

3/5/70

## ecs goodie implementation

[note that the actual objects used do not have to be ecs objects!]

### A] structures

#### i) HASH TABLE

keyed by unique name & points to c-list & contains an open count

#### ii) C-LIST

contains current object for an

iii) Local open list in each process [contains local open count]

### B] Representation in directory

a unique name + Type field (1 word)

↳ 18 bit part

### C] ecs goodie capability

contains (in databand) a unique name & type field

### D] operations

#### I) creation of ecs goodie in directory

[may not be needed, may have an operation that creates ecs goodies outside directories, and use hard link to get them into directories]

present a type field (data) (along with other data info for creating a directory entry)

creates an entry with a new unique name field, and type field that presented

#### II) get an ecs goodie from directory [may be a uniform operation for all objects]

returns an ecs capability with:

Type = "ecs goodie"

options = Those indicated by the access key

data = unique name + type field

III) open eis goodie with a given object

presents eis goodie capability

object capability

(destruction options must be on)

error if at least: i) proper bit not on in eis goodie options bits.

or ii) ~~type field in data word of eis goodie capability~~  
~~does not match that of presented object~~

Then the eis goodie unique name field is looked up in the Hash table

If found in the Hash table, F- return

if not, make an entry in the hash table for this unique name,

point it to a free position in the c-list, place a copy

of the presented object capability at this location in

the c-list. and set open count in the Hash table

entry to 1. (also enter in local process open list)

and set local open count to 1

if actual EIS object  
object moved

system alloc bck - charged as freed space

#### IV) open eis goodie (no object presented)

present & eis goodie capability

C-list index

look up unique name of data uid of eis goodie capability in hash table.

if not found, F-return

if found, ~~compare type field of data uid of eis goodie capability~~  
with that of the associated C-list entry.

~~if no match, error~~

~~if match, set open count to 1~~

return a copy of the associated C-list  
entry capability masked by the option

if actually EES object - bits of the given eis goodie capability.  
change forced EES open to local process

Check local process  
open list. If in the  
local list, bump local  
count by 1. If not in the  
local list, place in local  
list with local count = 1, and  
bump open count in hash table  
by 1

#### V) close an eis goodie

present & eis goodie capability

check local process open list. If in the local list, decrease local count by 1.

If local count goes to zero, remove from local  
list and decrease hash table entry open

count by 1. If hash table entry open count  
goes to zero, delete hash table entry.

If not in the local list, error

return forced EES if actual EES object  
destroy obj if global zero and  
if EES object