

3/19/70

"Disk process" structure

[in extreme cases, might be more than one actual process involved]

2 directories

- 1) a permanent directory (can be shared by several disk processes)
- 2) a temporary directory (1 per ~~per~~ disk process)

"Log in procedure"

Constructs temporary directory from info obtained from permanent directory

other items

- 3) list of scratch eis objects (mostly c-lists, operations, class codes?)

Subprocess ~~operations~~ operations

- 1) construction of a sub process creates a single operation to call the sub process
- 2) call the sub process with the single operation and it returns many ops [for complicated sub processes only]
- 3) or, it creates a "user" sub process and feeds that user sub process many ops.
- 4) The single op and the many ops are either
 1. eis codes
 2. stub eis objects
 3. simple objects

4) \emptyset ecs objects maintained by ^{directory} ~~the~~ system

ecs goodies
global ecs objects
scnten objects ?
files? (dir)

within open list
for each sub process?

when sub process destroyed
it closes all objects opened
by the sub process. [closing a
~~sub process~~ scnten object destroys it]

or B)

ecs goodies
global ecs objects

) open w.r.t only
for the process as a whole

scnten c-list ~~(dir)~~) for each sub process
scnten file ~~(dir file)~~

a \emptyset sub process destroys all but scnten c-list, scnten file;
Then jmp calls directory system to be destroyed.
which destroys it & scnten c-list, scnten file

A) (elaboration)

all opening done through directory system

2 ways to open a scnten object.

during sub process creation

by call from a running sub process with the object key passed to
the directory system already created.

A) continued

For each subprocess a list of ^{open items} ~~things~~ as follows

- i) type —
 - swapfile
 - stackfile
 - directoy
 - subprocess descriptor
 - pid number
 - global pid object
 - subprocess pid object

ii) scratch file [if any, item will be closed upon destruction of
the subprocess]

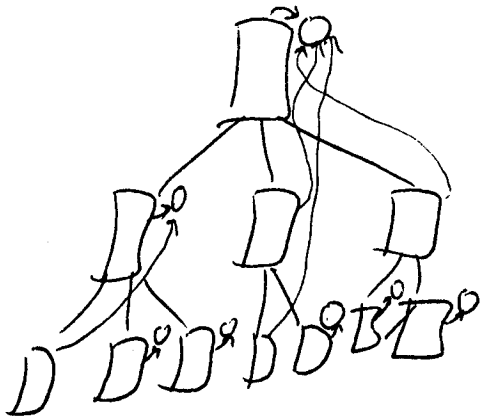
iii) open count [closed when open count goes to zero]

iv) c-list index of the object

v) unique name

uses at most 2 words per open object per subprocess opening it.

allocation account data associated with directories



directory ownership

~~The account data for each directory is either~~

- 1) directly associated with certain directories is an "account block"
- 2) no 2 directories have same "account block" directly associated with them
- 3) associated with each directory is an "account block"
- 4) if a directory has a directly associated "account block", then the associated "account block" is the directly associated "account block"
- 5) if a directory has no directly associated "account block" then the associated "account block" must be same as associated "account block" of father.
- 6) privileges of a directly associated "account block" must be obtained from the associated "account block" of the father