Juvel

Dump-load

① Scan the tree of directories (along ownership branches) and dump on tape all files, directories, subprocess descriptors. (needs a subroutine for dumping a lowlevel disk file onto tape - each such lowlevel file should be preceded with a flag to indicate what kind of object it is)
assume 1st item dumped is the master directory (root of tree)

②
Load the tape onto disk as a set of lowlevel disk files. while doing so, make up a hash table to give new unique name, (1st address) as a function of old unique name. Also flag each entry in the hash table as to type of object.
also remember where 1st object loaded (master directory)

scan the hash table linearly. for each entry that is a directory or subprocess descriptor, read in the lowlevel disk file that represents it, find all unique names and addresses occurring in the file and replace them with new correct values from the hash table.

Note: This ignores the problem of allocation blocks as I do not how understand them. specs from Bruce?

Basic hard parts

① lowlevel disk file dump and load routines (need to be fast, overlapped IO etc)
② routines to scan a file representing a directory or subprocess descriptor and find all unique name - disk address pairs to be fixed
③ Hash table routines.