

#### 14. RECOVERY FROM CHECKDSK ERRORS

As explained in Section 6, when Tenex is restarted, whether initially or due to an auto-restart after a crash, the programs Bsys and Checkdsk are run to verify the consistency of the file system. If either of these programs detects errors which cannot be corrected automatically, the message "Tenex not available: Disk needs fixing" is broadcast to all terminals instead of the usual "Tenex in operation" message, and logins are prohibited from all terminals except the two in the Maxc room.

The following procedures require wheel or operator status and are intended principally for reference by system personnel. With some assistance from a user in the Maxc room, a system maintainer can perform these procedures from a home terminal. Only in extreme circumstances should non-system personnel attempt any of these procedures.

Errors detected by Bsys (which are usually reflected by some further errors detected by Checkdsk) indicate inconsistencies in the structure of user file directories. Fixing these requires a fairly intimate knowledge of the Tenex directory structure; this should be left to system personnel. Information about the structure of directories may be found in the Tenex Monitor Manual, section VII, pages 2-5. Other helpful information is available in the Bsys manual, pages 33-36. Copies of both these documents are kept in the Maxc room book case.

Checkdsk errors come in a number of guises. For each file with errors, data will be printed as in the following example: *(NOTE: Before looking at CheckDsk errors you may have to do a "Diablo Copy ON" in the upper AltIO window to get hard copy output of the error messages; then Boot Micro-Exec, Start Tenex.)*

<NEELY>MESSAGE.COPY;3	Filename
40050172166 MDA 0	} List of errors
140050172170 MDA 63	}
1 PTE	} Error
2 MDA	} summary

If there are many errors in a single file, Checkdsk will print out only the first few, followed by the summary. Study the output carefully.

First, note that "NOT IN BT" errors *have been corrected by Checkdsk*, so don't worry about them. If these were the only errors that occurred, Checkdsk wouldn't have complained and the system would have flown on.

The other kinds of errors reported by Checkdsk are more serious:

MDA	Multiply-assigned disk address
IDA	Illegal disk address
PTE	Page table error

MDA errors are the only ones that cause Tenex to prohibit users from logging in, since further file activity is likely to make the damage spread.

Note that you will have to use some *judgment* in discriminating between garbaged page tables and real MDA errors. A file with a garbaged page table will have an enormous error count (in the

hundreds) with many categories of errors (IDA, MDA, PTE, etc). This is frequently caused by an untimely Tenex crash occurring between directory update and page table update during new file creation, so that the page table for the file will not have been written on the disk yet and whatever was on that page before will be interpreted as a page table. This type of error may result in many *other* files getting bogus MDA errors because some of the entries in the garbaged page table look like valid disk addresses that happen already to be assigned.

For further confirmation that the problem is a garbaged (unwritten) page table, a QFD of the filename should reveal that it was created within a minute or two before the time of the last system crash. Such a file should be deleted using the following procedure:

```
@ENABLE password <cr>
!CONNECT <directory with bad file> <cr>
!DELETE <bad filename> <cr>
!EXPUNGE <cr>
!
```

This procedure causes the bad file to be expunged from the directory. A number of valid addresses possibly in use by other files may be deallocated, but don't worry about this. The system will generate a number of BUGCHKs for illegal disk addresses, but don't worry about this either. (Be sure DCHKSW is set to zero, however, to prevent the system from breakpointing on these errors). Run Checkdisk again after performing this surgery to make sure you did it right and that there is nothing else wrong. Checkdisk will reallocate pages incorrectly deallocated by the preceding procedure and will type out "NOT IN BT" for these.

```
!CONNECT SYSTEM <cr>
!CHECKDSK <cr>
REBUILD BIT TABLE? N
SCAN FOR DISK ADDRESSES? N
```

(This currently takes about 15 minutes).

After all files with garbaged page tables have been eliminated (if there were any), any further errors are considerably more serious, particularly MDA errors. MDA stands for multiply-allocated disk address, meaning that a particular page has somehow been assigned to more than one file. For each such error, Checkdisk has printed out the *second* file owning the page that it encountered in its scan of the file system; you do not yet know the name of the other owner of that page. Hence you should follow this procedure:

```
!CONNECT <directory containing file with MDA error> <cr>
!COPY <affected filename> GARBAGE <cr>
!DELETE <affected filename> <cr>
!EXPUNGE <cr>
!RENAME GARBAGE <affected filename> <cr>
```

Repeat this procedure for all affected files. You should be careful to type the <affected filename> in full, including version number, so you don't mistakenly fix up the wrong file.

Next, re-run Checkdisk as explained above. While running, Checkdisk will type out a number of

NOT IN BT errors whose disk addresses correspond to the disk addresses in the original MDA error printouts; the filenames typed out will be those of the *other owners* of the pages that were multiply assigned. It may not be obvious which owner of a page has the correct copy (a QFD of the filenames will include the write dates, which may give some indication; i.e. the file with the newer write date is more likely to have the correct data), but you have done the best that can be done by giving non-conflicting copies to everybody involved. Use SNDMSG to notify all users who have (potentially) lost files. Both the original MDA and the final NOT IN BT files are involved in the loss.

When you have pieced the filesystem back together to what you believe is a reasonable state, you should open the system to users by the following procedure:

```
!QUIT <cr>  
./_  
FACTSW[ 500000,,0 400000,,0 <cr>  
<control-P>  
ABORT  
.^  
!LOGOUT <cr>  
LOGOUT JOB .....  
<control-C>
```

After you type control-C, the auto-jobs should start logging in, and shortly thereafter "Tenex in operation" will be broadcast.