

NAME salv -- file system salvage

SYNOPSIS /etc/salv_____ filesystem [-akfs___]

DESCRIPTION

`salv__` will place a given file system in a consistent state with almost no loss of information. This is the first step in putting things together after a bad crash. Salv performs the following functions:

A valid free list is constructed.

The previous step is always performed; the following steps are performed only if the "a" option is given. If the file system's only defect is missing blocks, "a" should not be specified.

All bad pointers in the file system are zeroed.

All duplicate pointers to the same block are resolved by changing one of the pointers to point at a new block containing a copy of the data.

Inodes (not directory entries) for special files are generated (mode 16).

Files whose size is too large for the number of blocks they contain (after bad pointers are zeroed) have their size revised downward.

The file system should be unmounted while it is being salvaged. In cases of extreme need the permanently mounted file system may be salvaged; in such a case the system must be rebooted before it has a chance to write out the old, bad super-block.

The "k", "f", and "s" options tell salv what magic numbers to use to generate the size of the free list and the i-node map. "k" is default (RK disk); "f" is RF; "s" is RK with swap space on it. If salv is to be used away from the mother system its code should be checked to verify the numbers.

After a salv, files may be safely created and removed without causing more trouble. If the "a" option had to be used, a dcheck (VIII) should be done to find the degree of the damage to the hierarchy.

SEE ALSO check(VIII), dcheck(VIII)

BUGS In only one (known) way does `salv__` destroy information: if some random block appears to be an indirect block for a file, all "bad pointers" (for example, ASCII text) in it will be zeroed. If the block also appears in another file, it may be scribbled on before it is copied.