Appendix C

Manual Pages

This appendix contains manual pages for the following commands:

itsdd(1)

itsdump(1)

itsibmtape(1)

itsrestore(1)

itstar(1)

Copyright © 1994 Thinking Machines Corporation

itsdd

Copies an input file to an output file, converting data as specified.

SYNTAX

itsdd {**-fromsda** | **-tosda**} [**ifbs=***n*] [**obfs=***n*] [**-a**] [**if=***name*] [**of=***name*] [**obs=***n*] [**bs=***n*] [**cbs=***n*] [**skip=***n*] [**files=***n*] [**seek=***n*] [**count=***n*] [**conv=***value*]

ARGUMENTS

-fromsda | **-tosda**
: The file's transfer direction must be specified by either **-tosda** or **-fromsda**. If **-tosda** is specified, it is assumed that the input file is an IBM 3480−compatible tape unit and the output file resides on the SDA. If **-fromsda** is specified, the reverse is true.

ifbs=*n*
: Uses the input coming from an IBM 3480−compatible tape drive, with a fixed-block size of *n* bytes, where $0 < n < 64K$. This option is used to put the tape drive in fixed-block mode for improved performance.

ofbs=*n*
: Sends output to an IBM 3480−compatible tape drive, with a fixed-block size of *n* bytes, where $0 < n < 64K$. This option is used to put the tape drive in fixed-block mode for improved performance.

-a
: Appends the input to the output file (rather than rewriting it if it already exists).

if=*name*
: Input filename. This option is required when **-fromsda** is specified.

of=*name*
: Output filename. This option is required when **-tosda** is specified.

ibs=*n*
: Input block size in bytes; default = 65,536 bytes. Some devices do not support block sizes greater than 65,535 bytes. See **bs**.

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

obs=*n* Output block size in bytes; default = 65,536 bytes. Some devices do not support block sizes greater than 65,535 bytes. See **bs**.

bs=*n* Sets both input and output block size to *n* bytes, superseding **ibs** and **obs**. Also, if **bs** is specified, the copy is more efficient since no blocking conversion is necessary.

cbs=*n* Conversion buffer size in bytes. Use this option only if **ascii**, **unblock**, **ebcdic**, **ibm**, or **block** conversion is specified. For **ascii** and **unblock**, *n* characters are placed into the buffer, any specified character mapping is done, trailing blanks are trimmed, and a newline is added before sending the line to the output. For **ebcdic**, **ibm**, or **block**, characters are read into the conversion buffer and blanks are added to make an output record of size *n* bytes.

skip=*n* Skips *n* input records before starting to copy.

files=*n* Copies *n* input files before terminating.

seek=*n* Seeks *n* records from beginning of output file before copying.

count=*n* Copies only *n* input records.

conv=*value* Performs specified conversion. *value* is a comma-separated list of any of the following:

- **ascii** Converts EBCDIC to ASCII.
- **ibm** Slightly different map of ASCII to EBCDIC (see **Restrictions**).
- **block** Converts variable-length records to fixed-length.
- **unblock** Converts fixed-length records to variable-length.
- **lcase** Maps alphabetics to lowercase.
- **ucase** Maps alphabetics to uppercase.
- **swap** Swaps every pair of bytes.
- **noerror** Does not stop processing on an error.
- **sync** Pads every input record to **ibs=***n* value.

Copyright © 1994 Thinking Machines Corporation

WHERE EXECUTED

ITS tape controller.

DESCRIPTION

NOTE: `cmdd`'s former IBM data set capabilities have been moved to a companion utility, **`itsibmtape`**, and further enhanced.

The command **`itsdd`** copies a specified input file to a specified output file with any requested conversions. The input and output block size may be specified to take advantage of raw physical I/O. After completion, **`itsdd`** reports the number of whole and partial input and output blocks.

Where size (*n*) is given for an option, the number may end with **`k`** for kilobytes (1024 bytes), **`b`** for blocks (512 bytes), or **`w`** for words (2 bytes). Also, two numbers can be separated by the character **`x`** to indicate a product.

RESTRICTIONS

`itsdd` does not support files that span more than one tape volume.

EXAMPLES

The following example shows how to read an EBCDIC tape (**`/dev/rst0`**) into the ASCII file **`x`** in an SDA file system. The tape is blocked in ten 80-byte EBCDIC card images per record. The resulting ASCII file has all lowercase characters.

```
% itsdd -tosda if=/dev/rst0 of=x ibs=800 cbs=80 \
conv=ascii,lcase
```

The following example writes a file (**`final.data`**) on the SDA (**`sda1`**) to a tape (**`/dev/rst0`**). 64 Kbytes are transferred at a time and no conversions are performed.

```
% itsdd -fromsda if=/final.data of=/dev/rst0
```

Note the use of raw magnetic tape. The **`itsdd`** command is especially well suited to I/O on the raw physical devices because it allows reading and writing in arbitrary record sizes.

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

SEE ALSO

itsdump **itsibmtape**
itsrestore **itstar**

Copyright © 1994 Thinking Machines Corporation

itsdump

Archives SDA-based files or directories.

SYNTAX

itsdump [**-b** *number*] [**-f** *filename*] [**-n**] [**-s**] [**-v**] [**-V** *vol_name*]

ARGUMENTS

-b *number*	Sets the archive's blocking factor. For the ITS, the default is 3200.
-f *filename*	Specifies the archive file. For the ITS, the default is **/dev/rst0**.
-n	Disables Emacs-style editing and history.
-s	Disables output pagination.
-v	Enables verbose mode. By default, verbose is disabled.
-V *vol_name*	Specifies a volume name for the archive. By default, no volume name is given. **-V** must be used on **itsrestore** if the archive was created with a volume name.

WHERE EXECUTED

ITS tape controller.

DESCRIPTION

Use **itsdump** to back up SDA-based files to tape. **itsdump** is an interface, consisting of the commands listed below, that allows users to browse the current filesystem, mark the files to be backed up, and control some of the arguments passed to the **itstar** that **itsdump** forks. Each file is archived with its relative

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

pathname only; that is, the leading *hostname* **:/** portion of the path is stripped off before archiving.

itsdump supports shell-like globbing (wildcarding), Emacs-style command-line editing and command history (see the GNU Readline Library), and selectable output pagination.

Note that **itsdump** supports multiple archives; you can specify up to four **-f** archive arguments. Specifying multiple archives is useful when using multi-volume mode with tape drives that take time to rewind, such as StorageTek. If the **itsdump** command line is ... **-f /dev/rst0 -f /dev/rst1**, for example, **itsdump** uses **/dev/rst1** while it is waiting for **/dev/rst0** to rewind, unload, and reload.

Commands

The commands supported by **itsdump** are listed below in alphabetical order.

blocking *blocking-factor*
: Sets the archive's blocking factor.

cd *directory*
: Changes the working directory for the **ls**, **find**, and **mark** commands.

display
: Prints a list of the currently marked files.

emacsedit
: Enables/disables Emacs-style editing and history.

execute
: Runs **itstar** with appropriate arguments on the marked files.

exit, **quit**
: Exits **itsdump** or **itsrestore** and returns to the shell.

find *expression*
: **find** recursively descends the directory hierarchy from the current directory, seeking files that match a logical expression written using one of the available operators. It is nearly identical to **cmfind**, with the following exceptions:

  - Unlike **cmfind**, this version does not take any pathname arguments.
  - There are two new operators, **-mark** and **-unmark**. These provide a flexible method to (un)mark files based on any of the characteristics supported by **cmfind**.
  - Do not specify **-print** and **-(un)mark** in the same invocation.

Copyright © 1994 Thinking Machines Corporation

help, **?** Prints a brief synopsis of all available commands.

ls [**-aAcdfFgilqrRstu1**] *files*
Lists the contents of a directory. This command is identical to **cmls**, with the exception of the **-C** (columnar output) flag, which here is the default, and which can be overriden by the **-1** (the numeral 1) flag.

mark [**-r**|**-n**] **files** [[**-r**|**-n**] *files*]
Marks files for backup. The **-r** flag specifies that the filenames following it (until the next **-n** flag) are directories, and all files in the subtree rooted at those directories are to be marked. The **-n** flag (the default) specifies that all files following it (until the next **-r** flag) are not to be marked recursively.

Specifying a directory to be marked non-recursively has no effect. Specifying a regular file to be marked recursively is the same as specifying it non-recursively. In order for a file to be marked, it must reside in either the current directory or one of its subdirectories.

In verbose mode, all files being marked that aren't already marked are listed. See **unmark**.

more Toggles output pagination via the UNIX **more** command. This command has no effect on the output of the forked **itstar**.

prompt *string* Sets the command-line prompt to *string*.

pwd Prints the name of the file system's current working directory.

status Displays **itsdump** status including archive filename, blocking factor, current directory, verbose mode, prompt, Emacs-editing/history mode, and output pagination mode.

unmark [**-r**|**-n**] **files** [[**-r**|**-n**] *files*]
Specifies that previously marked files should be unmarked. The **-r** flag specifies that all files following it (until the next **-n** flag) are directory names, and all files in the subtree rooted at those directories are to be unmarked. The **-n** flag (the default) specifies that all files following it (until the next **-r** flag) are not to be unmarked recursively.

Specifying a directory to be unmarked non-recursively has no effect. Specifying a regular file to be unmarked recursively is the same as specifying it non-recursively. In order for a file to

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

be marked, it must reside in either the current directory or one of its subdirectories.

In verbose mode, all files being unmarked that aren't already unmarked are listed. See **mark**.

verbose Enables/disables verbose mode. In verbose mode, **itsdump** prints the names of the files as they are marked or unmarked.

volname *volume_name*
Specifies the volume name of the archive.

EXAMPLE

```
% CMFS_PATHTYPE unix
% setenv DVHOSTNAME sda1
% setenv DVWD /big_project
% itsdump
Using /dev/rst0 for archive Current directory is /big_project
itsdump% ls
june4    june3    june2  june1    may31    may 30
jim1    sean    test
itsdump% mark -r sean -n may30 may31
itsdump% display /big_project/
/big_project/may30 /big_project/may31
/big_project/sean/
/big_project/sean/abc
/big_project/sean/tree
itsdump% unmark may31
itsdump% display /big_project/
/big_project/may30
/big_project/sean/abc
/big_project/sean/tree
itsdump% volname test
itsdump% execute
itsdump% itstar -c -M -V test -b 3200 -f /dev/rst0 -T - -G
itstar: Removing sda spec from names in the archive.
itstar: Removing leading / from absolute pathnames in the archive.
itstar: Prepare volume #2 (/dev/rst0), then hit return
itsdump% exit
```


Copyright © 1994 Thinking Machines Corporation

RESTRICTIONS

As with **itstar**, you must have read permission on each file and directory that you archive. Archives created by **itsdump** cannot be restored using UNIX **tar**.

SEE ALSO

itsdd **itsibmtape**
itsrestore **itstar**

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

itsibmtape

Imports/exports IBM data sets and handles label reading, writing, and verification.

SYNTAX

itsibmtape {**-fromsda** | **-tosda**} **if=***inputfile* **of=***outputfile* \
label={ **isl** | **ial** | **osl** | **oal** | **osl, wsl** | **osl, wal** } \
vsn=*serial_number_list* **dsn=***dataset_name* [**bs=***buffersize*] \
rformat=*fmt*:*blen*:*rlen* [**stackers=***n*] [**-leds**] [**-compress**] [**-a**]

ARGUMENTS

-fromsda | **-tosda**

The file's transfer direction must be specified by either **-tosda** or **-fromsda**. If **-tosda** is specified, it is assumed that the input file is an IBM 3480−compatible tape unit and the output file resides on the SDA. If **-fromsda** is specified, the reverse is true.

if=*inputfile* Specifies the name of the input file. If a data set is being written (SDA-to-tape), this specifies the name of the SDA source file. If a data set is being read from tape, this specifies the name of the device to read it from (e.g., **/dev/rst0**, **/dev/rst1**, etc). An input file must be specified.

NOTE: When the source is an SDA file, it may be specified as a standard UNIX filename or in the form *hostname*:**/***filename*, where *hostname* is the name of the host processor on which the SDA file system is mounted.

of=*outputfile* Specifies the name of the output file. See **if=**.

label=isl | **ial** | **osl** | **oal** | **osl,wsl** | **oal, wal**

Specifies the type of label processing to apply against the given input or output data set. The first character in these tokens indicates whether the data is being read from (**i**) or written to (**o**) tape. **sl** and **al** indicate that the label is written as either Standard Label (EBCDIC) or ANSI, respectively. Specifying **wsl** or **wal** with the corresponding option indicates that the user wishes to replace the existing label with the given label data, rather than using it just for verification.

Copyright © 1994 Thinking Machines Corporation

isl Standard Label processing on input tape. Data is read from the IBM Standard Label that precedes the input data set and compared against the values given in the command-line **dsn** and **vsn** options. If any of these fields do not match, the operator is alerted. If no **rformat** option is specified, the data format, block size, and record size in the label are used for processing. When the end of volume on each tape is reached, the number of blocks read from the tape is compared with the block count given in the label. If these counts do not match, the operator is alerted. If the tape label indicates that the tape was written in compressed format, but the hardware the tape is loaded on is not capable of reading compressed media, the tape is rejected.

ial ANSI label processing on input tape.

osl Standard Label processing on output tape. Data is read from the IBM Standard Label that the tape was initialized with and compared against the values given in the command-line **dsn** and **vsn** options. If these do not match, the operator is alerted; if the operator chooses to proceed, the tape is written, but the label is not altered.

oal ANSI label processing on output tape.

osl, **wsl** Replace any existing Standard Label with the **dsn** and **vsn** specified on the command line.

oal, **wal** Replace the existing Standard Label with the given ANSI labelled data, rather than using it simply for verification.

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

vsn=*serial_number_list*

Specifies one or more six-digit volume serial numbers (VSNs) in the data set. These numbers are separated by commas without spaces. On input tapes, only the volumes specified are processed. For output tapes, enough VSNs must be specified to accommodate all the data; extra VSNs are ignored.

dsn=*dataset_name*

Specifies the name of the current data set. The name is matched against the label of the input or output tape.

bs=*buffersize* Specifies the size of the buffer to allocate in ITS main memory. The **bs** value must be a constant multiple of the tape's block length (as specified in the tape label or on the command line via the **rformat** argument).

Performance Note: The ITS tape controller board provides 4 Mbytes from its 8 Mbyte I/O buffer for **itsibmtape** transfers; the other 4 Mbytes are used for parity and other buffering. To maximize performance, you should allocate as close to 50% of the available buffer memory as possible. Because **itsibmtape** double-buffers the data it transfers, 50% is the most that can be allocated. This means the maximum possible value for the **bs** option is 2 Mbytes.

rformat=*fmt*:*blen*:*rlen*

Specifies the format of the data on the tape. If this option is omitted for input standard label tapes, the formatting information is extracted from the label (an extra step that slows performance) and used. **rformat** must be provided for output data. *fmt* specifies the format of the records on the tape, and must be one of **f**, **fb**, **v**, **vb**, **vbs**, or **u**. *blen* specifies the length of each physical tape block. *rlen* specifies the length of each logical record on the tape (0 for variable-format tape). The record length must divide evenly into the block length, and the block length must divide evenly into the buffer size (that is, **bs** >= *blen* >= *rlen*, and **bs**%*blen* = 0, and *blen*%*rlen* = 0). The *fmt* field definitions are:

f	Fixed-length records
fb	Fixed-length blocked records
v	Variable-length records

Copyright © 1994 Thinking Machines Corporation

vb	Variable-length blocked records
vbs	Variable-length, blocked, spanned records
u	Undefined

For more information, see the **Description** section.

stackers=*n* Causes up to four input stackers to be used. This allows the operator to overlap rewinding of one tape with the loading and reading or writing of the next tape in the data set. It also allows more than 10 tapes to be loaded concurrently, each stacker on an IBM 3480−compatible tape system typically holding 10 tapes.

NOTE: Multiple stackers must be sequential, beginning with the device specified by the **if=** or **of=** option. For example, the following arguments:

itsibmtape if=/dev/rst0 . . . stackers=2

would cause **itsibmtape** to use **/dev/rst0** and **/dev/rst1**. If the following arguments were used instead:

itsibmtape if=/dev/rst2 . . . stackers=3

itsibmtape would use **/dev/rst2**, **/dev/rst3**, and **/dev/rst4**.

−leds Displays the volume serial number of the tape currently being processed on the status display of the tape drive.

−compress Enables the use of any built-in data compresion hardware in the IBM 3480−compatible tape drives. This option is used only for writing to tape. Compressed tapes are uncompressed automatically on input and require no special processing. If compressed tapes are loaded on a drive that does not have the compression hardware, the operator is alerted and the tapes are rejected. New compressed tapes are labeled as such and cannot be read by tape units that do not have the data compression hardware.

−a Specifies that the data set is to be appended to the destination file.

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

WHERE EXECUTED

ITS tape controller.

DESCRIPTION

itsibmtape allows the import and export of IBM data sets on IBM Standard Label 3480-type tapes via a compatible tape drive connected to an ITS node installed in a CM-5 SDA backplane. The ITS node acts as a gateway between the SCSI cabling used by the tape drives and the CM-5 Data Network (DR). The DR, in turn, serves as the communication path between the ITS node and SDA file systems mounted on the CM-5.

itsibmtape is designed to maximize throughput between an IBM 3480−compatible tape system and the SDA. However, the streaming-tape access method used by 3480-type tape drives should be understood so that optimal use can be made of **itsibmtape**.

Streaming-tape access methods mean the tape system is able to read and write data as fast as the tape passes beneath the tape heads. If the system runs out of data to write, however, the tape drive will not stop motion immediately. This means the tape must be rewound past the point at which I/O stopped and then brought back up to speed again before data transfer can be resumed. If the tape system is "starved" for data when writing a tape, or if the SDA is not able to receive data as fast as the tape system provides it, more time will be spent repositioning the tape.

To avoid this problem, the **itsibmtape** command employs a technique known as double buffering. When the user specifies a buffer size with the **bs** option, two buffers of the specified size are allocated in the ITS buffer memory. When reading from tape, data is read into one of the two buffers. When that buffer has been filled, the tape unit starts depositing data into the second buffer and the first buffer is used to write to the SDA file system. As the tape unit completes transfers, these buffers are repeatedly switched, ensuring that the tape is kept in continuous motion as much as possible. A similar process is employed for writing, with the SDA filling one buffer with data while the tape unit is removing data from the other.

In addition to enhancing the performance of the tape unit, **itsibmtape** also performs the standard tasks of label reading, writing, and verification. Many of the options to **itsibmtape** are used to perform verification, including **label**, **dsn**, **vsn**, and **rformat**.

Copyright © 1994 Thinking Machines Corporation

NOTES on **`rformat=`***fmt*

If the *fmt* field of the **`rformat`** option is **`v`**, **`vb`**, or **`vbs`**, **`itsibmtape`** exhibits the following behavior:

- On input of variable-record format tapes, **`itsibmtape`** creates a control file (named **`output_filename.ctl`**), which contains information about the record and block lengths of the input tape. The block descriptor words (BDW) and record/segment descriptor words (RDW/SDW) are *not* stripped from the data.

- On output of variable-record format tapes, **`itsibmtape`** checks for the existence of a corresponding control file. If the file exists, **`itsibmtape`** uses it to determine the size of each tape block to write. The data must either contain the correct BDW and RDW/SDW entries, or contain placeholders (zeros) at the correct locations, as specified by the control file. In the latter case, **`itsibmtape`** fills in the placeholders with the correct descriptor words extracted from the control file. If no control file is found, **`itsibmtape`** peruses the data itself to determine the size of each tape block to write (the data must contain the correct descriptor words).

- When processing variable-record output tapes, **`itsibmtape`** attempts to use the fixed-block mode if it determines that all the block sizes in its buffer are the same size. In such cases, the buffer size specified on the command line does not need to be a multiple of the fixed-block size. This use of fixed-block mode cannot be overridden by **`itsibmtape`**; that is, **`itsibmtape`** will choose an appropriate block size, ignoring any block size specified on the command line.

  If the *fmt* field of the **`rformat`** option is **`f`** or **`fb`**, **`itsibmtape`** automatically uses the fixed-block mode to improve performance. To take full advantage of this, specify a large buffer size (usually on the order of several megabytes) to **`itsibmtape`**. The buffer size must be a multiple of the value the tape unit uses for fixed-block mode, which is determined as follows:

  - For **`isl`** processing, **`itsibmtape`** automatically sets the fixed-block mode to the size specified in the block length field specified in the HDR2 label.

  - For **`osl`** or **`onl`** processing, **`itsibmtape`** sets the fixed-block mode to the block length specified in the **`rformat`** argument.

  To prevent **`itsibmtape`** from automatically using fixed-block mode as described above, specify a fixed-block mode size of **`-1`** on the command line (using the **`ifbs`** or **`ofbs`** argument).

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

PERFORMANCE CONSIDERATIONS

In general, IBM 3480−compatible tape systems can process the various fixed-record formats (**f**, **fb**) much faster than the variable formats (**v**, **vb**, **vbs**, and **u**). Because CM-5 file systems do not support structured files, variable-format files require additional processing as they are read from tape. For instance, the data streams of variable-format files contain start-record and end-record indicators, which most application programs want to remove or skip. To facilitate this, **itsibmtape** creates an index into the data set as it is read. This index is called a *control file* and is named **dataset_name.ctl**.

Creating the control file can be quite time-consuming since the entire input buffer must be scanned before it is written to disk. Although **itsibmtape**'s double buffering provides extra time for the program to scan the buffer, extremely short variable-length records can still cause a considerable performance degradation. Therefore, if a variable-length data format is being used to store primarily fixed-length data, it would be advantageous to switch to a fixed or unblocked format if at all possible.

A second cause of performance degradation is frequent reading of short blocks — shorter, that is, than the block size specified by **itsibmtape**'s **rformat** option. Short blocks degrade performance by causing the tape unit to pause as it reports the short block size to **itsibmtape**. **itsibmtape** must then take the drive out of fixed-block mode, rewind the tape one record, reread the short record, and put the drive back into fixed-block mode before it can continue with the read. (Short blocks on fixed-record tapes may occur because of a variable number of records stored per block. For example, if the **rformat** option specifies **fb:8000:80** (record length of 80 bytes, block size of 8000 bytes), each tape block should contain 100 records. Short blocks would result if only 99 records were put on some of the blocks.)

RESTRICTION

Although **itsibmtape** supports reading multiple data sets from a single volume, it does not support writing multiple data sets to a single volume.

Copyright © 1994 Thinking Machines Corporation

EXAMPLES

To read a four-volume, IBM Standard Label data set with VBS format and maximum block size of 32760 bytes:

```
% itsibmtape if=/dev/rst0 of=sda1a:dataset -tosda label=isl \
dsn=TMC.IBM.DATASET vsn=811302,808911,823984,822922 \
bs=3276000
```

To write an IBM Standard Label data set spanning no more than four volumes, with fixed-size records of 80 bytes and tape block size of 8000 bytes:

```
% itsibmtape if=sda1a:data of=/dev/rst0 -fromsda label=osl \
dsn=TMC.IBM.FIXEDDATA vsn=812391,822661,824907,808961 \
rformat=f:8000:80 bs=800000
```

SEE ALSO

itsdd **itsdump**
itsrestore **itstar**

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

itsrestore

Restores (extracts) files and directories from a tape archive.

SYNTAX

itsrestore [**-b** *number*] [**-f** *filename*] [**-n**] [**-s**] [**-v**] \
[**-V** *vol_name*]

ARGUMENTS

-b *number*	Sets the archive's blocking factor. The default blocking factor of an ITS node is 3200. **-b** is required for **itsrestore** if the archive is not blocked in the default record size.
-f *filename*	Specifies the archive file. The ITS default is **/dev/rst0**.
-n	Disables Emacs-style editing and history.
-s	Disables output pagination.
-v	Enables verbose mode. By default, verbose mode is disabled.
-V *vol_name*	Specifies a volume name for the archive. By default, no volume name is given. **-V** must be used on **itsrestore** if the archive was created with a volume name.

WHERE EXECUTED

ITS tape controller.

DESCRIPTION

Use **itsrestore** to extract files from the tape archive and place them into the current file system. When executed, **itsrestore** builds a tree representation of the archive's directory information and provides an interface, composed of the commands listed below, that lets you browse through the tree, mark the files to be

Copyright © 1994 Thinking Machines Corporation

extracted, and control some of the arguments passed to the **itstar** that **itsrestore** forks. Files are extracted relative to the directory that is current when **itsrestore** is invoked. **itsrestore** supports shell-like globbing (wildcarding), Emacs-style command-line editing and command history (see the GNU Readline Library), and selectable output pagination.

Note that **itsrestore** supports multiple archives: you can specify up to four **-f** archive arguments. Specifying multiple archives is useful when using multi-volume mode with tape drives that take time to rewind, such as StorageTek. If the **itsrestore** command line is ... **-f /dev/rst0 -f /dev/rst1**, for example, **itsrestore** uses **/dev/rst1** while it is waiting for **/dev/rst0** to rewind, unload, and reload.

Commands

The commands supported by **itsrestore** are listed below in alphabetical order.

blocking *blocking-factor*
: Sets the archive's blocking factor.

cd *directory*
: Changes the working directory within the archive's tree for the **ls** and **mark** commands.

display
: Prints a list of the currently marked files.

emacsedit
: Enables/disables Emacs-style editing and history.

execute
: Runs **itstar** with the appropriate arguments on the marked files.

exit, **quit**
: Exits **itsdump** or **itsrestore** and return to the shell prompt.

help, **?**
: Prints a brief synopsis of all available commands.

ls
: Lists the contents of a directory. This is a simplistic version of **cmls** that lists names only, does not provide columnated output, and takes no arguments other than one or more filenames.

mark [**-r**|**-n**] **files** [[**-r**|**-n**] *files*]
: Marks files for restoration. The **-r** flag specifies that the file names following it (until the next **-n** flag) are directories, and all files in the subtree rooted at those directories are to be marked. The **-n** flag (the default) specifies that all files

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

following it (until the next **-r** flag) are not to be marked recursively.

Specifying a directory to be marked non-recursively has no effect. Specifying a regular file to be marked recursively is the same as specifying it non-recursively. In order for a file to be marked, it must reside in either the current directory or one of its sub-directories.

In verbose mode, all files being marked that aren't already marked are listed. See **unmark**.

more Toggles output pagination via the UNIX **more** command. This command has no effect on the output of the forked **itstar**.

prompt *string* Sets the command-line prompt to *string*.

pwd Prints the name of the file system's current working directory.

status Displays **itsrestore** status including archive filename, blocking factor, current directory, verbose mode, prompt, Emacs-editing/history mode, and output pagination mode.

unmark [**-r**|**-n**] **files** [[**-r**|**-n**] *files*]
Specifies that previously marked files should be unmarked. The **-r** flag specifies that all files following it (until the next **-n** flag) are directory names, and all files in the subtree rooted at those directories are to be unmarked. The **-n** flag (the default) specifies that all files following it (until the next **-r** flag) are not to be unmarked recursively.

Specifying a directory to be unmarked non-recursively has no effect. Specifying a regular file to be unmarked recursively is the same as specifying it non-recursively. In order for a file to be marked, it must reside in either the current directory or one of its sub-directories.

In verbose mode, all files being unmarked that aren't already unmarked are listed. See **mark**.

verbose Enables/disables verbose mode. In verbose mode, **itsrestore** prints the names of the files as they are marked or unmarked.

Copyright © 1994 Thinking Machines Corporation

EXAMPLE

This is a continuation of the example presented in the **itsdump** man page.

```
% itsrestore
Retrieving directory info from archive(s) /dev/rst0...done
Current directory is big_project
Files will be restored into SFS directory sda1:/big_project
itsrestore% ls
big_project/sean/ big_project/may30
itsrestore% cd sean
itsrestore% ls abc tree
itsrestore% cd ..
itsrestore% mark -r sean itsrestore% display big_project/sean/
big_project/sean/abc
big_project/sean/tree
itsrestore% execute itstar -x -M -b 3200 -f /dev/rst0 -T -
itsrestore% exit
%
```

RESTRICTIONS

As with **itstar**, you must be owner of the extracted files, and you must have write permission on the directory to which you restore them. Also, if a file with the same name as an extracted file already exists in the directory, you must have write permission on that file, as it will be overwritten.

itsrestore cannot extract files or directories archived by UNIX **tar**.

SEE ALSO

itsdd **itsdump**
itsibmtape **itstar**

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

itstar

Archives a file to tape.

SYNTAX

itstar *command* [*options*] [*option_arguments*] [*files*]

ARGUMENTS

Commands

One of the following must be specified. **NOTE:** The preceding hyphen is optional for commands only.

-c Creates a new archive containing files. If files are not specified, no files are archived. If the archive file already exists, it is overwritten; the old contents are lost.

-d Compares the archive with the files in the file system and reports differences in file size, mode, owner, and contents; also reports if a file exists in the archive but not in the file system. If files are specified, they are compared with the tape and they must all exist in the archive. If files are not specified, all the files in the archive are compared.

-r Adds files to the end of the archive. Files must be specified. The archive file must already exist and must be in the proper format (this probably means it was created previously with **itstar**). If the archive is not in a format that **itstar** understands, the results will be unpredictable.

-t Displays a list of the files in the archive. If files are specified, only those specified that appear in the archive are listed.

-u Adds files to the end of the archive, but only under one of two conditions: if a file is not already in the archive or if a file is newer than the version in the archive (the time of last modification is compared). Files must be specified. This command can be very slow to execute.

Copyright © 1994 Thinking Machines Corporation

-x Extracts files from the archive. If files are not supplied, all the files in the archives are extracted.

-A Concatenates several archive files into one big archive file. All files should be archive files; these are appended to the end of the archive file on which **itstar** is operating (see the option **-f**). The original files are not changed. The UNIX **cat** command cannot be used to concatenate archive files because each archive contains an end-of-archive marker, so material added to the end by **cat** is ignored. The **-A** command removes the end-of-archive markers before appending archive files.

-D Deletes files from the archive. This command is extremely slow. Warning: Deleting files from an archive stored on magnetic tape may result in a scrambled archive, because there is no safe way, except for completely rewriting the archive, to delete files from a magnetic tape.

Options and Option Arguments

Zero or more of the options listed below may be specified. Some of the options are meaningful only with certain commands. The options below are grouped according to the commands with which they are used.

Options that do not take arguments may appear either after the command (with no hyphen and with no space between command and options), or separately (with the hyphen). The single exception is -version, which must be specified separately to avoid confusion with -v.

Options that take an argument must appear separately (with the hyphen) and must be immediately followed by the argument.

The following options are meaningful with any of the commands:

-b *number* Specifies a blocking factor for the archive; **itstar** reads and writes the archive in blocks of *number* x 512 bytes. The default blocking factor is set when **itstar** is compiled, and is usually 3200. A blocking factor of 3200 is fine for the StorageTek tape drive, but it is too high for most standard tape drives. We recommend using a blocking factor of 126 for standard reel tape drives, 1/4-inch Sun cartridge tape drives, and the like. (The maximum size of the blocking factor is limited only by the

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

maximum block size of the device containing the archive, or by the amount of available virtual memory.)

Use the same blocking factor when creating, updating, and extracting an archive.

-f *filename* Specifies the filename of the archive on which **itstar** operates.

Multiple archives are supported: you can specify up to four **-f** archive arguments. Specifying multiple archives is useful when using multi-volume mode with tape drives that take time to rewind, such as StorageTek. If the **itstar** command line is ... **-f /dev/rstc0 -f /dev/rstc1**, for example, **itstar** uses **/dev/rstc1** while it is waiting for **/dev/rstc0** to rewind, unload, and reload.

If **-f** is not supplied, but the shell environment variable TAPE exists, its value is used; otherwise, a default archive name (picked when **itstar** was compiled) is used. The default is normally set to **/dev/rstc0**, which is the first StorageTek tape drive (if the system has a StorageTek unit).

itstar supports remote access only to magnetic tape device files, but not to regular files. Therefore, if you are accessing a remote magnetic tape device, *filename* can be specified as *hostname* **:***filename* (*filename* must start with **/dev/**). If *hostname* is not indicated, the tape drive on the machine you're logged in to is used. If *hostname* is given, the tape drive on *hostname* is used. If *hostname* contains the **@** symbol, it is treated as *user***@***hostname* **:***filename*. If *hostname* indicates a remote tape drive, **itstar** invokes the command **rsh** (or **remsh**) to start up an **/etc/rmt** on the remote machine. If you give an alternate login name, it will be given to the **rsh**. Naturally, the remote machine must have a copy of **/etc/rmt**. The program **/etc/rmt** is free software from the University of California, and a copy of the source code can be found with the sources for **itstar**. This program must be modified to run on non-BSD4.3 UNIX systems.

If *filename* is **-**, **itstar** reads the archive from standard input (when listing or extracting), or writes it to standard output (when creating). If **-** is given when updating an archive, **itstar** reads the original archive from its standard input and writes the entire new archive to its standard output.

Copyright © 1994 Thinking Machines Corporation

-C *dir* Changes the working directory to **dir** before continuing. This option is usually interspersed with the files on which **itstar** is to operate. It is especially useful when several files from different directories are to be stored in the same directory in the archive. For example, executing the following command places on a tape the files **iggy** and **ziggy** from the current directory, followed by the file **melvin** from the directory **baz**.

```
% itstar -c iggy ziggy -C baz melvin
```

The file **melvin** is recorded in the archive under the precise name **melvin**, not **baz/melvin**. Thus, the archive contains three files that all appear to have come from the same directory; if the archive is extracted with **itstar -x**, all three files will be created in the current directory. In contrast, the following command records the third file in the archive under the name **bar/melvin**, so that **itstar -x** creates the third file in a subdirectory named **bar**.

```
% itstar -c iggy ziggy bar/melvin
```

-M Writes a multi-volume archive: one that may not fit on the medium used to store it. Instead of aborting when it cannot read or write any more data, **itstar** asks for a new volume. For example, if the archive is on a magnetic tape, the tape must be changed. Each volume of a multi-volume archive is an independent **itstar** archive. Any volume can be listed or extracted alone, as long as **-M** is not specified. However, if a file in the archive is split across volumes, the only way to extract it successfully is with a multi-volume extract command (**-xM**) starting on or before the volume where the file begins.

-N *date* Operates only on files whose modification or inode-changed times are newer than date. When creating an archive, only new files are written; when extracting an archive, only newer files are extracted. date must be quoted if it contains any spaces. It is parsed using **getdate**.

-R Prints, along with every message normally produced, the record number within the archive where the message occurred. This option is especially useful when reading damaged archives because it helps pinpoint the damaged sections. **-R** can also be useful when making a log of a file system backup tape. Its results allow you to locate the file you want to retrieve on

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

several backup tapes and choose the tape where the file appears closest to the front of the tape.

-T *filename* — Instead of taking the list of files on which to operate from the command line, reads it from *filename*. If *filename* is **-**, the list is read from standard input. Specifying both **-T -** and **-f -** at once can only be done with the **-c** command.

-V *name* — Writes out a volume header at the beginning of the archive. If **-M** is specified, each volume of the archive is given a volume header of name **Volume** *n*" where *n* is 1 for the first volume, 2 for the next, etc.

-v — Invokes verbose mode. Normally, the command to list an archive (**-t**) prints just the filenames and the other commands are silent. The command **-tv** prints a full line of information about each file, like the output of **cmls -l**. The **-v** option with any other command prints the name of each file operated on. The output from **-v** appears on the standard output except when creating or updating an archive to the standard output, in which case the output from **-v** is sent to the standard error.

-version — Prints the version number of **itstar** to the standard error. To avoid confusion with the **-v** option, **-version** must be given as a separate option preceded by a hyphen.

-w — Prints a message for each action **itstar** intends to take, requesting confirmation. If the response begins with **y** the action is performed; otherwise it is not. Actions requiring confirmation include adding a file to the archive, extracting a file from the archive, deleting a file from the archive, and deleting a file from disk. If **itstar** is reading the archive from the standard input, it opens the file **/dev/tty** to ask for confirmation.

-X *file* — Reads a list of filenames (actually regular expressions) from *file* and ignores files with those names. For example, **itstar -c -X foo**. prevents files in the current directory whose names end in **.o** from being added to the archive. Multiple **-X** options may be specified.

-Z, **-z** — Compresses the archive as it is written, or decompresses it as it is read, using the compress program. This option works on physical devices (tape drives, etc.) and remote files as well as on

Copyright © 1994 Thinking Machines Corporation

normal files; data to or from such devices or remote files is reblocked by another copy of the **itstar** program to enforce the specified (or default) block size. The default compression parameters are used; to override them run compress explicitly instead of using this option. This option cannot be used with the option **-M**, nor with the commands **-u**, **-r**, **-A**, and **-D**.

Options for Creating/Updating Archives

The following options are meaningful with the commands for creating or updating an archive (**-c**, **-r**, **-u**, **-A**, and **-D**). These options control which files are placed in an archive and the format in which the archive is written.

-G At the beginning of the archive, writes an entry for each of the directories on which **itstar** will operate. **-G** is used only when creating an incremental backup of a file system. The entry for each directory includes a list of all the files in the directory at the time of the backup, and a flag for each file indicating whether the file is going to be put in the archive. This information is used when doing a complete incremental restore. **NOTE:** This option causes **itstar** to create a non-standard archive that may not be readable by non-GNU versions of the UNIX **tar** program.

-h If a symbolic link is encountered, archives the file to which it is linked instead of simply recording the presence of a symbolic link. If the file is archived again, an entire second copy of it is archived instead of a link.

-l Prevents crossing of file system boundaries when archiving parts of a directory tree. This option does not affect files on the command line; these are archived even if they reside in various file systems. Rather, this option only affects files that are being archived because they reside in a directory that is being archived. Any files not archived because of **-l** are printed on the standard error. This option is useful for making full or incremental archival backups of a file system, as with the UNIX **dump** command.

-o Writes an old format archive, which does not include information about directories, pipes, FIFOs, contiguous files, or device files, and which specifies file ownership by numeric user and group IDs rather than by user and group names. In most

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

cases, a new format archive can be read by an old UNIX **tar** program without serious trouble, so this option should seldom be needed. When updating an archive, do not use **-o** unless the archive was created with **-o**.

-W Verifies the archive after writing it. Each volume is checked after it is written and any discrepancies are recorded on the standard error. Verification requires the archive to be on a medium capable of back-spacing. This means pipes, some cartridge tape drives, and some other devices cannot be verified.

Option for Updating Archives and Listing/Extracting Files

The following option is meaningful with the commands for updating an archive (**-r**, **-u**, **-A**, and **-D**), listing files (**-t**), and extracting files (**-x**).

-B If an attempt to read a block from the archive does not return a full block, keeps reading until a full block is obtained. This behavior is the default when **itstar** is reading an archive from standard input or from a remote machine. Without this option, **itstar** would fail in those situations where the pipe could not supply the amount requested in a single read.

Options for Listing and Extracting Files from Archives

The following options are meaningful with the command for listing files in an archive (**-t**) and extracting files from an archive (**-x**):

-G The archive is an incremental backup. The behavior of **-G** depends on which command it modifies.

With **-t**: for each directory in the archive, lists the files in that directory at the time the archive was created. The format of this list, while not easy for users to read, is unambiguous for a program; each filename is preceded by `Y` if the file is present in the archive, by `N` if the file is a directory, or by nothing if it is not in the archive. Each filename is terminated by a null character. The last file is followed by an additional null character and a newline to indicate the end of the data.

With **-x**: when the entry for a directory is found, deletes all files in that directory that are not listed in the archive. This behavior is convenient when restoring a damaged file system from a

Copyright © 1994 Thinking Machines Corporation

succession of incremental backups; it restores the entire state of the file system to that which obtained when the backup was made. Without **-G**, the file system will probably fill up with files that shouldn't exist any more.

-i Ignores blocks of zeros in the archive. Normally a block of zeros indicates the end of the archive. This option allows **itstar** to read archives containing blocks of zeros, such as damaged archives, or archives created by concatenating several archives together with **cat**. This option is not on by default because many versions of **tar** write garbage after blocks of zeros. Note that this option causes **itstar** to read to the end of the archive file, which may sometimes avoid problems when multiple files are stored on a single physical tape.

-K *filename* Begins extracting or listing the archive with the file *filename*; considers only the files starting at that point in the archive. This is useful if a previous attempt to extract files failed when it reached *filename* due to lack of free space.

-s Indicates that the list of filenames to be listed or extracted is sorted in the same order as the files in the archive. This option allows a large list of names to be used, even on a small machine that would not otherwise be able to hold all the names in memory at the same time. Such a sorted list can be created by executing **itstar -t** on the archive and editing its output. This option is unlikely to be needed on modern computer systems.

Options for Extracting Files from Archives

The following options are meaningful with the command for extracting files from an archive (`-x`).

-k Does not overwrite existing files with files of the same name from the archive.

-m Leaves the modification times of the files **itstar** extracts as the time when the files were extracted, instead of setting it to the time recorded in the archive.

-O Instead of creating the files specified, writes the contents of the files extracted to the standard output. This may be useful when extracting the files in order to send them through a pipeline.

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

-p Sets the modes (access permissions) of extracted files exactly as recorded in the archive. If this option is not used, the current **umask** setting limits the permissions on extracted files.

files One or more SDA files to be stored in an archive file. An archive file describes the names and contents of constituent files, providing a way to transport a group of files from one system to another, and to store several files on one tape while retaining their names.

When extracting or listing files, the files are treated as regular expressions, using almost the same syntax as the shell. The shell matches each substring between backslashes separately, while **itstar** matches the entire string at once, so some anomalies occur; for example, ***** or **?** can match a **/**. To specify a regular expression as an argument to **itstar**, quote it so the shell will not expand it.

WHERE EXECUTED

ITS tape controller.

DESCRIPTION

itsrestore provides for storing many SDA files into a single archive, which is a file that describes the names and contents of its constituent files. An archive file is the only way to store several files on one tape and retain their names. (A tape can store several files in sequence, but it stores no names, just their relative positions on the tape). **itsrestore** creates new archives, adds files to an existing archive, lists the files stored in an archive, and extracts files from an archive.

Creating an archive is useful for packaging a set of files for moving to another CM , as well as for making backup copies of a file system. **itsrestore** has special features for making incremental and full dumps of all the files in a file system.

The first argument to **itsrestore** must include exactly one command, and it may also include zero or more options. The command specifies whether to create a new archive (**-c**), compare an archive with files in the file system (**-d**), add files to an archive (**-r**, **-u**), list files in an archive (**-t**), extract files from an archive (**-x**), concatenate several archives together into one larger archive (**-A**), or delete specified files from an archive (**-D**).

Copyright © 1994 Thinking Machines Corporation

The next arguments to **itsrestore** may be one or more options. Since the options control the behavior of the commands, some of the options only make sense with certain commands. The options can be specified in either of two ways. The first method is as described above under **Syntax**; options that don't take an argument may be specified either with the command or individually preceded by a hyphen, while options that do take an argument are specified individually, preceded by a hyphen and immediately followed by the argument. The second method is that the option letters are specified with the command with no preceding hyphen, and the option's arguments, if any, follow in the same order as the options. The following two commands, which both accomplish the same thing, illustrate both methods:

```
itstar -c -v -b 20 -f /dev/rst0

itstar cvbf 20 /dev/rst0
```

The last argument to **itsrestore** is *files*, which consists of a list of the files and directories on which **itsrestore** is to operate. For example, they are the files **itsrestore** is to place in a new archive or extract from an existing archive. If a directory is named, **itsrestore** recursively operates on the directory, its files, and its subdirectories. If no *files* argument is supplied, the default behavior depends on which command was used. Some commands have no default and report an error if files are not specified. The files on which **itsrestore** is to operate can also be read from a file; see the **-T** option.

If a full pathname is specified when creating an archive, it is written to the archive without the hostname and initial backslash (**/**) so the files can later be read into a place other than the original location, and a warning is printed. If files are extracted from an archive that contains full pathnames, they are extracted relative to the current directory and a warning message is printed.

When reading an archive, **itsrestore** continues after finding an error.

The command **itsrestore** is derived from the GNU **tar** program, which was written by John Gilmore and modified by many people, and whose GNU enhancements were written by Jay Fenlason.

RESTRICTIONS

As in UNIX **tar**, a bug in the Bourne shell usually causes an extra newline to be written to the standard error when remote archives are used.

Version 7.3, April 1994
Copyright © 1994 Thinking Machines Corporation

As in UNIX **tar**, a bug in **dd** prevents turning off the x+y records in/out messages on the standard error when **itsdd** is used to reblock or transport an archive.

SEE ALSO

itsdd **itsdump**
itsibmtape **itsrestore**

Copyright © 1994 Thinking Machines Corporation