nu även filkommentarer.
Använder numera Thor softwares HUNK-program för att
processa und64 före distribuering (ger mindre program).
PhxAss 4.35 användes för att assemblera.
Storlek: 25852

38.15 Rekompilerad med den till Aminet senast uppladdade
versionen av PCQ pascal.
Assemblerad med PhxAss 4.37beta.
Använder nu vissa 2.x-funktioner isf. egna dito
Ersatte StrtoInt-funktionen med motsvarande i dos.library
Strängkopieringsrutinen fungerade inte alltid så bra,
men det skall vara historia nu.
Optimerade handhavandet av zipcode-filer.
Ändrade det sätt på vilket programmet behandlar sökvägar.
Storlek: 25984 (3.7.97)

1.24 Kvar att göra

Kvar att göra

- Skriva om denna guide
- Resterande saker att göra är för tillfället smått hemliga...

1.25 Upphovsmannen

Upphovsmannens adresser

Jess Sosnoski
651 Hillside Drive
Mount Carmel, PA 17851-2463
USA

Min emailadress:
starblaz@postoffice.ptd.net

IRC Nick: starblazr
vanligtvis på:

Dalnet: oftast då #nin och #miggyl el. andra Amiga-kanaler.

BeyondIrc: Oftast då #amirc el. #styx

Galaxynet: Oftast #amichat

Min hemsida, om du vill kalla den så då den inte på något sätt är klar, kan
du hitta på:
<http://home.ptd.net/~starblaz/>

Om du känner ett brinnande behov av att skicka mig pengar, presenter eller en A4000 Tower, tackar jag och tar emot.

```
  .  
 /|  
/_|  
_/ _|MIGA... Muuuuu!
```

1.26 Också ett citat

"Korrekturläst av Näringsdepartementet", stod det på pappret. Illa.

"Vem det är? Jag vet inte, faktiskt. Det kanske är prinsen Rolle?"
- Alvar Andersson

1.27 Den före detta 'Amiga 600 From Hell'

Den före detta "Amiga 600 From Hell":
Kickstart/Workbench 3.1 (Kickstart 40.63 för att vara exakt)
Apollo A620 68020/68882 25Mhz Accelerator
8 MB 32-Bit Fast ram
2 MB Chip

Ovanstående var 'The 600 from hell' tills jag träffade en snubbe som hade en 600 Tower. Så nu är den inte på topp längre, och lådan kommer aldrig att kunna stängas riktigt igen.

1.28 Tack till

Tack till:

Kay Drangmeister
för buggrapporter, källkod för readargs och testning på '060.

Jouko Valta, upphovsman till X64
för att ha skickat några .x64-filer för testningsändamål.

Michael P. Steer
för källkoden för zipkodade filer.

Johan "S.Duvan" Alfredsson och Henrik "Puh" Nilsson
Oepir Risti Translations (d95duvan@dttek.chalmers.se) (alvar@one.se)
för den svenska versionen av denna guide.

Mitch B. Parker
För att han inspirerade mig till att komma igång med UNd64.

Dave Pedergnana
Utan denne kille skulle jag aldrig blivit så intresserad av datorer.

Amiga Technologies/International och Gateway 2000
för att de återupplivat Amigan.

Alla som skickat mig idéer, email och bugrapporter.

Alla som registrerat sig - 27 hittills.

Alla som spridit programmet.

Om du någonsin har skickat ett email med buggrapport/idéer och ditt namn
inte återfinns ovan, och det är din önskan att så skulle vara fallet, maila

mig

.

1.29 Stöd för 1!zipkodad RLE-komprimerad split diskimage

Till slut fann jag tid att lägga till stöd för denna typ av arkiv;
detta för att formatet numera är vanligt förekommande.

Zipkodade filer har ingenting att göra med .zip-kompression, vilken används
flitigt på PC och UNIX. Istället är de partitionerade komprimerade
diskimages.

Det finns fyra filer i varje ziparkiv, vilka har prefixen 1! 2! 3! och 4!
När und64 ställs inför en zipkodad fil behöver du bara specificera den
första delfilen.

Följande måste gälla för zipkodade arkiv; de fyra filerna måste ha
nedanstående karakteristika:

Arkiv	Minsta storlek	Största storlek
1!	508	43348
2!	506	43346
3!	518	44378
4!	527	45152

Und64 laddar filerna för att konvertera dem internt till en .d64-fil.
För tillfället finns ingen valmöjlighet att explicit konvertera zipkodade
filer till .d64 till disk.

De typer av kompression som zipkodade filer stöder är:

1. Direkt data (ingen kompression)
2. Statisk (fylld med en specifik byte)
3. RLE-komprimerad data

1.30 Översättarnas kommentarer

Översättarnas funderingar

Vi har ingalunda försökt att skriva en komplett datormanual, utan endast översatt och i viss mån kompletterat den ursprungliga texten. Precis som i den engelska texten förutsätts en viss datorkunskap, och vi anser inte, att man skall ändra i en text som inte är ens egen.

Det primära syftet med översättningen är att förmedla UNd64s manual till dem som av någon anledning föredrar svensk text framför engelsk. Vi påstår på intet sätt att vår översättning är den optimala, eller att våra ord är lag; vi vill bara ge en översiktstext för dem som kanske annars skulle avstått från att använda UNd64.

Vi har försökt att återge texten på en komplett och korrekt svenska. Många manualer - och denna med dem - innehåller inte fullständiga meningar, t.ex "Version 1.22 - fixed the PCIP bug.", och dylika meningar har vi översatt rakt av, d.v.s. utan att försöka göra en hel mening av en halv.

Vi gör inte heller anspråk på att vara perfekta; det kan mycket väl finnas fel och eller oklarheter i vår text. Dessa ber vi om ursäkt för. Emellertid anser vi att det är bättre att göra så gott man kan utifrån vad man har att tillgå än att inte göra någonting alls.

Alla dementier (disclaimers) är översatta till svenska för att ge en inblick i vad det hela handlar om. Vi hänvisar emellertid till de engelska originalen för närmare studier, då man egentligen inte skall hålla på och översätta dylika texter hur som helst (d.v.s. utan att författaren sitter bredvid). Tag alltså de svenska dementierna som en måttstock, och gör dina riktiga mätningar på de engelska.

Denna översättning har vi gjort utan att få någon som helst ersättning. Det är vår tro att man skall ställa upp för dem som har gjort någonting bra, och att man mycket väl kan göra det utan att pengar är med i leken. Vi hoppas nu, att vårt jobb kommer att falla er i smaken, och att ni mailar era eventuella synpunkter till oss. Vi är även öppna för diverse spännande förslag.

Askim, februari 1997, uppdaterad maj 1997,

Oepir Risti Translations in 1997 - Project 2

Johan "S.Duvan" Alfredsson (d95duvan@dtek.chalmers.se)
Henrik "Puh" Nilsson (alvar@one.se)