

**DiskSpare**

**COLLABORATORS**

	<i>TITLE :</i> DiskSpare		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		February 6, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>DiskSpare</b>	<b>1</b>
1.1	DiskSpare.guide . . . . .	1
1.2	copyright . . . . .	2
1.3	kurzinfo . . . . .	2
1.4	installation . . . . .	3
1.5	flags . . . . .	4
1.6	änderungen . . . . .	5
1.7	mountlist1 . . . . .	6
1.8	mountlist2 . . . . .	7
1.9	device1 . . . . .	8
1.10	device2 . . . . .	9
1.11	welche probleme? . . . . .	10
1.12	tips & tricks . . . . .	13
1.13	adresse . . . . .	15
1.14	information . . . . .	15
1.15	pfs . . . . .	16

---

# Chapter 1

# DiskSpare

## 1.1 DiskSpare.guide

DiskSpare.Device (DSD) V 3.0 copyright 1992-1994 by Klaus ↔  
Deppisch

KurzInfo

Tips & Tricks

Information

Installation

Flags

Programmers

Mountlist allgemein

Änderungen

@{ " V36NoClick " Link "Support/V36NoClick.dok/main"}

Mountlist und DiskSpare

Probleme

@{ " QuietTD " Link "Support/QuietTD.dok/main"}

Device allgemein

Adresse

PFS

DiskSpare im Besonderen

Copyright

---

## 1.2 copyright

Copyright

~~~~~

Das Device ist FREEWARE. D.h. es darf frei kopiert werden, solange es in KEINSTER Weise verändert wird und solange ALLE dazugehörigen Files mitkopiert werden.

Das Device darf auch im Zusammenhang mit anderen, auch kommerziellen Programmen verwendet und vertrieben werden, solange KLARGESTELLT ist, daß es sich bei dem Device um FREEWARE handelt UND solange ALLE Files mitkopiert werden. (see also support/diskspare.doc)

Eine Reassemblierung des Devices ist selbstverständlich NICHT gestattet.

Falls Sie eine erweiterte oder spezielle Version benötigen, setzen Sie sich bitte mit mir in Verbindung.

Mein Diskformat möchte ich auf keiner Diskette mit einem hohlen NDOS Spiel finden !!!

AM WICHTIGSTEN:

Die Benutzung des beschriebenen Programms erfolgt AUSSCHLIEßLICH auf eigenes Risiko.

Der Autor kann auf KEINEN FALL für einen Schaden oder Datenverlust der direkt oder indirekt mit dem Gebrauch des beschriebenen Programms entstehen sollte verantwortlich gemacht werden.

Alle Rechte vorbehalten. Für Fehlermitteilungen oder Verbesserungsvorschläge bin ich (fast) jederzeit dankbar.

## 1.3 kurzinfo

KurzInfo

~~~~~

DiskSpare ist eine Ergänzung oder Ersatz für das trackdisk.device. Es ermöglicht es Ihnen mehr Daten, schneller als bisher auf Ihren Disketten unterzubringen. Dazu wird ein eigenes Aufzeichnungsformat verwendet, also keine Kompression oder ähnliches. Kapazität:

- 984K auf DD Disketten (normalerweise 880K)
- 1968K auf HD Disketten (normalerweise 1760K)

Das diskspare.device benötigt Kickstart Version 2.0 oder höher.

Wenn Sie DiskSpare testen wollen, klicken Sie einfach auf das

---

QuickTest-Icon. Damit wird ein Laufwerk mit 984K auf Unit 0 (DF0) gemountet. Zum Formatieren benötigen Sie dann allerdings die Shell. Geben Sie ein: 'format drive DS0: name Test <return>'. Wenn das Laufwerk beim Booten gemountet wird, können Sie Disketten aber ganz normal über die Workbench formatieren. Das ist bei jedem Laufwerk so, daß Sie nach dem Laden der Workbench mounten. Nach dem Formatieren können Sie die Diskette wie trackdisk Disketten benutzen.

## 1.4 installation

### Installation

~~~~~

Der einfachste Weg DiskSpare auf Ihre Festplatte bzw BootDisk zu kopieren, besteht darin das Installations Script zu verwenden. Klicken Sie einfach auf das 'Deutsch'-Icon um es zu starten. Sie müssen dann nur noch entscheiden, welche Laufwerke Sie installieren möchten und ob es HD Laufwerke sind. Das Script erkennt auch, welche Workbench Version (2.0 oder höher) Sie verwenden.

Das Script geht davon aus, daß eine komplette Workbench zur Verfügung steht. D.h. wenn Sie eine BootDisk verwenden, bei der das halbe 'C' Verzeichnis gelöscht ist, kann es Probleme geben.

Eine 'von Hand' Installation ist aber auch kein Problem:

- Kopieren Sie einfach das 'diskspare.device' in Ihren Devs: Ordner.
- Ab Workbench 2.1 brauchen Sie dann nur noch die gewünschten Mountfiles nach Devs:DOSDrivers zu kopieren und Ihren Rechner neu zu booten, das Laufwerk wird dann automatisch eingebunden.
- Bei Workbench 2.0 haben Sie leider etwas mehr Arbeit. Sie müssen die 'Mountlist' Files aus dem DD bzw HD Ordner mit einem Editor Ihrer Wahl an Ihre Mountlist im Devs: Ordner anhängen.

ACHTUNG: kopieren Sie NIEMALS irgendeine Mountlist einfach in Ihren Devs: Ordner. Sie überschreiben damit nämlich Ihre eigene Mountlist. Das kann dazu führen, daß dann auf Ihrem Rechner fast nichts mehr funktioniert.

Wenn Sie die Mountlist ANGEHÄNGT haben, müssen Sie noch die User-Startup (im S: Ordner) ergänzen. Fügen Sie die Zeile 'Mount DSX:' (X ist 0-3) und die Zeile 'Addbuffers DSX: >NIL:' mit dem Editor ein. Beim nächsten Booten wird jetzt das Laufwerk eingebunden.

Mit dem Install Script werden Laufwerke mit 984K (DD) bzw 1968K (HD) installiert. Wenn Sie Einstellungen ändern wollen, müssen Sie das entsprechende MountFile bzw den Mountlisteintrag ändern. Die Größe des Laufwerks richtet sich nach der Anzahl der eingestellten Zylinder. Sie kann über die

Flags  
verändert

werden.

## 1.5 flags

Flags

~~~~~

Bit	Wert	Bedeutung
	wenn	gesetzt

0 1 Bit ist 0: Es wird nach dem Mounten beim entsprechenden DFX Laufwerk kein Diskwechsel mehr erkannt.  
 Bit ist 1: Der Diskwechsel funktioniert sowohl bei DSX als auch bei DFX.

Das Bit darf nur bei Verwendung des FastFile-Systems gesetzt werden !

1 2 Bit ist 0: Es werden 80 Zylinder benutzt. (960K)  
 Bit ist 1: Es werden 82 Zylinder benutzt. (984k)

Beachten Sie bitte, daß bei mehr als 80 Zylindern das trackdisk.device gepatched werden muß !  
 Zusätzlich zum Bit setzen, muß auch der Eintrag 'HighCyl = ' auf den entsprechenden Wert gesetzt werden.

2 4 Bit ist 0: Es wird kein Wechsel zwischen HD und DD Disketten erkannt. Es wird nur die in der Mountlist angegebene Größe unterstützt.  
 Bit ist 1: Der Wechsel zwischen HD und DD Disks wird korrekt erkannt.

Das Bit darf nur bei Verwendung des FastFile-Systems gesetzt werden ! Wenn es auf 1 gesetzt ist, wird das FFS gepatched !

3 8 Dieses Bit hat bei dieser Deviceversion keine Bedeutung mehr. Ist aber trotzdem reserviert.

4 16 Bit ist 0: Normaler Schreibmodus  
 Bit ist 1: Es wird nach JEDEM Schreibzugriff der Track überprüft.

Zur Beachtung:

Mit dem Setzen dieses Bits halbieren Sie zwar die Schreibgeschwindigkeit, Sie haben dafür aber die Gewißheit, daß die Daten auch wirklich auf der Diskette angekommen sind. Speziell bei der Verwendung von billigen Disketten sinnvoll.

Um die Flags zu ändern, müssen Sie nur noch die Werte der Bits

die Sie auf 1 setzen wollen addieren, und in die 'Flags = ..' Zeile der Mountlist eintragen. Gelöschte Bits haben den Wert '0'.

Beispiel:

Sie wollen HD Disketten mit 82 Zylindern benutzen und der Diskwechsel soll auch erkannt werden. Dazu müssen Sie Bit 0, 1 und 2 setzen. Sie addieren also die Werte die hinter den Bitnummern stehen:  $1 + 2 + 4 = 7$  und haben somit den neuen Wert für die 'Flags = ..' Zeile.

## 1.6 änderungen

Änderungen

~~~~~

Version 3.0

Die augenfälligste Änderung betrifft sicherlich das DOC File. Jetzt für AmigaGuide.

Am Device wurde aber auch einiges geändert:

- komplett überarbeitet
- deutlich schneller (besonders auf langsamen Amigas)
- Verify Write zuschaltbar. Eine Idee von Kriton Kyrimis.
- es wird kein FastrAM mehr verwendet. Dafür aber mehr ChipRAM und der Blitter.
- Der DMA Buffer wird jetzt Diskettenabhängig allokiert. Keine Speicherverschwendung mehr bei DD Disketten.
- Der HD Modus funktioniert jetzt (deviceseitig) immer. Konnte früher (blödsinnigerweise) völlig abgeschaltet werden.
- schon wieder Fehlerkorrekturen

Version 2.2

- funktioniert jetzt auch mit dem PFS von Michiel Pelt. Mountfiles für PFS befinden sich im DD bzw HD Ordner. Die Namen PS0 und PS1 hat sich Brett Simcock ausgedacht.
- noch einige Fehlerkorrekturen

Version 2.1

- hatte leider einen Fehlstart
-

Version 2.0

- die erste diskspare.device Version mit eingebautem HD Modus.  
Das 'highdensity.device' von mir hat damit ausgedient.

Version 1.6

- erste Veröffentlichung.

## 1.7 mountlist1

Mountlist allgemein

~~~~~

Zu jedem Laufwerk gehört eine sog. Mountlist. Das ist eine Textdatei in der die Laufwerkskenngrößen festgelegt werden. Da mit dieser Mountlist wichtige Größen auch falsch eingestellt werden können, will ich hier einige erklären.

Device = ... Hiermit geben Sie an, welches Device beim Lesen und Schreiben angesprochen werden soll. Es muß sich um ein (mehr oder weniger) trackdisk.device kompatibles handeln.

Unit = ... Hier steht eine Zahl von 0 bis X. X ist device-abhängig. Bei diskspare und trackdisk ist X = 3.  
Bei Festplatten Devices kann Sie auch höher sein.

Flags = ... Diese Flags sind die einzige Möglichkeit dem Device irgendwelche Einstellungen vorzugeben (ohne Hilfsprogrammchen). Sie werden beim diskspare.device auch ausgiebig genutzt, sind aber deviceabhängig.

Surfaces = ... Dies sind die Anzahl der 'Oberflächen' = Köpfe.  
Sie sind device- und hardwareabhängig.

BlockSize = ... Es müßte eigentlich SectorSize heißen, da ab KS 3.0 auch mehrere Sektoren zu einem Block zusammengefaßt werden können. Auf jeden Fall gibt der Wert hier an, wieviele Bytes ein Sektor hat (üblicherweise 512).

BlocksPerTrack = ... Müßte auch SectorsPerTrack heißen. Gibt an, wieviele Sektoren auf einer Spur zu finden sind.  
Ein Track ist eine Spur mit einer Oberfläche,  
ein Zylinder ist eine Spur auf allen Oberflächen.  
-> Anzahl Tracks = Zylinder \* Surfaces

Reserved = 2 Dieser Eintrag gibt an, wieviele Blocks für Bootblocks reserviert werden sollen. Er MUß beim FFS AUF JEDEM FALL 2 (zwei) sein.

Interleave = ... Gibt den Sektorversatz im Track an. Bei den AmigaFloppies ist er aber uninteressant, da hier

sowieso der ganze Track gelesen und geschrieben werden muß.

LowCyl = ... Der Wert gibt den untersten Zylinder an, der vom Filesystem benutzt werden soll.

HighCyl = ... Hier steht der oberste nutzbare Zylinder des Laufwerks. Wenn ein Device 80 Zylinder zur Verfügung stellt, und Sie alle Zylinder nutzen wollen, ist der max. Wert für HighCyl = 79, wenn der LowCyl = 0 ist, da beim Computer immer bei 0 zu zählen begonnen wird.

Buffers = ... Hier steht wieviele Buffers sich das FFS für das Laufwerk reservieren soll. Das absolute Minimum sollte 50 sein. Bei zu kleinen Werten, wie z.B. 5 (Default für DFX) sinkt die Geschwindigkeit SEHR stark ab. Wenn Sie genug Speicher haben und viel mit Disketten arbeiten, kann der Wert ruhig > 50 sein.

StackSize = ... Wenn Sie ihn zu klein wählen passieren die 'tollsten' Sachen. Beim FFS sind 600 die Vorgabe, bei anderen FileSystemen kann er auch größer sein. KEINESFALLS den Vorgabewert unterschreiten.

DosType = ... Er gibt dem Filesystem genauere Angaben, wie es die Daten auf den Blöcken organisieren soll.

Mount = 1 Dieses Flag ist auch sehr wichtig. Mit '1' stellen Sie ein, daß das Laufwerk SOFORT und KOMPLETT eingebunden wird. Wenn Sie die 'Mount =' Zeile weglassen, wird zwar eine (betriebs-systeminterne) Struktur angelegt, das zugehörige Filesystem und Device aber gar nicht geladen. Dieses Verhalten stammt wahrscheinlich noch aus der 256K Amiga-Zeit. Auf jeden Fall kommen einige Programme nicht damit zurecht, daß das Laufwerk nur scheinbar da ist.

## 1.8 mountlist2

Mountlist und DiskSpare

~~~~~

Der Devicename muß mit 'Device = diskspare.device' angegeben werden.

Die Unitnummer kann von 0-3 eingestellt werden, und entspricht dann dem jeweiligen DFX Laufwerk (Unit0 = DF0 usw).

Die

Flags

stellen Sie nach Ihren eigenen Vorstellungen ein.

Die tatsächlichen (physikalischen) Werte sind für:

```
Surfaces      2
SectorSize    512
SectorsPerTrack 12 (DD) bzw 24 (HD)
LowCyl        0
HighCyl       79 oder 81 (je nach Flags)
```

Wenn das AutoDetect-Bit (2) in den Flags gelöscht ist, können Sie (beim FFS) für die oben genannten Kenngrößen auch andere Werte einstellen. Sie müssen nur sicherstellen, das die Gesamtkapazität stimmt. Ein Wechsel von DD und HD Disketten ist dann aber nicht mehr möglich ! Ein Vergrößern des SectorSize (und Verkleinern des SectorsPerTrack) ist aber eigentlich nur zum Testen Ihres Diskettenmonitors zu gebrauchen, (die meisten versagen bei anderen Werten als 512 Bytes/Sector VORSICHT!!) schneller wird das diskspare.device dadurch jedenfalls nicht.

Dafür verschwenden Sie damit aber jede Menge Platz auf der Disk. Dazu ein Beispiel: Sie stellen den SectorSize auf 2048 ein. Jetzt speichern Sie mit 'Ed' einen Text mit einem Buchstaben ab. Damit haben Sie 2048 Bytes (nur für die Daten) auf der Disk verbraucht. Das entspricht einem Wirkungsgrad von 0,05%. Das ist zwar zugegeben worst\$^2\$ case, aber von welchem File läßt sich die Länge schon durch 2048 teilen ?

In dem Zusammenhang lohnt es sich übrigens, das MSDOS FileSystem (das mit den langen Filenamen) mal anzuschauen, was (sich) dieses auf großen Festplatten leistet, ist wirklich beeindruckend. Das FFS spart zwar auch nicht gerade Platz auf der Diskette, aber dafür ist es UNMÖGLICH eine FFS Diskette mit ein paar Read/Write Errors völlig unbrauchbar zu machen, man hat also wenigstens etwas von dem hohen Platzverbrauch.

Die anderen Werte in der Mountlist betreffen das Filesystem und sollen daher hier nicht interessieren.

## 1.9 device1

Device allgemein

~~~~~

Da wohl nicht jeder weiß, was ein Device macht und vor allem, was es NICHT macht (interessiert ja auch nicht unbedingt, solange alles klappt) möchte ich hier einen kleinen Einblick in die Funktion eines Devices geben.

Es ist eigentlich sehr simpel. Ein Device stellt eine definierte Schnittstelle zu irgendeiner Hardware dar. Es stellt also dem Betriebssystem (im wesentlichen) Funktionen zum Lesen und Schreiben von bzw auf diese Hardware zur Verfügung. Das muß JEDES Device können. Soll ein Device von einem Filesystem zur Datenspeicherung benutzt werden, müssen weitere Funktionen zur Verfügung gestellt werden, z.B. Schreib-/Lesekopf positionieren, Motor ein- und ausschalten usw. Sie sehen schon, sehr kompliziert ist das gar nicht.

Wenn also ein Filesystem auf ein Device zugreift, wie das beim diskspare.device der Fall ist, besteht die Aufgabe des Devices in der Hauptsache darin, Sektoren zu lesen und zu schreiben. Was in den Sektoren an Daten steht ist AUSSCHLIEßLICH Sache des Filesystems. Das Device hat keine Ahnung davon, ob gerade ein Directory oder ein Ballerspiel eingelesen wird. Diese Aufteilung hat den Vorteil, daß so ohne Probleme verschiedene Filesysteme genutzt werden können. Und natürlich auch verschiedene Device.

Der Weg für Daten aus dem Speicher auf Diskette sieht so aus:

Anwenderprogramm oder Shell will Daten schreiben

```
dos.library      schaut nach, welches Filesystem
                  für das gewünschte Laufwerk zu-
                  ständig ist und übergibt ihm den
                  Auftrag des Anwenderprogramm bzw
                  der Shell (ist das gleiche).
```

```
Filesystem      überprüft ob noch Platz auf der
                  Disk ist, trägt den Namen ins
                  Directory ein, schaut nach
                  welches Device für das Laufwerk
                  zuständig ist und übergibt ihm
                  den Auftrag ab Block sowieso
                  soundsoviel Blocks an Daten
                  hinzuschreiben. SEHR vereinfacht!
```

```
Device          überprüft ob die Sektornummern
                  im gültigen Bereich liegen und
                  schreibt dann die Daten auf die
                  Diskette.
```

Das ist (sehr grob) der Weg der Daten vom Speicher auf die Disk. Wie Sie sehen, ist das Device eigentlich nur der Laufbursche des Filesystems.

## 1.10 device2

DiskSpare im Besonderen

~~~~~

Das diskspare.device verhält sich (fast) wie jedes andere Device auch, das mit einem Filesystem verwendet werden kann.

Das 'fast' bezieht sich auf die ABSCHALTBARE Patcherei. Die des trackdisk.devices (für mehr Zylinder) und die des FastFileSystems (für die Erkennung des Kapazitätswechsels).

Ansonsten gilt das Gleiche wie bei allen anderen Devicen auch, das diskspare.device weiß nicht was auf den Sektoren steht, die es ans Filesystem weitergibt bzw von ihm erhält.

## 1.11 welche probleme?

### Probleme

~~~~~

Hier will ich einige grundsätzliche Probleme besprechen. Für die Probleme, die mit Fehlern zusammenhängen siehe auch 'Support/Programmers.doc' bei known bugs.

#### A. Das Mountlist Problem

~~~~~

Wenn Sie die Mountlisteinträge verändert haben und Sie dann beim Formatieren nur noch NDOS Disketten erhalten, ist zu 99,9% NICHT das Device schuld, sondern ein falscher Mountlisteintrag. Wenn Sie den FFS-Patch mit Bit 2 ausgeschaltet haben, MÜSSEN die Flags mit den restlichen Einträgen (insbesondere HighCyl = ) zusammenpassen !

#### B. Das Laufwerk Problem

~~~~~

Da diskspare.device einen anderen Trackaufbau hat als das trackdisk.device, bei dem auch einige Bytes mehr geschrieben werden, ist es nicht mit letzter Sicherheit gesagt, daß es mit allen Laufwerken zusammenarbeitet. Was läßt sich schon mit letzter Sicherheit sagen ? Bisher habe ich jedenfalls noch kein Laufwerk gefunden, das nicht funktioniert hat UND es hat sich auch noch niemand bei mir deswegen gemeldet.

#### C. Das FileSystem Problem

~~~~~

Das Problem dabei besteht eigentlich nur darin, daß diskspare die Diskettenkapazität ändern kann (DD <-> HD und verschiedenen HighCyl). Da nämlich einige Festplatten Device in der Beziehung Probleme machen (oder gemacht haben) erlaubt das FileSystem (FFS) dies NUR dem trackdisk.device (neuerdings auch dem carddisk-device). Damit ein Kapazitätswechsel trotzdem erkannt wird, muß das FFS (leider) gepatcht werden, was u. U. bei späteren Versionen (des FFS) Probleme machen könnte.

#### D. Das trackdisk.device Problem

~~~~~

Es tritt nur dann auf, wenn mehr als 80 Zylinder verwendet werden (Default ist 82 !). Es ist nämlich nur mit einem Patch (ein ekliges Wort) möglich mit dem trackdisk.device mehr als 80 Zylinder zu nutzen. Es könnte also möglich sein, daß das mit späteren Versionen Probleme macht.

Das Problem mit zukünftigen Versionen vom FastFileSystem und dem trackdisk.device dürfte sich wohl leider erledigt haben.

#### E. Mehrere Laufwerke auf eine Device Unit mounten

~~~~~

Es ist nicht möglich mit dem FastFileSystem mehr als ein Laufwerk auf eine Device Unit zu mounten. Sie können also nicht DS00: mit 80 Zylindern und DS01: mit 82 Zylindern auf Unit 0 mounten. Das liegt daran, daß das FFS das TD\_REMOVE Commando anstatt TD\_ADDCHANGEINT verwendet.

Mit anderen Filesystemen die TD\_ADDCHANGEINT verwenden (z.B. PFS) gibt es diese Einschränkung nicht.

#### F. Das LoadWB Problem

~~~~~

Wenn ein Laufwerk (egal mit welchem Device) nach dem Booten, besser nach 'LoadWB' gemountet wird, erhalten Sie für unformatierte Disks kein Icon auf der Workbench. Ist bei CrossDOS oder mit Festplatten genauso.

#### G. Das Disketten Problem

~~~~~

Nachdem nicht nur ich in letzter Zeit mit NoName Disketten massiven Ärger hatte, eine Bitte:

Wenn Sie DiskSpare testen und sich eine neue Diskette nicht formatieren läßt, testen Sie dieselbe Diskette bitte auch mit DFX (X=0-3) bevor Sie das Device in die Ecke schmeißen.

Besonders bei HD Disketten ist Vorsicht geboten. Ich hatte mir 40 HD Disketten gekauft und versuchte sie zu formatieren, was mir allerdings mit keinem meiner Laufwerke gelang (weder mit diskspare noch mit trackdisk.device). Bevor ich die Disketten zurücktragen wollte, probierte ich sie noch mal auf meinem PC und siehe da, sie ließen sich ohne Fehler formatieren und auch beschreiben.

Damit will ich NATÜRLICH NICHT sagen, daß der PC die Disketten besser lesen/schreiben kann als der Amiga, aber das es offensichtlich HD Disketten gibt, die so schlecht beschichtet sind, daß sie für den 150RPM 'Trick' nicht zu gebrauchen sind. Siehe Induktionsgesetz.

Allgemein kann man sagen, daß die NoName Disketten mittlerweile so schlecht sind, daß es sich wirklich nicht mehr lohnt, den Mist noch zu kaufen.

Und das sage ich jetzt nicht, um möglichst wenig Ärger mit meinem Device zu bekommen, sondern weil es so ist. Jedenfalls war die Qualität der NoNames früher deutlich besser. Sonst hatte ich fast keine Ausfälle, zuletzt waren es ca. 10-20%. Ich kaufe keine mehr.

#### H. Das DirUtil Problem

~~~~~

Einige Leute haben mir geschrieben bzw gesagt, daß ihr DirUtil den freien Platz auf DiskSpare Disketten falsch anzeigt. Ich habe daraufhin verschiedene Programme probiert und festgestellt, daß tatsächlich eines (kommerzielles!) dabei ist, das den Wert falsch anzeigt. Aber nicht nur auf DiskSpare Disketten sondern generell. DirectoryOpus ist es aber (wenigstens in der Version

mit der ich es probiert habe) nicht.  
Sie können die Anzeige aber leicht selbst kontrollieren, öffnen Sie ein Shell Fenster und geben Sie 'Info DSX:' (x=0-3) ein. Nun sehen Sie in der Ausgabe eine Spalte, da steht: Free und darunter eine Zahl. Diese Zahl multiplizieren Sie mit 512 (oder, wenn Sie einen anderen SectorSize eingestellt haben sollten, mit Ihrem Wert) und erhalten somit den freien Platz auf der Disks. Wenn dieser Wert nicht identisch mit dem Wert ist, den Ihr DirUtil anzeigt, sollten Sie vielleicht darüber nachdenken, ein anderes zu verwenden.

#### I. Das '????' Problem

~~~~~

Mittlerweile haben sich mehrere Leute bei mir 'beschwert', daß immer wenn sie DiskSpare Disketten einlegen, ein Icon erscheint: z.B. 'DF0:????'.

Damit hat das diskspare.device absolut nichts zu tun. Es ist nur die (ganz normale) Anzeige dafür, daß in DF0: eine Diskette liegt, die nicht gelesen werden kann. Bei Verwendung von DiskSpare haben Sie ja zwei (logische) Laufwerke, die auf ein (physikalisches) Laufwerk zugreifen, nämlich DFX: und DSx:. Wenn Sie auch noch mit CrossDOS arbeiten, haben Sie sogar drei Icons für Ihr Laufwerk, nämlich z.B. DF0:, DS0: und PC0:.

Wenn Sie nur stört, daß das ????-Icon genauso aussieht wie das normale Disk-Icon, können Sie das leicht selber ändern:

Starten Sie 'IconEdit' (im Tools Ordner). Stellen Sie den Typ auf 'Disk' und malen Sie eine durchgestrichene Diskette (von mir aus auch einen Totenkopf, einen Blumenstrauß oder sonstwas) und speichern Sie dieses Icon als 'Vorgabepiktogramm' (Default). IconEdit speichert es nun einmal in

```
SYS:Prefs/Env-Archive/Sys/def_disk.info
```

und auch noch in

```
RAM:ENV/Sys/def_disk.info.
```

Benennen Sie diese beiden def\_disk.info in def\_kick.info um, dann erscheint in Zukunft für eine nicht lesbare Diskette Ihr eigenes Icon. Das erleichtert das Finden von 'ansprechbaren' Disketten auf jeden Fall.

**ACHTUNG:** Wenn Sie bereits ein eigenes Default Icon eingestellt haben, müssen Sie dieses VOR der Prozedur irgendwohin sichern und danach wieder zurückschreiben !

Stört es Sie überhaupt, daß da zwei Icons für ein Laufwerk rumliegen, können Sie das MFS von Nicola Salmoria verwenden. Nach der Installation von 'Multi File System' haben Sie nur noch ein Laufwerk DF0:. Es verwaltet dann (das eigentliche) DF0:, DS0:, PC0: und was Sie sonst noch wollen.

#### J. Das Kopierprogramm Problem

---

~~~~~

Das Problem hat nicht DiskSpare mit den Kopierprogrammen, sondern die Kopierprogramme mit DiskSpare. Es gibt meines Wissens noch kein Programm, das DiskSpare Disketten SCHNELL kopiert. Sie müssen also wohl oder übel das normale DiskCopy verwenden. Wenn es Ihnen nicht schnell genug geht, können Sie die Option 'NOVERIFY' (in der Shell) angeben. Das sollten Sie aber nur tun, wenn sie 249,5 %ig sicher sind, daß die Zieldiskette in Ordnung ist ! Ein Beispiel: 'diskcopy DS0: to DS1: noverify' kopiert also die Diskette die in Laufwerk 0 liegt auf die Diskette im Laufwerk 1 OHNE Überprüfung.

Der Nibble Mode einiger Programme kopiert zwar DiskSpare Disks, er dauert aber sicher länger wie 'DiskCopy' MIT Verify.

#### K. Das Format Problem

~~~~~

Das Format Programm, das ab Workbench 2.1 dabei ist, hat neben dem, daß die Anzeige (in der Shell) völlig falsch ist, dem daß beim Formatieren mit der Workbench immer ein Rest angezeigt wird (auch der Blitter kann keine 4.05 Pixel setzen) und dem, daß es bei Rückgabe einer Fehlermeldung bei TD\_FORMAT total ausrastet (laut off. Docs darf TD\_Format aber einen Fehler zurückliefern !) noch einen übelsten Fehler. Zum Glück tritt er aber 'nur' bei schon einmal formatierten Disketten auf. Dabei passiert dann folgendes: Wenn ein Zylinder formatiert wird, wird er ja nochmals überprüft. Wenn dabei allerdings eine Fehlermeldung zurückgegeben wird, wird der Zylinder nicht, wie zu erwarten, nochmals formatiert und überprüft, sondern es wird der nächste Zylinder nur gelesen ?!

-> Wenn der nächste Zylinder in Ordnung ist, wird die Diskette weiterhin formatiert OBWOHL sie einen Defekt hat !

-> Die Datenspeicherung auf solchen Disketten wird zum reinen Glücksspiel.

Dieser Fehler ist deviceUNabhängig ! D.h. er passiert GENAUSO bei trackdisk Disketten. Zum Glück (?) merkt man aber beim fileweisen Schreiben auf die Disk, daß sie einen Fehler hat.

Das (für mich) Schlimmste ist, daß mich dieser Sch...fehler zig Stunden Arbeit gekostet hat, weil ich ihn in meinem Device gesucht habe (ich war kurz davor einen Exorzisten hinzuzuziehen).

Abhilfe: Verwenden Sie den Format Befehl von Workbench 2.0 (sieht zwar nicht so gut aus, aber dafür funktioniert er !!) oder formatieren Sie diskspare Disks auf trackdisk und umgekehrt bevor Sie sie dann nochmal anders formatieren.

## 1.12 tips & tricks

### Tips & Tricks

~~~~~

Diese Rubrik ergänze ich immer wieder mit Ihren Tips & Tricks. Wenn Sie also etwas entdeckt haben, das auch anderen helfen könnte, dann schreiben Sie bitte.

Beim Mounten unter Workbench 2.0 ist es besser in der

User-Startup die 'addbuffers' Zeile zu verwenden, als den Mountlisteintrag mit 'Mount = 1', da Sie dann keinen 'No Disk in Drive XXX' Requester erhalten.

Wenn Sie Ihre alten Datendisketten auf diskspare umformatieren möchten, ist es am besten eine Ramdisk mit den gleichen Werten wie eine diskspare Diskette zu mounten. Beispiele dazu finden Sie im DD bzw HD Ordner. Sie können dann ihre Daten fileweise in diese Ramdisk kopieren. Wenn diese voll ist, kopieren Sie sie mit (z.B.) 'diskcopy RDD: to DS0:' auf Diskette. Das ist die (mit Abstand) schnellste Möglichkeit Ihre Daten umzukopieren. Sie benötigen aber 1 (DD) bzw 2 (HD) Megabyte zusammenhängenden (am Stück) Speicher.

Wenn Sie diskspare Laufwerke gemountet haben, kann es z.B. bei Backups stören, daß das Device auf die Diskette zugreift. Die beste Möglichkeit dies abzustellen ist: Öffnen Sie ein Shellfenster und geben Sie ein: 'diskcopy DSX: to DSX:' (X=0-3). Das klappt auch wenn X bei beiden Laufwerken gleich ist. Also z.B. 'diskcopy DS0: to DS0:'. Jetzt werden Sie von 'DiskCopy' aufgefordert <Return> oder Ctrl-C zu drücken, machen Sie aber weder das eine noch das andere. Diese Aktion hat zur Folge, das Disk-Copy dem FileSystem verbietet auf die Laufwerke zuzugreifen -> Sie können Disketten wechseln, ohne das das diskspare.device darauf zugreift.

Booten von diskspare Disketten ist direkt nicht möglich, es gibt aber einen (indirekten) Trick trotzdem davon zu booten. Mit ihm ist es möglich von jedem Device mit jedem Filesystem zu booten. Erstellen Sie eine (normale, also nicht mit diskspare) Bootdiskette und mounten Sie mit dieser eine RAD: mit ca. 30K. Dann kopieren Sie in diese alle nötigen Files um diskspare zu mounten und lenken das Booten auf diese Disk um. Ziemlich umständlich, werden Sie vielleicht meinen, aber sehr flexibel. Sie können so nämlich auch mit anderen Filesystemen booten, die weder im ROM noch auf einem RigidDiskBlock sind. Sie müssen nur die RAD: entsprechend größer mounten, damit das Filesystem auch hineinpaßt. Außerdem, nachdem auch das diskspare.device nicht im ROM ist, müssen Sie auf jeden Fall EINMAL von irgendwas anderem booten. Nachdem das mit der RAD: nicht so ganz einfach ist, (und das Handbuch sich (nicht nur) in dieser Richtung bedeckt hält) habe ich dafür ein Script geschrieben, um eine solche Bootdiskette mehr oder weniger automatisch zu erstellen. Sie finden es in 'Support' und es heißt 'MakeBootDisk'. Dieser Tip gefällt mir auch deshalb so gut, weil er mir keine zusätzliche Arbeit macht. Um das Device 'richtig' resetfest zu machen, müßte ich erstens das Device umschreiben (ginge ja noch), dann müßte ich einen 'Install' Befehlersatz schreiben, da der originale Install Befehl auf trackdisk- und carddisk.device testet und deshalb unbrauchbar ist. Und dann könnte man immer noch nicht mit einem anderen Filesystem booten.

Hier könnte Ihr Tip stehen.

## 1.13 adresse

Adresse

~~~~~

Klaus Deppisch  
Beckenwengert 3  
97246 Eibelstadt

Tel.: 09303 1579

## 1.14 information

Dies und das

~~~~~

### A. Sicherheit

~~~~~

Das Device wurde seit Dezember 1992 von mir auf allen greifbaren Amigas (500-4000) getestet. Und was viel wichtiger ist, es wird (wenigstens von mir und einigen Bekannten) auch benutzt ! Wir haben alle zusammen ca. 800 Disketten auf DiskSpare Format (um-) kopiert und bisher keine Probleme. Sie sehen also, ich habe ein starkes Interesse daran, daß das diskspare.device mindestens so sicher arbeitet wie das trackdisk.device.

Wenn Sie den neuen Verify Mode einschalten, ist das diskspare-device zwar langsamer (nur beim Schreiben) aber dafür deutlich sicherer als das trackdisk.device.

### B. Geschwindigkeit

~~~~~

Das diskspare.device (version 3.0) ist deutlich schneller als das trackdisk.device (für DF0-3 verantwortlich). Tests mit dem V37 (KS 2.0) FastFileSystem.

disk/mode diskspare trackdisk

DD	w	bis 12300 Bytes/sec	bis 10900 Bytes/sec
DD	r	bis 25000 Bytes/sec	bis 22700 Bytes/sec
HD	w	bis 13500 Bytes/sec	bis 11600 Bytes/sec
HD	r	bis 26400 Bytes/sec	bis 23500 Bytes/sec

Diese Werte wurden mit DiskSpeed 4.2 ermittelt. Der einzige Punkt, bei dem das trackdisk.device die Nase vorn hat, ist die CPU Belastung. Es belastet die CPU (durchschnittlich) 10-15% weniger als das diskspare.device. Je schneller der Rechner, desto geringer der Unterschied.

### C. Viren

~~~~~

Wenn Sie DiskSpare Disketten verwenden, haben Bootblockviren bei Ihnen keine Chance mehr, da selbst wenn sie sich überhaupt

auf die Bootblöcke schreiben können, richten sie (zumindest damit) keinen Schaden an. Von DiskSpare Disketten kann nicht gebootet werden, also wird der Virus damit auch nicht mehr aktiviert.

Auch bei der (indirekten) Bootmethode über die RAD wird der Bootblock NICHT ausgeführt ! Vor Linkviren schützt Sie das aber leider auch nicht.

## 1.15 pfs

Professional File System

~~~~~

Version 9.5 vom 10.10.94 (z.B. AMINET)

In dieser Version befindet sich eine diskspare.device Version 2.2, die von Michiel Pelt OHNE MEIN WISSEN und ENTGEGEN meiner (und auch seiner !) Copy-Bestimmungen modifiziert wurde.

Das ich nach dieser SAUEREI kein sonderliches Interesse mehr an einer Zusammenarbeit mit Michiel Pelt habe, dürfte wohl klar sein !

Desweiteren untersage ich es ihm hiermit AUSDRÜCKLICH ab Nov. 94 mein Device, egal welche Version, zusammen mit PFS zu vertreiben.

PFS Version 9.5 schreibt doppelt so schnell wie ältere Versionen. Da ich nicht will, daß das mit meinem Device zu Datenverlusten führt, MUß bei Verwendung von PFS V9.5 der VerifyWrite Modus eingeschaltet werden ! Die beigefügten Mountfiles sind bereits dahingehend geändert.

Ich rate Ihnen DRINGEND davon ab, ältere Versionen als DSD V3.0 ohne den VerifyWrite Modus mit PFS ab Version 9.4 zu verwenden.

Version 7.5 vom 20.4.94

DSD Version 3.0 läuft mit PFS Version 7.5 auf 68000 NICHT mehr. Wenn Sie meinen, Sie müßten unbedingt PFS verwenden, besorgen Sie sich bitte die letzte Version (9.5).