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 or to tape.
   Each such lowlevel file should be preceded with a flag to
   indicate what kind of object this 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 your hash table to give new unique name
   at 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)

3. 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
   don't now understand them. specs from dave?

Basic hardware parts

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