

SCAVENGER

February 28, 1975

1

SCAVENGER

A subsystem for checking and correcting disk packs is available as SCAVENGER. Invoke it with no parameters and it will give you an opportunity to (1) change disks and (2) prevent it from altering your disk seriously (see below).

The scavenger does the following:

1. Corrects header blocks, prompting for confirmation.
2. Corrects check sum errors, by re-writing whatever came in, prompting for confirmation.
3. Discovers all well-formed files and all free pages. Any disk page (except page 0) that is neither free nor part of a well-formed file is considered bad.
4. Makes the serial numbers of all well-formed files are distinct.
5. Corrects the system's notion of what pages are free.
6. Corrects the system's latest serial number.
7. Corrects the directory to contain precisely the well-formed files. If a directory entry points into a chain of bad pages it attempts to salvage the file. If need be a directory is created from scratch.
8. Links all bad, unsalvaged pages together as part of the file Garbage.\$.
9. Describes all changes to the disk in the file ScavengerLog, even those which were not actually performed.

The data in bad pages is not changed so you can attempt to reconstruct a lost file by suitable operations on Garbage.\$, consulting ScavengerLog to interpret its contents.

A hopelessly smashed disk may be put back in shape by the following:

1. Invoke scavenger on a good disk and answer yes to "Do you want to change disks?"
2. Replace the good disk with the bad one.
3. Answer yes to "Is the new disk ready?" when the yellow ready light comes on.
4. Answer yes to "May I alter your disk to corrct errors?"

When the scavenger exits your system may crash if certain files like Executive.run were lost. You should take precautions to avoid losing vital files (such as QUICKing your disk to another disk pack prior to running SCAVENGER).

↑L

SCAVENGER

February 28, 1975

2

PARC information

The following, more or less independent, procedure can be used to recover vital files that might have been lost during scavenging.

1. Invoke FTP on a good disk.
2. At an early point in the dialogue replace the good disk with the bad one and wait for the yellow ready light to come on.
3. Retrieve the needed files from MAXC (Executive.Run and FTP are the minimum required, I think.)
4. Quit out of FTP.
5. Run the scavenger. It will correct the DiskDescriptor file which became inaccurate during this process.

↑L