

### 24.7. Destroying a Maxc Account

Maxc directories are destroyed after either an explicit request from a Maxc directory owner (rare), after the sign out procedure that occurs when someone leaves PARC, or when we find out by some other means that a directory is dead. Ron Weaver is notified when someone leaves PARC, and he either destroys the Maxc directory (most desirable) or converts it to files-only (less desirable but hard to avoid if there are too many still-useful files archived out of the directory). Directories for users outside PARC may become dead, but we have no convenient method of automatic discovery.

Prior to destruction, the directory owner should delete unnecessary files and rename useful ones into the directory of a coworker. Users are sometimes sloppy about cleaning up their directories, so the person who is going to kill the directory should, as a safety measure, delete obsolete versions of files and archive current versions as follows:

```
!CONNECT olddirectory<cr>
!DELVER<cr>
!ARCHIVE FILE *.*;<cr>
---prints out what is going to be archived---
!DELETE *.*;<cr>
---deletes archived files, but not those about to be archived---
```

*Note:* If files exist on Maxc2, get someone to clean up the directory or move them to Maxc1 for archiving.

Then wait for the weekly or twice-weekly archiving run to take place. After that, all information about files will be contained in the archive directory, and you can continue with the procedures discussed below. If the directory does not have an archive directory, this safety procedure cannot be followed.

The person who is about to destroy the directory should list *one copy of the archive directory before destruction*, as follows:

```
!CONNECT olddirectory
!INTERROGATE *.*;&,
!!EVERYTHING<cr>
!!LPT<cr>
!!<cr>
```

This copy is stapled to the original directory application and filed in the CSL dead directory file.

When a directory is destroyed, any files remaining in it will be deleted. It is less automatic to retrieve archived files belonging to a deleted directory. We have avoided reusing directory numbers for defunct directories so that, if we have to, we can then recreate the directory to retrieve files from the archive. We expect to continue this policy until we run out of directory numbers (probably in 1980). Retaining a listing of the archive directory is a new practice, so directories destroyed before 19 April 1978 can readily be accessed only by recreating the directory.

The procedure for destroying a directory involves retrieving the original directory request form from our files (Haychan Sargent), marking it as destroyed, and refileing. For PARC directories, the form should also be considered for destroying IVY IFS directories for the person who is departing, if that is appropriate.

To destroy a Maxc directory, proceed as follows:

1. Login to Maxc1, enable yourself, and connect to the <ACCOUNTS> directory. Do a LD or WHO to verify that none of the other account maintainers are possibly modifying the account information at that time.
2. Edit the accounting files to remove the directory being destroyed. PROJECTLIST has to be edited for all directories, UACHK.TXT for login directories, and <SYSTEM>ARCHIVE-FILES-ONLY.TXT for files-only directories. <SYSTEM>SUBSYSTEMS.old-directory (if it exists) has to be archived for message-only directories.
3. After removing the appropriate entries from these files, destroy the directory with the following dialog:

```
!EcCREATE olddirectory,<cr> (Don't omit the ",")
[SUPERPASSWORD] GUESS<cr>
!!KILL<cr>
!!<cr>
[CONFIRM]<cr>
```

If you have a number of directories to destroy, you can delete all of them at this time, and then do the stuff below just once for all of them.

```
!RUNFIL <cr> COMMANDS FROM : UACHK.RUNFIL<cr>
!EcINITIALIZE ACCOUNTS<cr>
!PUPFTP MAXC2<cr>
*CONNECT ACCOUNTS<cr>
*PRESERVE VERSION<cr>
*STORE PROJECTLIST<cr><cr>
<User-Password incorrect
LOGIN (user) yourself yourpassword<cr>
*STORE <SYSTEM>UACHK.FILE<cr><cr>
*CONNECT SYSTEM
*STORE <SYSTEM>ARCHIVE-FILES-ONLY.TXT<cr><cr>
*Q<cr>
!DET<cr>
```

Chat to Maxc2

```
@ENABLE yourpassword<cr>
!CONN ACCOUNTS<cr>
!EcCREATE old-directory,<cr>
[SUPERPASSWORD] GUESS<cr>
!!KILL<cr>
!!<cr>
[CONFIRM]<cr>
```

If you have a number of directories to destroy, you can delete all of them at this time, and then do

the stuff below just once for all of them. If you deleted any SUBSYSTEMS.dirname files on Maxc1, then delete them on Maxc2 as well.

```
!EcINITIALIZE ACCOUNTS<cr>
!
```

For each directory destroyed, a history of the directory is left in <ACCOUNTS>DELETED-USER-PARAMETERS.directoryname(directorynumber), and if it has any archived files, the archive directory is moved to <ACCOUNTS>DELETED-ARCHIVE-DIRECTORY.directoryname(directory-number). Also, don't forget to edit the Grapevine Data Base (Section 24.5.)

#### 24.8. Operations on MESSAGE.TXT Files

It is sometimes necessary to create a mailbox in a directory that doesn't have one, to delete a mailbox, or to move a mailbox from Maxc1 to Maxc2, or vice versa. Since MESSAGE.TXT files have a funny "permanent" attribute that can't be set or cleared by any Exec commands, you have to create and destroy mailboxes using the MSGFIX subsystem, which requires that you enable yourself. It has commands for creating and deleting mailboxes (i.e., MESSAGE.TXT files). When a mailbox is deleted, its contents are copied into a file called OLDMESSAGE.TXT. The command sequences to MSGFIX are as follows:

```
!MSGFIX
*Create MESSAGE.TXT for directory: DIRNAME <cr>
*Q
!
```

to create a mailbox, or:

```
!MSGFIX
*Delete MESSAGE.TXT in directory: DIRNAME <cr>
*Q
!
```

to delete a mailbox.

It sometimes happens that a MESSAGE.TXT file has an excessive number of pages because the file has not been truncated for some reason. This can be remedied by using the following safe and simple procedure:

```
!CONN dirname<cr>
!RENAME MESSAGE.TXT FOO<cr>
!APPEND FOO MESSAGE.TXT<cr>
!DELETE FOO<cr>
```

In the case of a race occurring, the order of the old and new messages gets reversed, but no messages are lost. Note that MESSAGE.TXT files acquire extra pages only when manipulated by MSG, so only MSG users (never Laurel users) should suffer this problem.

When a MESSAGE.TXT file suffers a disk error, the recovery procedure is:

```
!CONN dirname<cr>
!RENAME MESSAGE.TXT BADMESSAGE.TXT<cr>
```

```

!TECO<cr>
*;Y<esc>
Input File: BADMESSAGE.TXT<cr>
(unexpected daya error interrupt)
*;U<esc>
Output File: BADMESSAGE.TXT<cr>
*

```

Then tell the user that BADMESSAGE.TXT may be retrieved for examination with Bravo or whatever. The bad data should NOT be put back into MESSAGE.TXT, as that might cause the internal structure of MESSAGE.TXT to be ruined.

#### 24.9. Reinstantiating a Destroyed Directory

Sometimes a person whose Maxc directory has been killed needs to have that directory recreated. This could be done by creating a brand new directory, but it is usually more convenient to recreate the directory from the information that was saved when it was previously destroyed.

The way to do this is to retrieve <ACCOUNTS>DELETED-USER-PARAMETERS.DIRNAME-(DIRNUMBER) and <ACCOUNTS>DELETED-ARCHIVE-DIRECTORY.DIRNAME(DIRNUMBER) from archival storage (i.e., they will usually have been archived by the time that you want to recreate the directory); in some cases the DELETED-ARCHIVE-DIRECTORY file won't exist. Then recreate the directory from these two files, as follows:

```

!EcRECREATE DIRNAME<cr>
[sUPERPASSWORD] GUESS<cr>

```

This recreation procedure only works if the directory number has not been reassigned to some new directory in the interim. If it has been reassigned then you must treat the creation as if it were a new directory. Some special things must be done to the Archive-Directory if one existed before. (See Ed Taft for details.)

This reinstates the directory on Maxc1. Then add DIRNAME to PROJECTLIST and either UACHK.TXT or <SYSTEM>ARCHIVE-FILES-ONLY.TXT just as though a brand new directory were being created, and carry out the other procedures associated with creating a new directory on Maxc1. Then copy <ACCOUNTS>DELETED-USER-PARAMETERS.DIRNAME(DIRNUMBER) to Maxc2, RECREATE the directory on Maxc2, etc.

If the directory should have a MESSAGE.TXT file, then you should create the MESSAGE.TXT file with MSGFIX as discussed in the previous section.

#### 24.10. Retrieving Archived Files for Defunct Directories

The easiest way to do this is to recreate the directory as discussed in the previous section, and then retrieve the files from archival storage in the normal way.

An alternative method of retrieving archived files from defunct directories is to find out on what tape the file is archived, and use the BSYS "Single File" command to copy it from the tape to a disk file in some existing directory. Unless the archive directory for the deleted directory was listed prior to destruction, the tape containing a particular archived file will generally be unknown. Directories destroyed after 19 April 1978 have a listing of the archive directory on file in the CSL dead directory file.

#### **24.11. Printing Accounting Information**

Running <ACCOUNTANT>USERAC.SAV will produce a printout of all the information on all the directories (requires enabled wheel status). The various options for this subsystem will be revealed when you type "H" as a subcommand.

The accounting printout periodically posted on the CSL bulletin board is obtained by connecting to the <ACCOUNTANT> directory and running <ACCOUNTANT>MAXC-ACCT, which renames the fact files from the <SYSTEM> directory into <ACCOUNTS> and churns through all the fact files over the time period selected.

RUNFIL <ACCOUNTS>HOGS.RUN will produce an output file named HOGS.DATE that shows the disk storage in use and the storage tied up by obsolete versions of files or files unreferenced in 90 days.