GTAK 2.58

Contents

About

Author: Andreas Kaiser
Danziger Str. 4
D-70597 Stuttgart
Germany

Internet: Andreas.Kaiser@stuttgart.netsurf.de

Fidonet: 2:246/8506.9

Fax: 49-7531-18708 - see note below.

If you can use electronic mail, please do so. The fax number is not my own, so sending stuff not related to GTAK gets dropped into the trashcan without reaching me.

If you use email just to tell me about bugs and limitations which are already mentioned in the docs, be prepared for an impolite reply. OS/2 VIEW can search for words pretty fast, so try to find comments about the subject in the docs before asking me, especially in the sections *known bugs* and *notes*.

In the case of Internet email, please use a valid domain style "From:" address. If you use bang paths, bitnet, the .uucp domain or some weird kind of embedded domain address, chances are good that you won't see any reply. Chances are only little better if you ask for "please reply to x@y.z" in the body instead of the header, especially if this note is not easily noticed at the top of your mail (if I don't notice it until my reply has been returned, it is almost certainly to late to be fixed).

If appropriate, mention the basic facts about your system (SCSI adapter, tape drive, used device driver, OS/2 version, ...) <u>and the version number of this software</u> in mails. It increases your chances for a useful reply.

Legal notices:

For components marked A.K. in the following chapter:

For components marked SSB:

```
Copyright (c) Stuttgarter Straßenbahnen AG 1993-1997.
```

Die Komponenten diese Paketes unterliegen nicht der GNU General Public Licence. Sie sind vielmehr als Erweiterung des Betriebssystems OS/2 gedacht, um Backup-Software mit SCSI-Streamern lokal wie über Netzwerk benutzen zu können. Der Quellcode ist nicht erhältlich. Sie können diese in kommerzieller Umgebung nutzen, Sie dürfen sie jedoch nicht verkaufen oder auf andere Art Geld oder andere Gegenleistungen dafür erhalten. Sie dürfen diese Komponenten nicht ohne ausdrückliche Einwilligung des Autors als Basis eines kommerziellen Produktes einsetzen, auch nicht als Beigabe.

Die Komponenten für Netzwerkunterstützung enthalten Code, der Eigentum der Stuttgarter Straßenbahnen AG ist. Sie wurden freundlicherweise für eine Nutzung entsprechend den o.A. Einschränkungen freigegeben.

English translation of the above sentences (the German version has precendence):

The components of this package are not governed by the GNU General Public Licence. They are intended as extension to the OS/2 operating system, to allow backup software access to SCSI tape devices, local or remote via network. The source code is not available. You can use them within a commercial environment, but you may not sell them or make profit in other ways. You may not use it as a base for a commercial product or as an add-on to a product, except when explicitly permitted by the author.

The components for network support contain code owned by the Stuttgarter Straßenbahnen AG and were made available for use according to the restrictions mentioned above.

Files

Executables

File	SCSI	NetBIOS	TCP/IP	Description	Copyright
tape.exe	x	client	client	Common tape control utility.	A.K.
netbserv.exe		server		NetBIOS server.	SSB and A.K.
tcpserv.exe			server	TCP/IP server.	SSB and A.K.
netbagnt.exe		[client]		Network agent for NetBIOS.	SSB and A.K.
tcpagnt.exe			[client]	Network agent for TCP/IP.	SSB and A.K.

Support Files

File	SCSI	NetBIOS	TCP/IP	Description	Copyright
buffer.exe	x	client	client	Stream buffer.	A.K.
scsitape.dmd	[x]	[server]	[server]	SCSI tape driver.	A.K.
aspitape.sys	[x]	[server]	[server]	SCSI tape driver.	A.K.
aspitape.old	[x]	[server]	[server]	SCSI tape driver of GTAK 2.45.	A.K.
scsitape.sys	[x]	[server]	[server]	SCSI tape driver.	A.K.
scsitape.exe	x			Tape control utility for SCSI.	A.K.
netbtape.exe		client		Tape control utility for NetBIOS.	A.K. and SSB
tcptape.exe			client	Tape control utility for TCP/IP.	A.K. and SSB
netwtape.exe		[client]	[client]	Tape control utility for network agent.	A.K. and SSB
rmt_scsi.dll	x			TAR interface for SCSI.	A.K.
rmt_disk.dll				TAR interface for floppy.	A.K. and K.U.R.
rmt_netb.dll		client		TAR interface for NetBIOS.	A.K. and SSB
rmt_tcp.dll			client	TAR interface for TCP/IP.	A.K. and SSB
rmt_netw.dll		[client]	[client]	TAR interface for network agent.	A.K. and SSB
aspitape.exe				DOS version of the tape control utility, based on ASPI.	A.K.

Documentation

File	Description	Copyright
GTAK.INF	This file in OS/2 hypertext format.	A.K.
GTAK.PS	This file in Postscript format.	A.K.

Prerequisites

Device Drivers

ASPITAPE.SYS (A)

SCSI adapter with ADD driver.

Installed OS2ASPI.DMD.

SCSITAPE.SYS (S)

SCSI adapter with ADD driver.

Installed OS2SCSI.DMD.

Usage is discouraged.

SCSITAPE.DMD (D)

SCSI adapter with ADD driver.

This driver is no longer maintained. Usage is discouraged.

SCSI Adapters

If you run a SCSI adapter not listed below, you may try all 3 versions but please try the ASPI based driver first (**A**). Be prepared for immediate or delayed system crashes. Note that a SCSI driver fault can damage filesystems, so backup your filesystems before. If you are successful, please send a note to one of my email addresses (and be prepared to find your name in this table next time).

Adapter	D	A	S	Note	Reported by, Comments
Acculogic ISAport	x	?	?		rovero@oc.nps.navy.mil
Adaptec AHA1522	x	x	?		kline@juncol.juniata.edu
Adaptec AHA154xB	x	x	x		Author
Adaptec AHA154xC[F]	x	x	?		Brian Macy - bmacy@tuba.calpoly.edu
Adaptec AHA1640	x	?	?		Christian Laubscher - 2:301/111@fidonet
Adaptec AHA174x[A]	x	x	?		Brian Macy - bmacy@tuba.calpoly.edu
Adaptec AHA274xT	-	x	?		Kai Uwe Rommel - rommel@ars.muc.de
Adaptec AHA2842	-	x	?		Mike Isely - isely@fnal.fnal.gov
Adaptec AHA2940[A][UW]	-	x	x		Kai Uwe Rommel - rommel@ars.muc.de
AMI "Fast Disk" EISA	?	x	?		<mprager@semi3.sefc.miami.edu>
Buslogic 445S	x	?	?		arishem@ac.dal.ca
Buslogic 747S EISA	x	?	?		71043,45 CompuServe
Buslogic KT930LC	x	-	?		Frank Blatzheim - 2:2453/30.74@fidonet
Buslogic 946C PCI	x	?	-		Matt Austern - matt@physics.Berkeley.edu
COMPAQ Proliant 2000 Integrated FAST SCSI-2	-	x	-		Russ Herman <rwh@gov.on.ca> Requires CPQ53CX.ADD from COMPAQ OS/2 support disk version 2.04B. instead of CPQC710.ADD.
Data Technology DTC3250, DTC3280	x	?	?		Jan P. Finegan - jfinegan@crl.com
DPT 2011	x	?	?		Mark Leidecker - 2:241/5111.3@fidonet
DPT PM2122 EISA	x	?	?		Hemmat Ragheb - infotech@ritsec1.com.eg
Future Domain TMC-885	-	-	x		Rob Prikanowski - rpr@oce.nl
Future Domain TMC-1640 MCA	-	x	?		Andre Kajnik - 2:2448/7835.13@fidonet
Future Domain TMC-1650/70	-	x	?		martin@matchb.demon.co.uk
Future Domain TM;C-1660	-	x	-		Jacco de Leeuw - leeuw@fwi.uva.nl
Future Domain TMC-1680	-	x	-		Nigel J. Clarke - nclarke@bda-hp.bda.nasa.gov
Galaxy DC 880	?	?	x		Heinz Repp 100434,2106 Compuserve Using the AHA154x driver.
IBM MCA 32-bit cached SCSI	x	x	x	1	Author, Kai Uwe Rommel - rommel@ars.muc.de
IBM RAID Adapter	x	?	?		Tim Rand <tim@datapage.com>
IBM SCSI-2 Fast/Wide	-	x	-		Andre Kajnik - 2:2448/7835.13@fidonet

NCR 53c8xx chip (PCI)	x	x	x		Author, Kai Uwe Rommel - rommel@ars.muc.de V3 driver (OS2CAM). For V4 see Symbios.
PAS-16 (Soundcard)	?	?	?		Gerd Hardt - 2:2455/240.1@fidonet
QLogic Fast!SCSI VLB	x	x	x		Jan P. Finegan - jfinegan@crl.com
QLogic Fast!SCSI PCI	x	x	x		Markus Zangerle - maza0000@stud.uni-sb.de
Seagate ST01	x	?	x		You need the device driver written by Jochen Friedrich <jofried@fzi.de>.
Symbios 53c8xx chip (PCI)	-	x	-		Author V4 driver (SYM8XX). For V3 see NCR
Trantor T130B	x	?	?		Dave Chapman - dchapman@clunix.cl.msu.edu
Trantor T348 printer port adapter	?	x	?		Author
	x	?	?		100275,364 CompuServe Get T348SCSI.ADD, size 24509, date 4.5.1993, time 14.28.
Ultrastor 24F	?	x	?		tony@mpg.phys.hawaii.edu
Ultrastor 34F	-	x	?		phaniraj@badlands.NoDak.edu
WD7000	-	-	-		ray_stricklin@tscnet.eskimo.com **Does not work.**

Note 1: IBM microchannel SCSI adapters may timeout on long lasting commands. I have not yet found out how to reliably disable this timeout.

Tape Devices

If you are successful and your tape device is not listed below, or if you can update or correct some info in this table, please send a note to one of my email addresses (and be prepared to find your name in this table next time).

The **QFA** column describes absolute QFA only. Relative file access is supported by all tape devices, although it is not necessarily quick. If **Mode** is S2:

QFA = x DB supported, LB support unknown - results of responses to GTAK version 2.12 or earlier.

QFA = DB DB supported, LB not supported.

QFA = LB LB supported, DB not supported or not recommended.

Device	SCSI Level	Mode	QFA	Note	Reported by, Comments
Archive Viper 2150S	SCSI-1	S0	x		Clemens.Beckstein @informatik.uni-erlangen.de
Archive Viper 2525S	SCSI-1	S0	x		Steve Fairfax - fairfax@mit.edu
Archive DAT					See Conner DAT
Cipher ST150S1/S2	SCSI-1	S0	-		ray_stricklin@tscnet.eskimo.com ST150S2 shows ANSI level 1, so use S0 even though it shows SCSI-2 response format.
Conner 2GB DAT	SCSI-2	S2	LB	7,8	henckens@alpha.luc.ac.be
Conner DAT 4320NT	SCSI-2	S2	LB	8	Timothy F. Sipples - tsipple@vnet.ibm.com Use a fixed blocksize (512). The default does not seem to work.
Conner DAT 4584NP DDS1 Media Changer	SCSI-2	S2		8,9	Mike Bilow mikebw@bilow.bilow.uu.ids.net
Conner DAT 4586NP DDS2 Media Changer	SCSI-2	S2		8,9	Mike Bilow mikebw@bilow.bilow.uu.ids.net
Conner CTT8000-S	SCSI-2	S2	LB	7	Johannes Springst 2:2432/500.8@fidonet
DEC TLZ04 (DAT)	SCSI-2	S2	?		Guy van Baalen <guy_van_baalen@sydpcug.org.au>
DEC TZK10 525Mb	SCSI-2	S2	x		harry@stack.urc.tue.nl A Tandberg 3820 with modified firmware.
DEC TZ 887 DLT MediaChanger	SCSI-2	S2	LB	9	Kai Uwe Rommel <rommel@ars.de>
EXABYTE 2501	SCSI-2	S2	LB		tkal@vnet.ibm.com Explicitly requesting relative QFA increases performance when archiving. LB required for partition control.
EXABYTE 4200c	?	?	?		cjensen@netcom.com
EXABYTE 8200	SCSI-1	S0	-	5	Kenneth Crudup <kenny@panix.com>
EXABYTE 8505					See IBM 7206-011

HP DAT DDS1 2-8GB, HP DAT DDS2 4-16GB	SCSI-2	S2	LB	6	Author QFA logical and device block ids differ, so tape directories created in LB and DB modes are not compatible.
HP DAT DDS2 Media Changer (C1553A)	SCSI-2	S2	LB	9	Kai Uwe Rommel <rommel@ars.de>
HP DAT DDS3 (1554C aka C1537A)	SCSI-2	S2	LB		Carsten Ellermann 2:2432/215@fidonet
IBM 7206-005 (DAT 4GB)	SCSI-2	S2	LB	6	Author Identifies itself as "ARCHIVE IBM".
IBM 7208-011 (8mm 5GB)	SCSI-2	S2	LB	5,6	Author Identifies itself as EXABYTE 8505
Maynard 2.2GB	?	?	?		Steve Houle <wookie@panix.com>
Sankyo CP150	SCSI-1	S0	x	1	manne@minsk.docs.uu.se
	SCSI-2	S2	x		frank@sax.sax.de Unclear about SCSI version. Maybe there is both a SCSI-1 and a SCSI-2 version. Maybe there is a switch. Or some info is wrong.
Sankyo 525MB	?	?	?		Mark Leidecker - 2:241/5111.3@fidonet Equivalent to Wangtek 5525ES.
Seagate TST8000	SCSI-2	S2	LB	7	Johannes Springst 2:2432/500.8@fidonet Identifies itself as Conner CTT8000.
Sony SDT-2010, SDT-4000	SCSI-2	S2	LB		Thomas Schuster - 2:2480/3505.6.
Sony SDT-7000	SCSI-2	S2	LB	6	Author
Tandberg TDC3660 150MB	SCSI-1	S1	x	2	Author Original Tandberg firmware.
	SCSI-1	S0	-	2,4	Author IBM RS/6000 firmware.
	SCSI-2	S2	x	2	
Tandberg TDC3820 525MB	SCSI-1	S1	x	2	
	SCSI-2	S2	DB	2	Manfred Huber - 100021,3416 CompuServe It implements an early draft of the SCSI-2 standard and a few commands have changed since, so some SCSI-2 features don't work even though the functionality is available.
Tandberg TDC4120 1GB	SCSI-1	S1	x	2,7	The device may be upgraded to SCSI-2 firmware via SCSI upload.
	SCSI-2	S2	DB,LB	2,6,7	Author LB mode is not supported by early firmware revisions. Tape directories created in LB and DB modes are not compatible. It supports one additional fixed-size partition, using the last track (of 30) for the optional second partition.

Tandberg TDC4220 2GB	SCSI-2	S2	x	2	Eberhard Mattes - mattes@azu.informatik.uni-stuttgart.de
TEAC MT-2ST 60MB	SCSI-1	S0	?		me90plb@brunel.ac.uk
Wangtek 5150ES	SCSI-1	S0	(x)		QFA support depends on the firmware revision. New versions support QFA, early ones don't.
Wangtek 5525ES	SCSI-1	S0	x	1	s541575@sun15.tfh-berlin.de
	SCSI-2	S0	x	1	Michael Buenter - 2:301/602@fidonet
Wangtek W51000ESX	SCSI-1	S0	x	3	Roland_Mausolf@wildcat.fido.de
	SCSI-2	S2!	x	3	Richard Reuters - 2:2440/151.11@fidonet Shipped as SCSI-1 device without QFA capability but has been set to SCSI-2 by Wangtek program.
Wangtek DAT 6130HS, 6200HS	SCSI-2	S2	LB	3	Dave Chapman - dchapman@clunix.cl.msu.edu
WangDAT 3100	SCSI-2	S2	LB		Christian Laubscher - 2:301/111@fidonet
WangDAT 3200	?	S2	x		Uwe Janssen - 2:242/33.19@fido-classic

Note 1: Device reports SCSI-2 compliance in the INQUIRY data but QFA support requires SCSI-1 commands. One reason for this behavior could be GTAKs use of "device blocks" instead of "logical blocks" in SCSI-2 QFA up to release 2.12. 2.40 adds logical block QFA mode, see TAPE option `LB`. So if you have such a device, you may try again using SCSI-2 mode with `LB` specified. Please tell me if it works.

Note 2: Tandberg tape drives usually differ in the model name and the string returned by `tape inq`. For example, if you buy a TDC3660, `tape inq` nevertheless shows "3600". Some Tandberg devices are available both in a SCSI-1 and in a SCSI-2 version. Use `tape inq` (ANSI x) to see which SCSI-level your device implements.

Note 3: Some Wangtek drives seem to support both SCSI-1 and SCSI-2. Depending on the revision of the controller board, there may be a jumper to select the default SCSI level and/or it can be switched to SCSI-2 using a DOS program available from Wangtek BBS.

Note 4: There seem to be a lot of refurbished TDC3660 drives around, which were originally used for IBM RS/6000 machines. These drives have a different firmware and do not support QFA. Use `S0` for such a drive, **do not use S1**. If you don't know whether your device has such a firmware, try `tape tell` (QFA). If it shows `invalid command` or alike, it has. To get QFA, you have to replace the drive's firmware; for firmware upgrades ask Tandberg [(805) 579-1000].

Note 5: I received a report about problems with the auto-append option + in the tape specification. Do not use this option. In case of 2.3GB media it is not recommended anyway, since running to end of tape is awefully slow.

Note 6: Partition control is supported by GTAK. Other devices may or may not work, since partition control is one of the weak areas of the SCSI-2 standard definition.

Note 7: The device may run at different physical data transfer rates (or at least it seems so), selectable by `tape speed` *n*

Note 8: Identifies itself as "ARCHIVE Python".

Note 9: The media changer functionality is supported. The device allows transparent addressing and supports at least the sequential mode (see below).

Network

NetBIOS Support

NetBIOS interface DLL ACSNETB.DLL

Tested with IBM LAN Server 2.0 .. 4.0, Token-Ring and Ethernet networks.

TCP/IP Support

IBM TCP/IP 2.0 .. 3.0.

Tested with Token-Ring network.

Installation

Copy the required executables (*.EXE) into a directory mentioned in the environment variable PATH.

Copy the required DLLs (*.DLL) into a directory mentioned in the environment variable LIBPATH.

If you do not intend to access tape devices via network, you can remove all network related executables and DLLs and you can rename `scsitape.exe` to `tape.exe`. The distributed `tape.exe` just looks at the environment variable TAPE and calls the appropriate `*tape.exe`.

Device Driver

Select one of the three alternate tape device drivers, based on the SCSI adapter compatbility table shown above. Please use ASPITAPE.SYS whenever possible.

ASPITAPE.SYS

Be sure CONFIG.SYS contains the line

    BASEDEV=OS2ASPI.DMD

ASPITAPE.SYS can be placed anywhere on your disks, as long as you specify its full path in CONFIG.SYS. Add the following line to CONFIG.SYS:

    DEVICE=your_path\ASPITAPE.SYS

or

    DEVICE=your_path\ASPITAPE.SYS device-name Aano id Smode options

All command line parameters are optional, but the first one is always interpreted as device name (please note that this paragraph is valid for ASPITAPE.SYS only, the other drivers behave differently). The device name defaults to TAPE$4 and the SCSI level defaults to S0 (sense key). If the adapter number and/or the SCSI target id are not specified, the driver scans the adapters and/or SCSI devices for the first suitable device. Since the SCSI level is not used within the device driver itself, but only propagated to the application layer where the SCSI level can also be specified in the TAPE environment variable, simple installations may not require any command line argument (my tribute to plug and pray ;-).

Unless the option `I` is specified, the driver looks for the specified device on startup and complains if it is not present.

You may access devices which are permanently locked by other device drivers if you add `/ALL` to the command line of OS2ASPI.DMD. If the device is locked by a device driver based on ASPI, add `/SHARE` instead. **Attention:** `/ALL` and `/SHARE` bypass reservation locks for all devices accessed through ASPI (not limited to tape devices), leading to data corruption if the same device is used in different sessions at the same time. If neither `/ALL` nor `/SHARE` are specified, a device accessed though ASPI is permanently locked on first access. Other drivers will not be able to access the device afterwards.

There were several changes in ASPITAPE.SYS after 2.45. If you get into trouble you may try the old one, called ASPITAPE.OLD (which does require command line arguments) - in such a case, **please tell me**.

SCSITAPE.SYS

This driver may soon fall out of maintenance. Usage is discouraged.

Be sure CONFIG.SYS contains the line

```
        BASEDEV=OS2SCSI.DMD
```

SCSITAPE.SYS can be placed anywhere on your disks, as long as you specify its full path in CONFIG.SYS. Add the following line to CONFIG.SYS:

`DEVICE=`*`your_path`*`\SCSITAPE.SYS `*`device-name unit`*` S`*`mode`*

The driver installs even if there is no tape device present.

Differing from the other drivers, it expects a logical device number, not a SCSI id.

The device is locked by the driver as long as it is used by a program. You cannot access a device which is permanently locked by another driver (unless you remove the other driver and reboot, of course).

SCSITAPE.DMD

This driver is included for compatibility with earlier releases. It is no longer maintained. Usage is discouraged.

Copy SCSITAPE.DMD into the root or \OS2\BOOT directory (preferably) of your OS/2 boot partition (the only places where a BASEDEV driver is accepted - if you are running OS/2 2.11 or less, put it into \OS2 instead). Add the following line to CONFIG.SYS:

`BASEDEV=SCSITAPE.DMD `*`device-name`*` A`*`ano id`*` S`*`mode`*

Do not add `/!SM:`*`id`* to the SCSI driver's command line as this option disables the tape driver too. If the device is locked by OS2SCSI.DMD (e.g. CDROM), comment out OS2SCSI.DMD.

The driver does not install (fails quietly) unless the tape drive is available when the system boots. Tape access yields an error message like *file not found* if the device was not available on startup.

The device is locked by the driver as long as it is used by a program. You cannot access a device which is permanently locked by another driver (unless you remove the other driver and reboot, of course).

Device Driver Arguments

mode = 2 — For SCSI-2 devices.

mode = 1 — For Tandberg SCSI-1 devices only (the SCSI-1 versions of TDC3660, TDC3820, TDC4120).

mode = 0 — For other SCSI-1 devices.

Run `tape inq` and look for `ANSI 1` or `ANSI 2` to know what SCSI level your device supports. Use a mode of 0 for this purpose. There are some tape devices which claim to conform to SCSI-2 but need the Tandberg SCSI-1 commands for QFA, at least in version 2.12. Use a mode of 0 if the device supports neither logical block nor device dependent block QFA (`LB`, `DB` in TAPE).

In addition to QFA, the driver's mode argument determines the way, how both the driver and the tape interface interpret and print SCSI error codes. The info shown in `S1` and `S2` is much more detailed, but if your device doesn't support it, the info shown invalid - that's why you should not use `S1` with a TDC3660 having the RS/6000 firmware or with other SCSI-1 devices.

Defaults to 0 (sense key) if not specified).

ano Adapter number, starting with 0. Only required if you have multiple SCSI adapters in the system. Adapters are numbered sequentially in the order their drivers appear in CONFIG.SYS.

id SCSI ID.

Preferred SCSI IDs for a tape device are 4,5,6, unless you have a IBM PS/2 machine or a Trantor adapter which may order targets in reverse. Do not use 0 or 1 unless you run out of IDs, as some BIOSes reserve 0 and 1 for disks. Do not use ID 7, as this ID usually is used by the SCSI adapter itself.

Required for SCSITAPE.DMD, optional for ASPITAPE.SYS.

unit 1 for the first sequential device (default), 2 for the second and so on. Multiple SCSI adapters are handled automatically. If they use different device drivers, the order in which the drivers are mentioned in CONFIG.SYS is important.

SCSITAPE.SYS only, optional

Example:

```
set TAPE=SCSI:+LB,TAPE$4
BASEDEV=SCSITAPE.DMD TAPE$4 4 S2
```

or

```
set TAPE=SCSI:+LB,S2,TAPE$4
BASEDEV=C:\SYS\ASPITAPE.SYS
```

If you like to sort your CONFIG.SYS: The environment variable can be specified anywhere, even outside of CONFIG.SYS. The programs TAPE and TAR use it to determine the default tape device name.

If you have more than one tape device, install the driver once for each device, using different device names.

Hint: You may be tempted to use `TAPE$0` for the first and `TAPE$1` for a possible second tape device, but this way you are unable to specify the TAR tape device in short form as `-4` or `-5`, since the options `-0` and `-1` are reserved for floppy access.

Add `C`*x*, where *x* is the device type code, to the driver command line if you want to use the driver and the TAPE utility to access non-tape devices. This way you can run any arbitrary SCSI command on the device, unless it is locked by another driver. Don't use `C`*x* unless you definitely know what you are doing, as it can cause any kind of trouble, including system crashes, low-level disk reformatting and permanent device failure (that's why I don't publish the type codes, you should already know the numbers if you know how to use direct SCSI commands).

At installation time, SCSITAPE.DMD and SCSITAPE.SYS do not show errors when the tape device is not found. SCSITAPE.DMD checks device presence on startup and quietly fails when it is not found, so the driver is not loaded when the device is not available and the device name is not present. ASPITAPE.SYS checks device presence on startup and complains if the device is not found or of the wrong type (can be suppressed by the `I` switch). SCSITAPE.SYS does not check device presence on startup, instead it presents an appropriate error code when opened.

Network

Server

Install the appropriate drive driver as described above and make sure the all required EXE and DLL files are available. You may want to start the tape server automatically in STARTUP.CMD or the startup folder. See below.

Client

Do not install a device driver unless you have a tape device attached locally.

Emergency Boot Disk

If you have archived your OS/2 boot partition including extended attributes and all hidden and system files, you can restore the partition to disk running OS/2 from floppy disk (or from a separate partition, check the minimal system setup option of BOOT2X).

You may want to create a boot diskette with SCSI tape support, to be able to restore your files in case of a damaged hard disk. Create the *utility diskettes* using the program shipped with Warp or make a single boot diskette using the package BOOT2X, which is available on many FTP servers, BBSs and OS/2 shareware CDROMs.

Copy the device driver and its prerequisites (see above) to the second utility diskette and make sure the physical device driver of your SCSI adapter is present on the diskette and there is an entry for it in CONFIG.SYS, like

```
BASEDEV=OS2CAM.ADD
```

for the NCR PCI adapter family. This will probably not be the default, at least not in case of an NCR adapter.

Add the tape driver configuration statements as shown above. Copy the TZ environment setting, else the timestamps may be incorrect.

Caution: Both the non-CDROM installation disks and the utility diskettes will probably not allow any SCSI device access except for the first two BIOS disks, since there may be no SCSI driver installed. When booting non-IBM machines from unmodified diskettes, OS/2 may run the INT13 driver to access the disk drives instead of the proper SCSI driver. The INT13 driver does not allow SCSI device access except for the two BIOS mapped harddisks.

Copy `tar.exe`, `scsitape.exe` and `rmt_scsi.dll` to the third utility diskette. If you are using compressed backups, add `gzip.exe` (either use the shipped version which does not require additional runtime DLLs or be sure to copy those DLLs too).

To restore an archived OS/2 boot partition to a freshly formatted disk, you may have to use the program `sysinstx.com`. In my experience, FAT partitions may go without, but HPFS partitions occasionally need `sysinstx c:`. Note that you cannot change the filesystem type just by archiving and restoring the OS/2 partition, at least the OS/2 bootstrap program (OS2BOOT) is different between FAT and HPFS.

You may be able to restore via network using CID installation diskettes (untested). Of course you need the respective versions of the tape utility and `rmt_*.dll`.

Test it before you actually need it!

Environment

Tape Device Name Specification - TAPE

The default device is given by the environment variable TAPE. For readability, the following device name descriptions may contain blanks. However blanks are not allowed in the actual device specification.

Both TAR and TAPE accept `-0`...`-9` as a shorthand notation for the device name specification. It is mapped to the contents of the environment variables TAPE$0..TAPE$9, if defined, or to `DISK:A:`..`DISK:B:` (`-0`..`-1`) or `SCSI:LB,`TAPE$2 .. `SCSI:LB,`TAPE$9 (`-2`..`-9`) otherwise.

General syntax:

```
        interface : + device-specification
```

The + character is optional and signals TAR to automatically seek to end of tape before writing to tape. If it is omitted, TAR tries to write to tape wherever the tape head is positioned.

Both TAR and TAPE accept device specifications with the syntax as show here. If TAPE is `scsi:+tape$4`, then

```
        tar cv .
```

is exactly equivalent to

```
        tar cvf scsi:+tape$4

        tar cv0
```

is equivalent to

```
        tar cvf disk:a:

        tar cv2
```

is equivalent to

```
        tar cvf scsi:+tape$2
```

Locally Attached SCSI Tape Drive

Syntax:

```
        SCSI: + [options,]device-name
```

options	Optional. Multiple options are separated by comma.
S*mode*	Overrides the driver's S*mode*.
L*unit*	Specify the SCSI logical unit. Defaults to 0. Only allowed with ASPI.
`DB`	Use device dependent block ids for SCSI-2 QFA (default, compatible to GTAK 2.12).
`LB`	Use logical block ids for SCSI-2 QFA (preferred).

`FF`	Replace the "seek to end of tape" operation (`tape end`) by filemark skips. Use this option if the operation is not supported (EXABYTE 2.3GB) or buggy (EXABYTE 5GB).

Example:

```
SCSI:+S2,LB,TAPE$4
```

Floppy Disk

Syntax:

`DISK:` *`x`*`:`

X: is the (floppy) disk device name. It need not be a floppy disk however, you can easily kill a partition by specifying a drive letter other than `a:` or `b:`.

Remote Tape Device via NetBIOS

Syntax:

`NETB: +` *`adapter`*`:` *`NetBIOS-name`*

NetBIOS-name is the network-wide unique name of the tape server, at most 16 characters long and case-sensitive. If the *adapter* number, including the colon, is omitted, it defaults to 0.

Example:

```
NETB:+1:DAT194
```

Remote Tape Device via TCP/IP

Syntax:

`TCP: +` *`port@hostname`*

Port is the port number or tcp service name, *hostname* is an Internet address or host name of the server machine.

Remote Tape Device via Network Agent

Syntax:

`NETW: +` *`agent-name`*

Agent-name is the network name specified in the command line of the network agent.

Network

Network Server

The server provides network-wide tape access. There is no built-in security, whoever knows the name can use the device, as long as no other client is using the device at the same time. The server process is terminated by Ctrl-C. A client is automatically disconnected when no request is received within 5 minutes.

You can have multiple server processes running at the same time on the same machine, using the same or different tape devices. You can access the tape on the server locally without stopping the server. But note that there are no interlocks, so if you have both a TCP/IP and a NetBIOS server for the same device, things can get messed up quite easily.

NetBIOS Server

`NETBSERV` *`Adapter:NetBIOS-name device-name`*

Where *device-name* is the SCSI device specification including options, without the leading `SCSI:+` string. If the *adapter* number, including the colon, is omitted, it defaults to 0.

Example:

```
netbserv 1:dat194 db,tape$4
```

TCP/IP Server

`TCPSERV` *`port device-name`*

Where *port* is the port name or number described in the previous section.

Example:

```
tcpserv 2500 tape$4
```

The network protocol is not compatible to Unix remote tape sharing (rmt).

Network Agent

A network agent is an optional program, run on the client, which established a connection to the server and leaves the connection open until terminated or timed out. If the network agent is not used, each invocation of TAPE or TAR established a network connection, runs the command(s) and closes the connection. The agent is terminated by Ctrl-C.

A network agent is typically used interactively with the NetBIOS interface, since establishing a NetBIOS connection is quite slow - it can take 5-10 seconds just to reach the server.

While a connection is open, no other client can use the device via the same server name, so a network agent can also be used to lock the device.

The argument of the agent specifies both the network destination and the agent name as used by the tape device specification.

You can have multiple agents running at the same time, using different agent names.

NetBIOS Agent

```
start NETBAGNT NetBIOS-name
set TAPE=NETW:+NetBIOS-name
```

TCP/IP Agent

```
start TCPAGNT port@hostname
set TAPE=NETW:+port@hostname
```

Client Hints

The network transmission is synchronous, the client waits until the server acknowledges the successful execution of the request. Performance usually increases with the size of a TAR block, for example by `-b120` (60KB). The NetBIOS implementation as well as the device drivers SCSITAPE.* limit the the blocksize to 127 records (65024 bytes).

As with locally attached tape drives, using the buffered mode of TAR (`--buffered`) may increase the overall performance, since network transmissions run in parallel with file operations, but you loose QFA and multi-volume support.

Example

Server

Shown configuration: PS/2 machine, NetBIOS network, HP DAT as first tape device.

CONFIG.SYS:

```
device=c:\sysos2\scsitape.sys tape$4 1 s2
```

STARTUP.CMD:

```
start netbserv dat194 lb,tape$4
```

Client without Agent

CONFIG.SYS:

```
set tape=netb:+dat194
```

Client with Agent

CONFIG.SYS:

```
set tape=netw:+dat194
```

Use:

```
start netbagnt dat194
```

Backup/Restore Program - TAR

If you archive to tape and the device name contains a + sign in the proper place, TAR always appends to all existing data on the tape, it skips existing tape data before new data is written to the tape. It does not erase existing tape contents. If there already was some data on the tape before (even if cpio, Sytos, ...), this is what you get when reading the tape from the beginning. If this data does not have the same archive format (compressed archive or another backup program), you get an error message like "doesn't look like a tar archive".

If you want to rewrite the tape from the beginning, erase the tape (`tape erase`, see below, which is slow but recommended since it also retensions the tape in the case of a QIC tape device), write an empty filemark at the beginning of the tape (`tape rewind mark`) or remove the plus sign from the device name (so that TAR tries to write to tape wherever the tape is positioned at the moment).

If you write an empty filemark to erase the tape, you have to skip the first tape file (`tape file`) when reading the tape, else you get an error message.

Using the Tape Utility

Command line:

`tape [options] subcommand(s)`

Run it without any arguments to get usage information. You can specify multiple verbs in a single call, unless the `-Nowait` option is used. Options and subcommands can be abbreviated. Most options and subcommands are not case-sensitive, except for some dangerous commands which have to be specified in upper case.

Options

`-0` .. `-9`	Shorthand notation for the device name specification. See above. DOS ASPI version: SCSI-ID (default=4).
`-lun` *n*	Select SCSI LUN (ASPI only).
`-device` *name*	Device name.
`-nowait`	Do not wait until command completes. This is only a hint, many commands ignore this option.
`-ignore`	Do not terminate on error.
`-detailed`	Detailed display of mode and log pages.
`-stdout`	Display messages on standard output instead of standard error output.
`-block` *size*	Logical block size in bytes, for use with data transfer subcommands in variable length blockize mode (blocksize = 0). A suffix of `K` or `M` specifies kilobytes or megabytes. This option can also be used to increase the transfer buffer size beyond the builtin default of 30K.
`-limit` *size*	Stop data transfer subcommand after *size* bytes have been transferred. A suffix of `K` or `M` specifies kilobytes or megabytes.
`-inq` *name*	Use *name* as manufacturer name when decoding vendor specific info. Use this switch if the device pretends to be of a different brand as it actually is (such as a HP DAT pretending to be a Tandberg QIC). Currently the names ARCHIVE, EXABYTE, HP, SONY and TANDBERG are recognized.

Subcommands

Cartridge Control

`load` — Load media. Most devices are preset to autoload mode, so tapes are usually loaded and rewound automatically when new media is inserted.

`retension` — Retension media.

`unload` — Rewind and unload media. While QIC type devices usually cannot physically eject the media, they nevertheless distinguish between loaded and unloaded state, sometimes indicated by a LED.

`unloadend` — Same as `unload`, but position to the other end of the media instead. The net effect is a full rewind when the media is inserted next time, which retensions the tape. Useful when shipping QIC tapes.

`rewind` — Rewind.

`reserve` — Reserve the device (currently this is a SCSI reservation only, no network reservation implied).

`release` — Release a reserved device.

`wait` *[n]* — Wait until the device is ready, at most *n* seconds.

Media Changer

`move` *n1 n2 [n3]* — Move a medium from the source element *n1* to the destination element *n2* using the transport element *n3* (optional, defaults to 0). A tape changer element may be a slot or an embedded drive. Consult the subcommand `elements` or the device command reference for the interpretation of the element numbers.

`elements` — Show information about the type and number of tape changer elements, such as slots, medium transport elements (robots), data transfer elements (embedded drives) and import/export elements. Also reports which changer elements currently contain media and the source slot of the currently loaded media.

The information about the SCSI IDs and LUNs through which the host can access these subdevices is only meaningful in case of media changers with multiple embedded drives and/or robots.

`load` *n* — *HP media changer only:* Loads the medium from the designated slot into the embedded drive. Note that in this case, the slot number must be from the range 1-6 and differs from the tape changer element address (which would be 2-7).

`unload` *n* — *HP media changer only:* Unloads the medium from the embedded drive back to the source slot. Note that in this case, the slot number must be from the range 1-6 and differs from the tape changer element address (which would be 2-7).

`unloadmagazine`	*HP media changer only:* Unload the current medium and eject the media changer's magazine.

Information

`status`	Test drive ready.
`inquiry`	Display inquiry data.
`mode` *[n]*	Display mode page. The default is all available mode pages.
`log` *[n]*	Display SCSI-2 log pages. The default is all available log pages.
`clear` *[n]*	Clear SCSI-2 log pages. The default is all clearable log pages.
`version`	Display version numbers of tape and SCSI control. In case of remote tape devices, the first line shows the version of the client part, the second shows the version of the remote tape server.

Head Movement

`file` *[n]*	Skip beyond *n*-th filemark relative to the current position. Default = 1. Negative *n* skips backwards, if supported by the device.
`block` *[n]*	Skip beyond *n*-th tape block relative to the current position. Default = 1. Negative *n* skips backwards, if supported by the device.
`tell`	Show block id of current tape position. This subcommand is part of QFA support.
`seek` *n*	Position to a block id previously obtained by `tell`. Block ids are magic numbers, they are not easily related to the amount of data transferred. This subcommand is part of QFA support.
`end`	Position to logical end of tape, the beginning of the unused part of the tape beyond all written data.
`select` *n*	Select partition *n* (0-based). SCSI-2, optional.

Read & Write

`mark` *n*	Write *n* filemarks. Default = 1.
`setmark` *n*	Write *n* setmarks. Defautl = 1. SCSI-2, optional.
`erase`	Erase tape completely. Takes a few minutes on QIC devices and up to several hours on DAT or Video8 devices.

`erasequick` Mark the tape as erased. Whether completely or just from from current position to EOT may depend on the device. Useful for helical scan devices. Not implemented on the QIC devices I've seen.

`verify` *n* Verify readability of *n* tape files. Default = 1.

`read` *filename* | -
Read tape data to file or pipe. For additional parameters, see options `-limit` and `-block`.

`write` *filename* | -
Write file or pipe to tape. For additional parameters, see options `-limit` and `-block`.

Mode Control

`blocksize` *n* Set the device blocksize in bytes. A value of 0 enables variable blocksize mode.

`density` *n* Set the media density code. This is a device type dependant code. For values, consult the device reference.

`speed` *n* Set the speed code. Completety device dependant.

`compression` on|`off`
Set compression mode. Tries mode page 0x0F first, then 0x10 (see below).

SCSI-2, optional.

`partition` *control*
Control partitions.

control = `reset`
Remove additional partitions.

control = `fixed`
The number and the size of the partitions are defined by the device.

control = `select=`*number*
Create *number* additional partitions, the sizes are defined by the device.

control = `size=`*n1*`,`*n2*`,`...
Create partitions with specified size in megabytes (uncompressed). When two or more numbers are given, it is assumed that the first number specifies the primary partition (ignored unless the device strictly supports the SCSI-2 medium partition page definition). If only one number is given, it is the size of the addtional (second) partition. A non-numeric text instead of a number means all remaining space of the tape.

control = *number*
Tries `size=`*number*, `select=1` and `fixed`, in this order.

The partition control implementation includes a number of hacks required for tape devices which interpret the SCSI-2 standard in a way different to the intention of the standard (most of them do). So do not expect that everything works fine, just because there is no error message.

GTAK includes hacks for ARCHIVE 4GB DAT (IBM 7206-005), EXABYTE 8505 (IBM 7208-011) the HP DAT family and TANDBERG TDC4120 SCSI-2. SONY SDT-7000 is the only known device which really conforms to SCSI-2 partition control..

SCSI-2, optional.

`set` *control* Set mode page data. See below.

`define scsi-1|ccs|scsi-2[,save]`

Set the SCSI command set definition. Changes are permanent if `save` is specified.

SCSI-2, optional.

Media Changers

Device Selection

In the SCSI device model, media changer devices are logically separated from the tape devices which do the actual data transfer. So the media changer device is addressed either using a different logical unit number (if it is the same physical SCSI device) or a different SCSI ID (robots). However some devices, such as HPs media changers, simplify use by accepting both tape and media changer commands on the same logical unit even though the media changer appears on the SCSI bus as a second logical unit.

So in practice there are three possible scenarios:

Transparent: The media changer is logically integrated into the tape device, e.g. HPs media changers. In this case you do not have to care about the beforementioned difference between the tape device and the media changer device.

Separate unit: The media changer device is the same SCSI device as the tape device, but must be logically distinguished by the SCSI LUN. In this case, you have to use the ASPI based tape device driver and you may use the option `-lun` to distinguish between the different logical units. You may have to add the switch `/ET` to the SCSI adapter driver's command line.in CONFIG.SYS to enable LUN support.

Separate device: The media changer (typically a robot) is a different SCSI device having a different SCSI id, perhaps hooked to a different SCSI bus. In this case you have to install the tape device driver multiple times, once for each tape device and media changer device, with different device names each and you have to add the device type code specifier `C8` to the argument list of the tape device driver(s) associated with a media changer.

Media Selection

The media changer may support either or both of the following media selection modes.

Sequential mode: The `tape unload` command not only unloads the currently loaded medium and returns it to its original slot, it also loads the next available medium.

Random mode: The `tape unload` command unthreads the currently loaded medium but does not unload and return it to its original slot nor does it load another medium. The user or application software has to use the `tape move` command for this purpose and must itself keep track of the media usage.

The sequential mode is especially convenient to be used with GNU `tar` and multivolume support. For this purpose, the first medium to be written to with `tar` should be loaded with `tape load` *`slot`* (HP only, see below) or `tape move` (other devices). The `tar` program should be invoked with option `-M` aka `--multi-volume` and volume labels should be used (`-V` aka `--label`). In addition, the option `-F` aka `--info-script` with argument `"tape unload"` may be used to suppress `tar`'s prompt for a new medium at the end of the current medium and to instead cause the next available medium to be loaded automatically. The `tar` program will then continue writing on the next available medium without user intervention. If no new medium is available, `tar` terminates with a warning message.

Alternatively, instead of `"tape unload"`, any other command could be used as argument to `--info-script` that has the same effect (calls `tape unload` internally) but can do more sophisticated error handling (e.g. to send an e-mail message or to alert the operator if no new medium is available).

`tar`'s quick file access (QFA) does not know about media changer devices thus does not automatically select the required medium.

Additional notes for the HP media changer devices

This applies to Hewlett Packard DAT media changers only.

When a HP vendor specific `tape load` *`slot`* command is issued to load a medium from a magazine slot into the embedded drive or a medium is loaded using the load button, the media changer switches to sequential (stacker) mode. When a regular `tape move` or `tape elements` command or a HP vendor specific `tape unload` *`slot`* command is issued, the media changer switches to random mode.

Mode Control

If your device supports mode pages, you can use

```
tape set page options
```

to change individual bits or bytes within mode pages.

Options is a comma-separated list of assignments or options:

`save` Make all mode changes permanent, including the current device blocksize.

`byte``=`*`value`* Set byte at offset *byte* to *value*.

`byte``.`*`bit`*`=`*`value`*
Set bit *bit* at offset *byte* to *value*.

`byte``:`*`length`*`=`*`value`*
Set *length* bits starting at offset *byte* to value.

`byte``.`*`bit`*`:`*`length`*`=`*`value`*
Set *length* bits starting at byte offset *byte*, leftmost bit *bit* , to *value*.

All numbers are specified as in the C programming language:

`12`	decimal 12
`0x12`	hex 12, decimal 18
`012`	octal 12, decimal 10.

Use

```
tape mode x
```

where *x* is 0x40+*page* to find out whether a bit or byte within a mode page can be modified. Only bits shown as 1 are changable.

Attention

Changing unknown mode pages or unknown mode bits can result in permanent device failure. Consult the SCSI command reference of your tape device to find out about implemented mode pages and mode bits.

Just because the tape utility knows about some bits and pages does not necessarily imply that your device interprets them in the same way. Several pages or bits decoded by the tape utility are specific to some particular devices, such as Tandberg or EXABYTE, without notification.

Examples

Unless otherwise noted, all examples shown are based on the SCSI-2 and EXABYTE 8500 mode page specifications. Most examples are taken right out of the respective specifications and have <u>not</u> been tested.

`tape set 0x0F 2.7=1 / 2.7=0`
Enable / disable compression as implemented by HP DAT 2GB and EXABYTE 85xx (first choice of subcommand "compression"). Decompression is controlled by `3.7`.

`tape set 0x10 14=1 / 14=0`
: Enable / disable compression as defined by SCSI-2 draft 10L (second choice of subcommand "compression").

`tape set 0x10 8.4=1`
: Enable automatic velocity control (device selects speed to optimize streaming activity).

`tape set 0x10 6:16=50`
: Flush buffered data to tape after 5 seconds of inactivity (0 = disable automatic flush).

`tape set 1 2.2=1 / 2.2=0`
: Report / do not report recovered errors.

`tape set 1 2.0=1 / 2.0=0`
: Disable or enable error correction.

`tape set 1 3=24,8=16`
: Set the read retry count to 24 and the write retry count to 16.

`tape set 0x81 save`
: Reset mode page 1 to factory settings (0x80 + *page*) and make all mode pages and blocksize settings permanent.

`tape set 0xD0`
: Reset mode page 0x10 to saved settings (0xC0 + *page*).

For other mode pages and other control bits, consult the SCSI command reference of your tape device.

Exit Codes

0	Success.
1	Option, subcommand or argument syntax error.
2	System error, such as invalid file specified or file read/write error.
3	Unexpected device behavior.
4	Tape device, SCSI adapter or device driver error, except for those described below.
14	Filemark encountered. Sense key 0 only, not at end of media.
15	At end of media. Sense key 0 only. If supported, this is an early warning indicator, not the physical end of the tape, so there may still be some space left.
16 .. 31	Sense keys 0 .. 15.
16	No error (may indicate incorrect blocksize).
17	Recovered data. Whether a device reports recovered errors or not depends on the current mode page settings.
18	Device not ready.
19	Unrecoverable media error.
20	Hardware failure.
21	Unknown SCSI command or illegal parameter.
22	The media has been changed or the device has been reset.
23	The media is write protected.
24	The media is empty or the the head has reached the end of the used part of the tape.
25	Vendor unique.
27	Aborted command or SCSI bus parity error.
29	Media overflow.

Utilities

Pipe Buffer

Command line options:

`-s` *`number`* Buffer size.

`-b` *`number`* Reblock to *number* bytes per output block. Pad output with 0 if necessary.

`-i` *`number`* Refill buffer when *number* % empty.

`-o` *`number`* Flush buffer when *number* % filled.

Default for `-i` and `-o` is 100.

BUFFER is multi-threaded. If the buffer fill ratio permits according to the input and output thresholds, it will be active on both ends at the same time.

If the buffer size is not specified on the command line, the default size can be specified with the environment variable BUFFER in KB. If not specified, the buffer size is 2MB.

When copying a large file from one partition to another, BUFFER can considerably reduce the effects of disk positioning times by using

```
buffer input-file output-file
```

Blocks and Sizes

Most tape devices with a capacity of at least 525MB can be used with different blocksizes. Both with fixed length blocks of 512 and 1024 bytes as well as with variable length blocks. The default setting varies, so you should set the blocksize before running TAR, using `tape blocksize 512` or `tape blocksize 1024` for fixed size blocks or `tape blocksize 0` for variable length blocks.

You can get the current setting of the tape device with tape mode 0. But note that some devices do not show a valid value until some data has been read. Additionally, this value may depend on the type of the inserted tape. If you got a tape written in variable blocksize mode and don't know the used size, try

```
tape -limit 1 blocksize 0 read nul rewind
```

it should show the actual size of the first block read.

Tapes for QIC-24 (DC600A 60MB), QIC-120 (DC600XTD 120MB) and QIC-150 (DC6150 150MB, DC6250 250MB) must be used with a blocksize of 512 bytes. If DC6150/6250 tapes work and DC6320 or higher do not work, the tape device is probably set to a blocksize of 1024 bytes and switches down to 512 bytes when it recognizes a DC6150/6250 tape (Wangtec 525MB).

TAR works with block sizes of 512 and 1024 bytes (it should work with any reasonable fixed block size if it is a multiple of 512) as well as with variable block sizes. Tapes with variable length blocks of 64KB or larger can only be read or written when using ASPITAPE.SYS (tape driver implementation restriction, ASPITAPE.SYS is currently limited to 1MB).

Don't confuse the backup program's blocksize with the tape blocksize. Unless variable length tape blocks are used, the blocksize used by the backup program can be any value which is an exact multiple of the tape blocksize (TAR's default: 10KB). If variable length blocks are used, a tape block is the same as a transfer block of the backup program.

`tape blocksize 1024`

Set device blocksize to 1024 bytes.

`tar -b 80`

Set TAR blocksize to 40KB.

If you cannot read tapes written on Unix systems, you may have to experiment with the blocksize settings. QIC devices usually are used with fixed blocks of 512 or 1024 bytes. There is absolutely no rule for DAT and Video8 devices however. Operating systems often modify the default blocksize and two machines using the same operating system can have different defaults.

Appendix

Known Bugs

The device driver r/w interface (file access to TAPE$n, no longer available with ASPITAPE.SYS) will not always correctly sense the current device block size, so run `tape blocksize` *n* before, to synchronize driver and device. The driver r/w interface should not be used anyway, use `tape read -` and `tape write -` to/from pipe instead.

-

When a tape is inserted, the very first TAPE command fails with "Unit Attention" or alike. This is not a bug but a feature of SCSI tape drives. TAR itself ignores it using a dummy command but TAPE is intended as a direct device interface, so it does not attempt to hide informations.

-

The extended capacity support provided by the floppy-disk filter driver XDFLOPPY.FLT of OS/2 Warp 3.0 conflicts with the floppy-disk support of GTAK. Maybe it is unable to handle 1.44MB floppies which do not have a valid boot block. To use TAR floppies with Warp, remove the filter driver from CONFIG.SYS.

-

To be able to read blocks larger than 30K wih `tape read`, you have to increase the transfer buffer size using the `-block` option.

-

The device drivers SCSITAPE.DMD and SCSITAPE.SYS and the NetBIOS support limit blocks to somewhat less than 64KB - larger blocks cannot be read successfully. The device driver ASPITAPE.SYS and the TCP/IP support do not suffer from this limitation.

-

If the filesystems behave wierd after installation of SCSITAPE.DMD or SCSITAPE.SYS, you probably forgot to specify the device name in the driver's command line. The default device name of these drivers (????????) is a nice way to fool OS/2.

-

When a tape file exceeds 2GB, `tape verify` shows the wrong byte count.

Notes

I've got questions of the kind "I'm desperately seeking the command to format the tape.". Well, neither QIC (DC6xxx) nor DAT nor Video8 tapes have to be formatted before use. Only floppy tapes have to be formatted because of a limitation of the used floppy controller.

-

Usage info of TAPE is output on file handle 2 (error handle). To redirect the text to a file or printer, add `2>file` or `2>prn` to the command line. To redirect the text into an output pipe, add `2>&1` (redirects handle 2 into handle 1 - I was a little surprised that this is supported by CMD).

-

Please DO NOT USE a SCSI ID of 0 or 1, except in the case of an IBM SCSI adapter. Some BIOSes reserve them for disk devices.

-

From <rwyble@tlc.alcm.org>, who could install either tape or CDROM but not both: "The NEC CDR-38 simply does not coexist well with any other devices in the same SCSI chain. NEC's technical support finally verified this."

History

2.13 .. 2.40

The source code of the tape support is no longer available. But you can add your own tape support (QIC-02, floppy-tapes) now if you like, you just have to create your own `rmt_*` DLL.

The syntax of the device specification (TAPE environment variable) has changed.

Added additional SCSI-2 support to the tape control utility, such as partition control and log page display.

The tape utility can now be used to read and write data in variable blocksize mode (see `-block`) with blocksizes up to nearly 65KB (limit imposed by device driver).

SCSITAPE.DMD supports useful adapter numbers now. In previous versions, it was usable only with multiple adapters of the same kind.

Added EXABYTE bugfix option FF.

Accept "Recovered Data" status codes on read and write.

Added version numbers to tape control and SCSI control (client & server).

2.41 .. 2.43

Made subcommand "set" usable.

Added subcommand "compression".

Added the DOS ASPI version of the tape control utility to the distribution. Perhaps it makes a few GTAK 1.0 for DOS users happy and it can be used to show and set mode pages of devices otherwise locked by OS/2 (such as the write cache of a disk). SCSI-2 error codes are not decoded in this version.

Fixed a bug in SCSITAPE.SYS.

Enhanced error report for floppies. Fixed multi-volume floppy backup.

Added data transfer byte count limit to the tape utility.

The tape utility now returns useful exit codes.

This document now describes most tape utility options and subcommands.

2.44

When `tape write` had to pad the last block, part of the used data was overwritten.

While the "real" tape utilities returned a useful exit status, the stub program tape.exe always returned 0 (actually a bug of the IBM library).

Bugfix "speed".

Sony 4000 DAT does not accept 4-byte mode sense (violating SCSI standards). Avoid it.

2.46

The tape utility stub program tape.exe did not check `-device` for the device type, instead it crashed when TAPE was undefined.

2.47

The size of the pipe buffer's working set now depends on the amount of buffer space actually used (noticable on buffered restores). Previously the working set was the whole buffer.

Fixed some quirks in mode page and log page display.

Allow adapter number specification for NetBIOS connection.

2.48

ASPITAPE.SYS now accepts transfer requests up to 1MB, so the TAR blocksize can be increased to 1MB if desired (unless the NetBIOS support is used). The other two drivers, SCSITAPE.DMD and SCSITAPE.SYS, are still limited to 65535 bytes however. Note that you cannot use GTAK versions below 2.48 with the new ASPITAPE.SYS, since it is no longer compatible to the previous device driver interface. Likewise it no longer supports the direct device r/w interface.

2.49

EXABYTE does not accept mode pages without block descriptor. Now mode and blocksize settings work again. The 8505 supports an additional sized partition. Remove this partition with `size=0`, since `reset` does not work for this device.

Added log page display of HP DAT vendor unique pages. Nice info shown there, such as the left-over space on the tape and the compression ratio.

Reduced the default output of mode and log page display to make it more readable. Use the `-detailed` switch for the complete output.

Log pages can be cleared with `clear`.

Added `-inq` for devices which return fake inquiry info.

ASPITAPE.SYS checks the device type code on startup. Add the `I` switch to avoid error messages caused by devices which are not available.

ASPITAPE.SYS may share devices with other drivers when `/ALL` or `/SHARE` is added to OS2ASPI.DMD's command line (documentation update).

2.50

Some fixes required for GTAR verify/append/update.

2.51

Fixed severe bug in backup with QFA.

2.52

Described how to create emergency boot disks based on Warp's utility diskettes (documentation update).

ASPITAPE.SYS may be installed without command line arguments.

Fixed a few mode pages, revised and added some others.

Revised medium partition page (1) decode, according to SCSI-2/TIB-2 (and SCSI-3).

Made partition control less device dependant. It should now be possible to add or remove a second partition without knowing whether the device requires fixed or sizable partitions (if partitions are supported at all).

The implementation of partition control is nevertheless device dependant to some degree, since most devices do not conform to the SCSI-2 standard for the medium partition page.

The log page decoder knows about Archive 4GB DAT and Sony SDT-7000.

Starting with 2.48, ASPI driver error handling was incorrect.

ASPITAPE.SYS no longer locks the transfer buffer below the 16MB line unless required by the adapter. Note that this capability depends on a feature of IBM's ASPI implementation which was added in some release of OS2ASPI.DMD - if it is not supported by the used release of OS2ASPI.DMD, the transfer buffer is locked below 16MB. Adaptec's ASPI specification update defines this capability bit for the SCSI bus width instead.

Both the updated ASPI definition and IBM's implementation support residual byte count reporting if the adapter driver permits. NCR/Symbios SCSI drivers however pretend to support it, but do not return a valid value, so residual byte count reporting is disabled when ASPITAPE.SYS finds a SCSI driver with NCR SDMS (version 3) or SYM538XX (version 4) signature.

Due to the major changes in ASPITYPE.SYS after 2.45, the old version is included as ASPITAPE.OLD.

The tape utility stopped reading/writing beyond 2GB.

2.55

Fixed a few places where 32-bit integer arithmetic could overflow.

2.56

Thanks to Kai Uwe Rommel, there is some media changer support included, tested with devices from HP (DAT) and DEC (DLT).

Added SCSI LUN support.

2.58

ASPITAPE.SYS could not be installed if a device named TAPE$4 was already present, even though a different device name was specified (e.g. in case of multiple tape drives).

New translation scheme for device names and device numbers.

Index

T

U

V

W

X