MASSACHUSETTS INSTITUTE OF TECHNOLOGY
A. I. LABORATORY

Artificial Intelligence
Memo No. 214 March 1971

LINKING LOADER FOR MIDAS

Peter Samson

This memo was originally printed as MAC Memo 268, January 31, 1966.

Work reported herein was conducted at the Artificial Intelligence Laboratory, a Massachusetts Institute of Technology research program supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by the Office of Naval Research under Contract Number N00014-70-A-0362-0002.

The MIDAS Linking Loader is a PDP-6 program to load relocatable-format output from the MIDAS assembler, with facilities to handle symbolic cross-references between independently assembled programs.. Although it is arranged primarily to load from DECtape, the loader is able to load paper-tape relocatable programs.

To use the loader, load it off the MACDMP SYSTEM tape as the file STINK. (A file STINK NEW may exist, repairing old bugs or introducing new features.) Then the loader expects commands to be typed in on the on-line Teletype; two successive ALT MODE characters terminate the string of commands. The commands in a string are not performed until the string is thus terminated. While a command string has not been terminated, RUBOUT will erase the last typed-in character (and type it out again as a reminder). A command string may contain any number of commands, and the effect is the same whether the commands are together in one string or are in successively typed-in strings each delimited by two ALT MODEs.

The loader maintains two tables whose contents may change as programs are loaded: (a) the Loader Table, which contains definitions of global symbols and unresolved virtual usages; (b) the local symbol table, containing all program names, and the local symbols for each program for which their loading was requested.

In the following command descriptions, n is an octal number, or one of the command characters said to have a value. ($) is ALT MODE, which echoes out as $. ␣ represents SPACE.

Command Form	Meaning
P	Set to read from the paper tape reader. (Tape must be in the reader, and the reader must be on when this command is performed.)
nMname1‿name2 $	Set to read from beginning of file name1 name 2 on DECtape unit n. (If n is omitted, the last DECtape mentioned is assumed.)
N	Load selected input file without local symbols.
L	Load selected input file, saving local symbol definitions for DDT. (N and L set the Current Starting Address to that specified in the program loaded if that is not Ø.)
T	Copy all defined global symbols in the Loader Table into the local symbol table (for DDT); then delete same from the Loader Table.
D	Read in the relocatable version of DDT from DECtape unit 1, tell it of all symbols in the local symbol table; wipe out the loader and transfer control to DDT.
nF	List files of DECtape unit n. (As in the M command, the argument n may be omitted.)
G	Transfer control to the Current Starting Address.
nG	Set the Current Starting Address to n and transfer control thereto.
n=	Print the value of n as an octal integer.
X	Has the value of the Current Starting Address.
E	Has the value of the lowest address currently used by the loader.
K	Delete all local and global symbols from the local symbol table and the Loader Table.
Z	Zero core except registers 2Ø through 37 and the loader (from E up).
n[	Print contents of location n.

B

Has the current value of program relocation.

nR

Set program relocation to n.

nC

Set common relocation to n.

S

Print storage map: Each program in core has one line in the map. At the left is the program name, and at the right in octal is a word whose right half is the first location used by the program and whose left half is the last location used by the program.

?

Print storage map and missing list (short form). Each program loaded appears as follows: One line with the program name at the left and the first address used by the program at the right; any number of lines indented one space, each listing an undefined symbol used in that program, with the address of its first use therein. Symbols are global unless preceded by *, meaning local.

n?

Print storage map and missing list (long form). (Here the value of n is immaterial, but an argument must be given.) Like ? with the following changes: (a) following the program name is a 36-bit word in octal with first and last addresses as for the S command; (b) the address is given of each reference to each undefined symbol.

n <sym>

Define symbol sym with the value n. The symbol will be global unless a * is typed somewhere between <and >.

Error Message

Meaning

SCE adr

Storage capacity exceeded. The program being loaded collided with the loader at adr.

UGA adr sym

Undefined global assignment. The global symbol sym was undefined when needed by the loader to perform a parameter assignment or location assignment. The current loading address is adr.

MDG adr sym

Multiply defined global. A defined global appeared to the left of a : when adr, the current location, did not equal the value of the global. It was not redefined.

CKS	Checksum error.
FNF	File not found on DECtape specified.
TMS num	Too much symbols: occurs when loading DDT, and that DDT + symbols + program exceeds storage available by num registers.
ILM	Illegal memory reference: an error by the loader.

A tape labelled LIBRARY is available, containing various useful subroutines in the file LIBRAR 1. Up-to-date details are posted in the PDP-6 room. Each program in the library file was assembled with the .LIBRA pseudoinstruction, and so will be loaded only if in the Loader Table is a request for a global symbol defined in that program. Therefore the library file should not be loaded until all programs have been loaded which make reference to the library subroutines.

Command String Examples

(a1) To load the program APLHA RALPHA from DECtape unit 3, the program SUBR 1 from unit 2, and the program BR from unit 2, then to get a storage map and missing list:

```
3MALPHA␣RALPHA$ L2MSUBR␣1$ LMSUBR␣2$ L?$$
```

In this example, L was used for each program to load its local symbols. the N command could have been used in each case instead not to load local symbols.

(a2) Then to go to DDT:

```
TD$$
```

(b) To load the program in the paper-tape reader and transfer to its starting address:

```
PNG$$
```

The L command was not used here because DDT was not requested.

(c) To load PROG REL from DECtape unit 1 and the requested library routines from unit 4:

```
    1MPROG␣REL$⃝ L4MLIBRAR␣1 $⃝N$⃝$⃝
```