Sept 11, 1970

Translation basic command processor

This is essentially a super simple programming-language

I) internal state - 2 parts

A) directory ~~list~~ scan list.

a seq of pairs, each pair consists of

i) a directory

ii) an access key — can ∃ 2 ACLs for 1 directory?

B) a ~~fix~~ set of variables [max of ? ... names]

~~each variable has a value which is a pair:~~

~~i) [illegible]~~

i) each variable has a name (max size ?) (ID in)

ii) each variable has a value which is one of:

a) an internal action

b) an object (capability) ) + ACL

~~an object pair directory access key~~

II) (internal) basic action, called 'fetch'.

input is a name

procedure is as follows:

A) if the ~~identifier~~ is null, a null access way is returned

B) else if there is a variable with the input name, value is value of the variable.

C) else if no variable of that name, each pair in the directory scan list is ~~executed~~ processed in turn. If the input name can be found in the directory and the access key checks, the resulting object is the value.

D) else the fetch fails.

III) command action:

a command, a seq of identifiers separated by commas, is terminated by ? blank or period. is typed in and processed as follows:

The 1st identifier, called the act, is run through the 'fetch' action.

a) if an internal action is obtained, it is executed, see below

b) if an object is obtained, and is a file (sub process description) a sub process is formed and called (with what as params?)

proposals for parameters

See sub process description for internal commands

Fetch, store, and
Fathers share and uses
result as value of store

IV) ~~Initial value of variables~~

A) Fetch ~~Get~~ - value an internal action

The ~~first~~ second identifier is taken as a variable name,

~~The value of the variable~~, The third identifier is fetched and resulting object placed as value of the variable

~~If a more identifiers, each is fetched and results placed as values of the variable.~~

Certain variables can not be ~~set~~ changed !! They are

Fetch ~~Get~~

perm

Temp

SYS

Scan
Kill
SET
NORM
RESET

B) perm - value ~~[illegible]~~ is users permanent directory, ~~[illegible]~~

C) Temp - value ~~[illegible]~~ users temporary directory, ~~[illegible]~~

D) SYS - value system directory, w.r. ~~[illegible]~~ no null access key exists

~~[illegible]~~

Solve up to here 3:30

E) scan - value an internal action

each successive identifier is fetched,
(values must be, alternately, directory and access keys
~~fetched successively~~) and a new directory scan list
is made up consisting of these fetched
values.

F) null - The 2nd identifier is taken as a variable name, and that variable deleted.

V) initial scan list.
is result of

~~SCAN, SYS, norm, temp, perm~~

SCAN, SYS, norm, temp, perm.

G) SET - The 2nd identifier is taken as a variable name,
The 4th identifier is fetched, and ~~is assumed to be~~ BETTER a directory
The 5th identifier is fetched, and ~~is assumed to be~~ BETTER an
access key ~~if null, fetch access key~~
The 3rd identifier is looked up in the fetched directory
with the fetched access key. The object found is placed
as value of the variable (2nd identifier)

H) norm - an access key, for use with system directory

I) reset - an action that kills all except the fixed variables, then
~~resets~~ resets initial scan list