Subprocess descriptors

These objects are implemented as low level disk files. They
are used to direct the construction of subprocesses.

open

open mearly guarantees that the descriptor will remain in ecs.
It does not return a new capability.

Create a subprocess

1st parameter of the creation call will be the subprocess descriptor
subsequent parameters may be required for the construction. [This
will depend on the subprocess]

This call will return a block of 1 or more objects. [The number
depends on the subprocess]. The 1st object returned will be
an operation for ~~doing~~ makeing an initial call on the subprocess.

Create a sub process descripter

This operation requires the usual parameters for creation
of a disk object that is to be represented by a low level disk
file.

It also requires the presentation of a block of data which
describes the subprocess, plus a block of additional parameters.
[for the later do we have sufficient ecs facilities?]

The following will specify what information must be in the block
of data, but not neccessarily the exact form. That will be defined
by the implementer.

The block of data will contain a number of sections to define various
things related to the subprocess.

c) <u>object descriptors</u>

a number of sections require the specification of particular objects. A uniform method is used for this specification. It is 1 or more words of data as follows:

(1st wd)

1	18	18	18	5
T	N3	N2	N1	class

<u>$T=1$</u> indicates that the object is to be opened. certain objects will be opened regardless of T, i.e. files used in maps and the local c-list.

<u>class</u>

0   end marker for a list of objects

1   scratch object # $N1$ (see section on scratch objects)

2   parameter $N1$ after subprocess descriptor in subprocess creation call.

3   Hard link, given as parameter $N1$ in additional parameters of subprocess description creation call.

4   soft link, Hard link to directory given as parameter $N1$ in additional parameters of subprocess descriptor creation call, Access key given as parameter $N2$ ....

5   Null object [used at least in c-lists]

[text name given in $N3$ following wds]
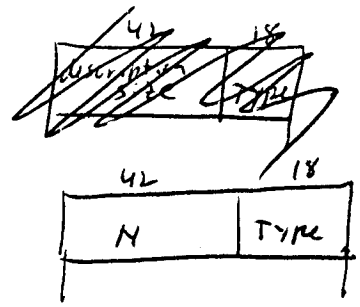
~~note added 4/4/00~~

note: 3 and 4 are supposed to mirror Hard and soft links in directories, so if those ideas change, these specifications will change accordingly.

I) scratch objects

Certain objects are constructed new at the time the subprocess is constructed. If these are disk objects they are placed in the process temporary directory with the scratch bit on. If ECS objects they should go into a list local to the process (?).

A list of descriptions of these objects is given as one section of the defining block of data.

each object is ~~data~~ described by a sequence of wds:

42	18
N	Type

Type	object
0	end of list marker
1	swapped file , N is shape
2	c-list , N is length
3	approved standard scratchfile
4	operation , N is number of wds follows to describe it.

    [form of rest of this description not yet defined]

Could consist of:

      seq of wds   0 for datum

                    option bits + type   for object

  + a class code object description

II) parameter description list

This section describes the parameters expected at subprocess creation time.

The list is preceeded by a count of words in the list.

The words in the list give the option bits and type for each parameter.

III) Basic subprocess

a list of items:

item	How given
Class code	object description
Class code of father	"
# map entries	an integer
compiled mapsize	"
sub process field length	"
entry point	"
i-list	object description

IV) <u>contents of map</u>

    a list of items formed as follows:

            object description [of a file]

            file address

            cm address

            count

            rw flag

    (terminated by an object description of type 0, without file address, ..., rw flag)

V) <u>Contents of c-list</u>

    list of object descriptors

VI) <u>objects to return at subprocess creation time</u>

    list of object descriptors

        [1st must be an operation to call this subprocess]

VII   <u>list of addresses of hard links</u>

    for the convenience of Load & Dump disk, a list of all words within this descriptor containing hard links

    (constructed by system, not supplied in call)