Dear Joel:

I have spent some time with the listing you gave me, and I have a number of random comments. Some of them will just be repeats of what I told you the other day, but it is well that all this shit be written down somewhere.

(1) eor should not have character 12(8), print name EOR, as its APVAL. Instead, the apval of eor should be character 55(8), print name 55(8), otherwise known as carriage return. This is a slight refinement of the scheme I suggested the other day, which introduced a superfluous APVAL for output EOR, distinct from the perfectly good existing APVAL for input EOR.

(2) We were much too shortsighted in discussing the LISP I/O structure. There should be two user-accessible calls, one each for input and output. These calls will be devicesetters, with the possibilities DISK, TTYPE, BOFFO. In the DISK case, two additional parameters will specify the file name.

This scheme will unify startread, advance, endread, the disk i/o stuff, pack, print, and so on.

(3) EOF from the disk should not be an error. It should read as EOF, if considered as a whole S-expression by itself. If it is encountered while trying to fill out an S-expression, it should give rise to a console message, with the option of going on and having EOF serve as an endless source of rpars.

(4) The present disk error procedure is overelaborate and simultaneously inadequate. After a disk error, the system should type 'disk error' and go into a LISTEN which will exit to redo the call. Most likely, the user will quit out of this LISTEN, save LISP, prnter, diddle with his files, resume, STOP the LISTEN, and thus continue. But the listen might be good for something. In any event, the procedure ought to be a LISP function, so that a user with different ideas about error recovery could (to an extent) substitute them without absolute patching. Let's not get into *that* bind again.

(5) Also on the disk, we need a user-accessible file-existence test. FSTATE and all that.

(6) What is EXPLODE? I must have missed something.

(7) Flaptraps and dcts should never bomb the system out. There should be a typed message, and the registers should be set to 0 for underflow, 377777777777 for overflow, and so on. If the user wants to take the computation down, that is his business.

(8) Reclaim should have the value list∠number of free words; number of full words; depth of pdl_/.

(9) As we said the other day, all disk output should be line-marked.

(10) The wall-paper trap should be suppressed.

(11) There should be a permanent object TAB, with apval pointing to character 72(8). That character should have PNAME 72(8).

(12) Lots and lots of supervisor code should be separate LAP pieces. Rather than say 'You users may reassemble LISP any way you want. If you don't want the character-handling routines, take them out, etc.' you should say 'LISP is a supervisor with a large library. The command will have a lot of this library in it, but you can build your own by starting from SOANDSO SAVED, which is little but the supervisor and LOAD.

(13) EOF should serve as STOP to LOAD and to LISP itself.

(14) Given the uniform I/O structure suggested in (2), it should be easy to let the user pick up input in units other than characters or S-expressions, just as the supervisor does with RD.

(15) If an impossible PNAME (too long, illegal characters, whatever) comes in, it should be printed in some manner (octal in some cases), and the user should be asked to type the PNAME which was intended. Number conversion problems fall under this heading.

(16) There was some talk a while back about a new, jazzed-up version of SUBLIS. Is it the one you have?

(17) The pairlist errors ought not to bomb the system. There should be a call to a LISP function which *might* bomb the system, but which might alternately go into an extended song and dance getting advice from the user.

(18) The same goes for the SPREAD error. Here and there, it will be necessary to call the error-catching function with enough argument information to allow some freedom of choice. For example, the SPREAD error-catcher must certainly have the list (too long) of supposed arguments.

(19) There is all sorts of garbage code around, going back to the tape system. The CH3 error, for example. Also the ERROR1 function, which is the kludge of all time.

(20) There are a number of errors connected with undefinition: Functions, go's, arguments, sets and setqs. In all these cases, the system should call a LISP function to type a complaint and go into a LISTEN which will come back to retry the offending operation if it is STOPped. In the case of set and setq, an a-list-extending function should be provided.

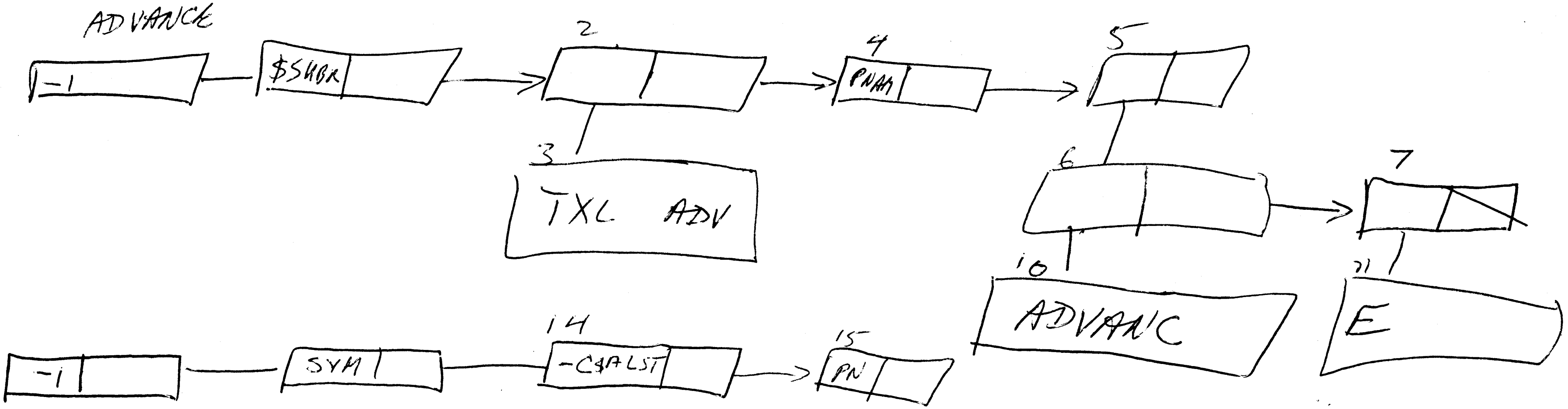
(21) How does expt work? If it takes logs, which it probably does, why isn't a log function available to the user? Why can't it take negative things to integral powers? Probably expt hasn't been recoded since the 704; probably it could be significantly jazzed up by stealing a replacement routine from the MAD library.

(22) Unsatisfied conditionals should have value NIL. They should never be errors.

(23) Colon and null characters might be useful. $IL35$ and $IL57$ certainly aren't.

(24) Lap errors should chncom in such a way that restart will cause the lap to be retried. Lap errors are almost always file problems, for which LISTEN would be useless.

F


ADVANCE
-1
$SUBR
2
3
TXL ADV
4
PNAM
5
6
7
10
ADVANC
11
E
-1
SYM
14
-C$ALST
15
PN

~~E$FWC~~
~~E$FS~~

Not sent!

```
TYPSET REPRF
W 121.0
 FILE  REPRF(MEMO) NOT FOUND.
INPUT
DEAR BOB,
    I HAVE CHANGED THE SYSTEM IN H"THE FOLLOWING RESPECTS.
1)CR IS AN ATOM WITH 55(8) AS AN APVAL
2)TYTAB IS AN ATOM WITH 72(8) AS AN APVAL,WHICH IS DIFFERENT
FROM $EOR$ OF ADVANCE.
3)PUNCH IS IN LINE-MARKED FORM, WITH NO ATOM SPLITTING

WITH RESPECTS TO YOUR REMARKS,
   1)YOU ANTICIPATED MY KLUDGE WRT $EOR$
   2)IT WOULD BE NICE TO DO THAT,YOU OVERESTIME"ATE THE USE OF
ADVANCE,ETC. AS FAR AS I KNOW ONLY ONE PERSON HAS USED IT THIS
YEAR.
   3)YOU MISSED A BIG KLUDGE IN THE DISK RA"EAD PROGRAM. I NEVER
CHECK FOR AN EOF. MAYBE I SHOULD, BUR"T I DO NOT THINK SO. STOP
IS A VERY NICE DEVICE AND HAS CAUSED US NO PROBLEMS SO FAR.
   4)YOUR COMMENT ON THE DISK IO ERROR IS REALLY THE SAN"ME  ONE
YOU HAVE MADE ABOUT THE OTHER ERRORS.
   5)MAYBE, I DONT THINK ANYBODY WOULD USE IT.
   6)EXPLODE RESULTS IN A LIST OF THE CHARACTERS WHICH FORM THE PRINT
VALUE OF THE S EXPRESSION GIVEN AS INPUT.
   7)MAYBE.
   8)LIKEWISE.
   9)DONE,".
  10)MAYBE, BUT DONT YOU LOVE THE SYMBOL WALLPC.
   11)TAB IS A FUNCTION NAME ",CONFUSING.
   12)NO.NO.NO.
   13)
```

NO, WELL MAYBE.

14) SOUNDS INTERESTING.

15) TOO MUCH USER INTERACTION ",MAYBE.

16)NO. WE COULD A NEW ONE IN THOUGH.

17)MAYBE.

18)MAYBE.

19)YOU ARE RIGHT, OF COURSE. ON THE OTHER HAND WHY DONT WE WAIT FOR LISP 2 TO APPEAR ON THE SCENE.

20)I THINK THIS IS BEST OF ALL. I HAVE THOUGHT ABOUT THIS IMPLEMENTATION, BUT HAVE COME TO NO CONCLUSIONS.

21)PROBABLY YOU ARE RIGHT. BUT IS IT WORTH THE EFFORT CONSIDERING ITS LIMITED USE.

22)I AM NOT SURE. I BELIEVE THAT MANY WILL ARGUE YOUR POINT.

23)MAYBE.

24)NO. ERRORE"S OF THAT SORT ARE SO INFREQUENT[THEY HAVNT OCCURRED TO ME AS YET AND I AM THE BIGGEST USER OF READLAP] THAT THEY DO NOT PAY OFF IN THE CODE NECESSARY TO GET AROUND THEM.

CONSIDER GETTING SALTZER TO REWRITE EDL FOR US TO GIVE US A PARENTHESES COUNTING COMMAND FOR WHICH WE SPECIFY A STARTING POINT AND RECEIVE THE LINE AT WHICH PARENS COUNT OUT(WRITTEN AS FAR AS THE LAST RIGHT PAREN TO MATCH).

WE ALSO NEED A CHANGE IN PRINT[REALLY RQUEST PRINT) TO PUT IN A BLANK IN COLUMN 1 OF LINE-MARKED FILES WITH SECONDARY NAME LISP. I WOULD RATHER NOT MAKE THE PRINT AND PUNCH ROUTINES PLACE ?NOT MAKE THE PUC"NCH ROUTINE PLACE A BLANK IN COLUMN 1.

JOEL

```
EDIT
FILE
R 7.166+11.583
```

EDL

```
       ALLOTTED   USED
 1        15        .8
 2        30      26.3
 3        45      39.3
 4        30      30.0
 5         0       0.

        STORAGE
DEVICE   QUOTA    USED
DRUM         0       0
DISK       200     198
TAPE         0       0

 R 1.666+.416

 edl rquest file?edl output rquest
W 049.6

n
 n 99INPUT FILE HAS IMPROPER FORMAT.

N
  QUIT,
R 1.700+.816

 print output rquest
W 050.3

 OUTPUT RQUEST       12/21  0050.3

  T0312  30477PUNCH LOMACS        FAP  EXPTEE        FAP  ZERRRF        FAP
 T0312  30470        ATOMS        FAP  FENFEN        ABS
 T0312  3047 PRINT FENFEN        BCD
R .633+.466

 delete fenfen symtb
W 051.1
R .566+.383

 runcom moses
W 051.3
 MOSES STARTED
 FILE   DEAR MOSES NOT FOUND.

dear joel..
     runoff dearjm for a thrill.  in brief,
          (1)  i have completely reassembled and reorganized
lisp, using macros and generally jazzing up the code.  using
startl14220500024000,weget6934wordsofbps.
          (2)  some features are altogether new.  not much though.
          (3)  it hangs on a lack of lisp2 lisp.  you bastard.
          (4)  i have 7-punched it off for track reasons.  keep in
touch.


t
 n 99

 DEAR JOEL..
     RUNOFF DEARJM FOR A THRILL.  IN BRIEF,
          (1)  I HAVE COMPLETELY REASSEMBLED AND REORGANIZED
LISP, USING MACROS AND GENERALLY JAZZING UP THE CODE.  USING
START L1 4220 5000 24000 , WE GET 6934 WORDS OF BPS.
          (2)  SOME FEATURES ARE ALTOGETHER NEW.  NOT MUCH THOUGH.
          (3)  IT HANGS ON A LACK OF LISP2 LISP.  YOU BASTARD.
          (4)  I HAVE 7-PUNCHED IT OFF FOR TRACK REASONS.  KEEP IN  TOUCH
```

```
W 1319.0
~~R~~ .866+.466

SDUMP FENFN1 1701 3
W 1320.6
01701  PZE 000000070437  PZE 000000070436  PZE 000000064424
R 1.083+.466

SPATCH FENFN1 17010" 0 70440 0 70437
W 1322.1
R 1.583+.516

SDUMP 170"""D"FENFN1 1701 3
W 1322.5
01701  PZE 000000070440  PZE 000000070437  PZE 000000064424
R .416+.333

SDUMP FENFN1 76651 2
W 1324.3
176651 SAVED NØT FØUND.
R 1.316+.616

SDUMP FENFN1 76651 2
W 1324.8
76651  PZE 000000006167  PZE 000000000000
R 1.116+.366

SPATCH FENFN1 17""76651 0 6166
W 1325.7
R 1.100+.550

SDUMP FENFN1 76650 2
W 1326.1
76650  TRA 002000076636  PZE 000000006166
R 1.700+.833

RUNCØM MØSES
W 1326.6
MØSES STARTED
 FILE   DEAR MØSES NØT FØUND.
INPUT
DEAR JØEL..
      NØW I NØTICE THAT THE VERY FIRST LINE ØF L1 LISP
IS AN ABSØLUTE PATCH.  GØØD GØD.  I THØUGHT THAT THE
WHØLE PØINT ØF THIS EXERCISE WAS TØ PUT T"AN END TØ THAT
SØRT ØF CRAP.
      BY THE WAY, YØUR PATCH WØN'T WØRK ØN MY SYSTEM,
SINCE I'VE WIPED NILLØC, MØVING LISTNW DØWN TØ 00077.
WHY NØT MAKE LISTNW A SYM ØR SØMETHING LIKE THAT (Q)
                                         F

EDIT
FILE
 MØSES HAS BEEN RUN
R 7.550+5.166

DL L1 LISP
W 1331.9
EDIT
C .. . 999
```

Dear Joel'":?Dear Joel:

(1) I looked into getting the disk editor changed to shift LM files with class name LISP over one column (i.e., no carriage control). My friends are no longer doing the disk editor; you (who are a tech employee, unlike me) should get in touch with Wm. Bjorstedt (sp.?) at the Comp Ctr. He will likely be able to do it without much delay.

(2) The name of my little darling is EENEEN. Each of the remaining numbered sections of this memo is devoted to an insert.

(3) IOMACS includes the old z1. Three important macros (TSSX, CALLIO, and OPEN) simplify calls to the supervisor, and to the file system in particular. The constant pool has been virtually wiped out, because (a) half the constants in it weren't used for anything, and (b) most constants are now used as literals (the literals live just below toprog). The disk error procedure is as follows: The message 'DISK ERROR. SAVE AND PRNTER.' is printed. Then, LISP goes dormant in such a way that start causes the io call to be retried. After consulting with Martin, I definitively removed the ESTATE from APEND.

(4) IOCNGE includes NOFILE, a subroutine which gets two arguments (a file-name). NOFILE returns *T* if there is no such file, NIL otherwise.

(5) ZERRRE includes ZELAP. DCT is a TRA 1,4. EPT is taken from a SHARE EPT.

(6) ZERR1 is yours.

(7) SETCNE incudes SAVEZ. TPC is now initialized by SETUP, not by the assembler. END9, END10, ..., END16 of SAVE were never used. They are out.

(8) GCOLEE is trivially improved (literals).

(9) TEMLIE. The EVLIS stuff, EVCDR and EAG11 of EVAL, and PIND of the character routines were never used. They are out.

(10) CONSZ is yours.

(11) BLCKRE includes DECONZ. Dull.

(12) EQUALZ is yours.

(13) PRNTZE has no WALLPC.

(14) PUNCHE is trivially now.

(15) FLOUME is trivially now.

(16) REDZ1E does half the work of leaving 'EOF' just after the end of a file being read.

(17) GTGCDE includes READZ. The rest of that eof stuff is here, at GTEOF, which was unreached before.

(18) SUBSTE is trivial. It does <u>not</u> include McCarthy's SUBLIS, which should be collected from HArt.

(19) APENDE. NOTS routine is out; the NOT atom runs to NULL. There is now an atom going to RPLACW.

(20) EVQTZF. EVALQUOTE treats EOF like STOP.

(21) CARXZF is trivial.

(22) ADV1FF includes ADVANC, MKNOZ.. ERROR1 is out. All sorts of tape-error shit is out. LOGOR uses to tell the PDL that it was LOGAND, and vice versa. That has been fixed.

(23) ARRAY1 is yours.

(24) FIXVAL is yours.

(25) ARAY2F is trivial.

(26) ARTH1F is trivial.

(28) EXPTFF is derived from the FORTRAN library routines. It includes natural-log and e-to-the-x entries. It is fast and short.

(28) ARTHMF is trivial.

(29) APPL1F. EVCON returns NIL when it runs out of clauses. I like this, but I will not weep if you take it out. At the end of the compiler service section, the literals come in and this insert ends.

(30) ATOMS is a radical recoding. All of the permanent list structure is generated in 8 tracks of FAP. After three pages of macro-definitions, the property lists of the non-alphabetic objects are laid down at the rate of one card per atom. This includes arranging for each atom that it be on the OBLIST. It is really pretty neat.The macros could haveeen cuter, but I had one set of supercuties that assembled at about 20 atoms a minute. The present macros are reallyquite straightforward, and they clean up the code something fierce. There is a separate set of macros for the letters.

There are two new alphabetic APVALs, COLON and PRIME. There are two new numerical APVALs, FSLEFT and FWLEFT. These point into the free-counters of the garbage collector. FILEGONE is a function connected to NOFILE (see (4) above). RPLACW is in. NUMVAL is in for the convenience of LAPpers. LOGSUB takes the natural log of a floating-point number in the AC, and it leaves the result in the AC. <u>It does not operate on LISP objects</u>. Nevertheless, a function taking LISP objects can easily be LAPped from LOGSUB. EXPSUB is similar, with e-to-the-x.

(31) SYNS includes ZSETUP. SYNS is very small, since all that PJ1 syn -)PJ1 crap is gone. In SETUP1, I replaced SUB =20 ADD =1 by SUB =19. This can't be a sleeper, can it, one of those hidden-repercussion jobs? Remember that SETUP sets $TPG.

(32) CNSFWL is yours.

(33) CNSF1 is yours.

(34) LAPZFF. There is some la-de-da at the end to make sure that the literals and RMT sequences come out as they should.

Dear Bob,

I have been unable to link to a file with first name FENFEN. Please explain the magic procedure.

There is a bug in the compiler which I fixed before I left. Since I was cramped for space I left LISP2 in temp mode and it disappeared. The LISP2 which is available right now is the old file which was available all along as LISP2P LISP.

The locations 77 and 100 octal are reserved for the compiler. Your comment with respect to them is accurate, but I am not sure the effort is worth it. Is it possible to delete the SYSIND location and all references to it(Q).

I S shall try to comment on your points individually.

3) Not bad. It is unfortunate that there are only few cases where another chance will be helpful. Nontheless this is better than the way I had it.

4) Good.

5)Good.

7) Is there no way to get to the other entrances(Q).

11)No good. I would lie to have DECONZ separate so that it can be easily removed.

17) EOF idea is good.

22) You appear to be right about the error. I think the only affect it would have had would have been on a trace or error.

28)Good.

29)All right. I am inclined to agree with you on this score, after all.

30)Excellent. I like the APVALs especially.

31)No it is not a sleeper. You will finf that the system has no constant 19 and I was too lazy to put one in.

FWC
FSC

I have looked at the error changes, but have not done anything about them. Right now I feel pretty tired of the whole thing and would like to get it to bed. I am willing to make your system into the new MAYBE for a testing period of about a month. If no bugs occur we can finish the memo and make it the new system. I am reluctant to leave the system hin its present condition for much longer.

Joel

LISTF , 12/5/65

```
ctest1 (smal)
W 326.9

     43 FILES    251 RECORDS
  NAME1   NAME2 MOD NOREC    USED
ADV1FF     FAP 104       4  01/05/66
APENDF     FAP 104       4
APPL1F     FAP 104      15
ARAY1F     FAP 104       2
ARAY2F     FAP 104       3
ARTH1F     FAP 104       3
ARTHMF     FAP 104       7
 ATOMS     FAP 104       8
BLCKF2     FAP 104       2
CARYZF     FAP 104       5
CNSFUF     FAP 104       2
CONSZF     FAP 104       2
DECONF     FAP 104       2
EQULZF     FAP 104       4
EVQTZF     FAP 104       3
EXPTFF     FAP 104       3
FENFEN     ABS 104      27
FENFEN     FAP 104       1
FENFEN   SAVED 104      77
FENFN1   SAVED 001      77
FENFN2   SAVED 001      77
FIXVLF     FAP 104       1
FLONMF     FAP 104       3
GCOLEF     FAP 104       6
GTGCDF     FAP 104       2
    IO    LISP 000       6
LOCNGF     FAP 104       3
LOMACS     FAP 104       3
LAPZFF     FAP 104       6
  MATH    LISP 000       8  12/21/65
 MOSES     BCD 000       1  01/05/66
OLDDIF    LISP 000       4
PERMIT    FILE 120       1
PRNTZF     FAP 104       5
PUNCHF     FAP 104       2
REDZ1F     FAP 104       4
SETCHF     FAP 104       2
  SIMP    LISP 000       5
SUBSTF     FAP 104       7
  SYMS     FAP 104       3
TEMLIF     FAP 104       2
UTILTY    LISP 000       1
ZERRRF     FAP 104       2

      53 LINKS
  NAME1   NAME2 MOD PROBN. PROGN. LNAME1 LNAME2
   ADV1    FAP 044  T0302   2517
 ADVANC    FAP 044
 APPEND    FAP 044
 APPLYZ    FAP 044
 ARRAY1    FAP 044
 ARRAY2    FAP 044
  ARTH1    FAP 044
 ARTHMT    FAP 044
 BLOCKR    FAP 044
  CARXZ    FAP 044
  CLOCK   LISP 044
  CNSF1    FAP 044
 CNSFUL    FAP 044
```

```
  CONSZ     FAP 044
 DECONZ     FAP 044
 ENDEND     FAP 044
 EQUALZ     FAP 044
 EVOTEZ     FAP 044
 EXCISE    LISP 044
   EXPT     FAP 044
 FIXVAL     FAP 044
 FLONAM     FAP 044
 GCOLEC     FAP 044
  GTCCD     FAP 044
  IOCNO     FAP 044
      .   SAVED 104   M1416  CMFL04
     L1    LISP 044   T0302    2517
   LAPZ     FAP 044
  LISP2    LISP 044
   LIST   SAVED 104   M1416  CMFL04
 LISTEN    LISP 044   T0302    2517
  LITUP   SAVED 044
 MASTER     FAP 044
  MAYBE   SAVED 044
  MKNOZ     FAP 044
 PRINTZ     FAP 044
  PUNCH     FAP 044
  READZ     FAP 044
 READZ1     FAP 044
 REMOBB    LISP 044
  SAVEZ     FAP 044
 SETCON     FAP 044
 SETMEM    LISP 044
 SQUASH   SAVED 104   M1416  CMFL04
  SUBST     FAP 044   T0302    2517
 TEMLIS     FAP 044
 UNMAST     BCD 044
    WHO   SAVED 104   M1416  CMFL04
     Z1     FAP 044   T0302    2517
  ZERR1     FAP 044
 ZERROR     FAP 044
  ZFLAP     FAP 044
 ZSETUP     FAP 044

R 3.133+.933
```

```
just loaded it?r 23.133+.650

save fenfn1 t
W 213.0
R 3.150+.716

start 11 4220 5000 24000
W 213.4
 6939
 VALUE
 NIL
 VALUE
 0
 VALUE
 NIL
 VALUE
 NIL
 VALUE
 DEF1
 VALUE
 DEFLIS1
 VALUE
 (DEFLIST DEFINE CONC1 CSET PROG2 SELECT2)
 VALUE
 (CONC CSETQ SELECT)
 VALUE
 (FLAG TRACE)
 VALUE
 381
 VALUE
 (EFFACE REMFLAG UNTRACE OPDEFINE)
 VALUE  OPDEFINE
 NIL
 VALUE
 ((A . 15566Q))
 VALUE
 ((A . 15572Q))
 VALUE
 ((A . 15576Q))
 VALUE
 ((A . 15602Q))
 VALUE
 ((A . 15606Q))
 VALUE
 ((A . 15612Q))
 VALUE
 ((A . 15616Q))
 VALUE
 ((A . 15622Q))
 VALUE
 ((A . 15626Q))
 VALUE
 ((A . 15632Q))
 VALUE
 ((A . 15636Q))
 VALUE
 ((A . 15642Q))
 VALUE
 ((A . 15646Q))
 VALUE
 ((A . 15652Q))
 VALUE
 ((A . 15656Q))
```

```
V
 ALUE
((A . 15662Q))
VALUE
((X . 15671Q))
VALUE
((X . 15706Q) (A . 15674Q))
VALUE
((X . 1572Q1) (A . 15711Q))
VALUE
((X . 15737Q) (L . 15733Q))
VALUE
((Q . 15771Q) (U . 1577Q1) (U . 15765Q) (B . 15763Q) (X . 15761Q)
(Y . 15756Q) (Z . 15755Q) (C . 15747Q))
VALUE
((Y . 16002Q) (X . 1603))
VALUE
(APVALSYM)
VALUE  APVALSYM (A)
OBLIST
VALUE
((Y . 16027Q) (X . 16025Q))
VALUE (B)
(READLAP)
VALUE CONS SINCE TOP
1024
VALUE FILE-SEEK
NIL
VALUE  READLAP
NIL
VALUE
1298   CONS' (C)
VALUE  EXCISABLE
T
VALUE   READLAP
NIL
VALUE  UNCOMMON
(FARG)
VALUE  UNSPECIAL
NIL
VALUE  EXCISABLE
NIL
COMVAL        COMVAL
COMPLY        COMPLY
COMLIS        COMLIS
LAC   LAC
STORE         STORE
COMPROG       COMPROG
COMPACT       COMPACT         REMOBED
COMBOOL       COMBOOL
COMCOND       COMCOND
CEQ   CEQ
CALL  CALL
ATTACH        ATTACH
LOCATE        LOCATE
DELETE1       DELETE1
PA1   PA1
PA4   PA4
PA3   PA3
PA5   PA5
PA6   PA6
PA7   PA7
PA11  PA11
PA14  PA14
PA12  PA12
PI1   PI1
```

```
 PI2  PI2
 PI3  PI3
 PALAM       PALAM
 PAFORM      PAFORM
 PAIRMAP     PAIRMAP
 PA8  PA8
 PA9  PA9
 PA2  PA2
 VALUE  REMOB
 NIL
 VALUE
 NIL
 VALUE
 ((APPEND PRINT2) (CAADR UNPACK ZEROP) NIL (COM1 PAIR) (FSLEFT
 -) (COMMON ERRORSET READ) (LISP) (TO CURCHAR) (UNDEFINED CDDADR
 CLA COUNT TAB J) (STR LIST) (PHASE2 FN FLAG) (STZ APVAL FSUBR)     OBLIST
 (RETURN K) (AQ0215 GREATERP NOT) (BQ0003 AQ0406 PRIME SLASH)
 (AQ0825 REMOBR CADDAR COLON NUMOB) (LOGOR L) (EXPT SEARCH)
 (CDR DOLLAR ERROR) (EQUAL EXCISABLE RPLACA) (EXP M) (EXPLODE
 FLOAT MINUS) (CDAAAR GO NUMVAL RPLACD $EOF$) (EXCISER EXCISE)
 (XCA HAS GCCAG N) (CDADR SPECIAL TRACE) (LXD EVAL LAP) (TXH
 COMPRINT EQP FILEDELETE) (AQ1271 DEF1 O) (COMPILE TIX) (SUB
 TNZ CR EQ MAP OCTAL) (AQ1464 ATTRIB OHEP) (AQ1294 CAADDR CAAAR
 MAX QUOTE P) (COPY *T*) (FIXP NIL) (COM2 LPAR PLUSS) (AQ1279
 ARRAY EULEFT PRIN1 STARTREAD Q) (REMOB DASH) (SXA CPJ) (NAME
 LISP2 CDDDAR) (FIX FUNCTION R) (PERIOD REMAINDER) (PXD FILESEEK)
 (TXI EOR LOGSUB) (PAX MAPCON) (SETC PUNCHP RDFLX) (AQ0392 UNCOMMON
 CADAAR) (=) ($) (OB CAAR CADDR) (PROGITER) (EVLIS ') (GIVEN
 TYTAB *) (NUMBERP) (GOLIST STD DIFFERENCE DIVIDE) (MEMBER *MN
 LEFTSHIFT UNCOUNT) (CDADDR UNTRACE CONC1 APPLY CDAAR RPLACW
 SET $IL32$
) (TRA PUNCHT $IL15$) (COMRSTR CHARCOUNT LAMBDA
 RECIP $IL17$ $IL36$) (REMFLAG RPAR $IL57$ $IL76$) (LESSP) (
 LOGAND PRINT) (CSET MIN SAVBK SETBK) (CAAADR CONS) (AND PACK
 STAR) NIL (DELETEL STOMAP SPECRSTR LDQ BPSREMAINDER) (+) (PDX
 DEFLIST DEFLIS1 CLEARBUFF *LIST  ) (AXT NCONC PUNCH) (CDDAAR
 ADVANCE LITER) (ENDREAD A) (FORM READLAP /) (SELECT SELECT2
 CDAR CDDDR) (MODE) (GET RECLAIM SPEAK B) (COMP AQ0490 TXL DIGIT
 S) (AQ0852 RETN) (EXPR SUB1) (BACKTRACE C) (UNDECLARED CADDDR
 CADAR T) (PASSONE REVERSE OBLIST PROG) (SETQ STOP) (D) (FUNC
 COMMA EQSIGN U) (PROP) (SXD) (CDAADR CONC E) (STO SUBR V) (
 LISTEN TIMES) NIL (LISTING PRO F) (LOGXOR W) (SPECBIND $ALIST)
 (SWITCH CAADAR AS GETBCD) (AQ1474 OPDEFINE G) (AQ1209 APVALED
 FUNARG MKNO SYM X) (CADR) (LENGTH *N) (TERPRI H) (AC INTERN
 Y) NIL (TSX) (ADD1 GENSYM I) (CDDDDR CDDAR Z) (UNSPECIAL LXA
 PLUS) (DEFINE OR) (AQ0114 EFFACE FILEENDRD SASSOC) (ATOM PNAME
        ) (ARGS AQ0572 FILEGONE MKNAM) (AQ0307 AQ0193 BLANK LABEL)
 (AQ0764 CADADR *MOVE .) (AQ0974 AQ0936 ,) NIL (CSETQ PROG2)
 (G2 )) (NULL SUBST () (AQ0998) (CS CDADAR FEXPR MAPLIST SUBLIS)
 (COND QUOTIENT REMPROP :) (STQ EOF $IL52$) (COMBIND PXA CAR
 CDDR $IL16$) (TZE $IL37$ $IL56$ $EOR$ $IL75$) (FARG TLQ FILE1
 $IL77$) (FLOATP MINUSP) (APVALSYM CAAAAR TNX EXPSUR OPCHAR)
 (AQ1174 *RETURN))
 VALUE
 9349   CONS'S
R 46.550+2.183

save fenfn?save henfn2 t
W 220.3
R 3.133+.683

restor fenfn1
W 221.2
R .016+3.633

save henfn1 t
```

JOEL:

(1) RE LITER: SEE p 154 IN ADDITION TO THE ONES (88, 90, 92, 93, 149) YOU HAVE

(2) RE SUBLIS: ACCORDING TO THE MANUAL, THE CAR'S ON THE ALIST ARG NEED NOT BE ATOMS. I THINK THAT MOST PEOPLE HAVE ONLY ATOMS AS CARS, SO I FAVOR A VERSION OF SUBLIS WHOSE DESIGN FEATURES ARE

(A) SPEED WHEN CARS ARE ATOMS

(B) COMPACTNESS

(C) SPEED ON NON-ATOMIC CARS

IN THAT ORDER. TRY

~~speci[illegible]~~ NEW

NEW

~~sublis[a;s] = prog[[g;aa;y];~~

~~g = gensym[];~~

~~aa = NIL;~~

~~map[a;λ[[l];~~ NEW

~~[atom[caar[x]] → rplacd[caar[l]; cons[g; cons[cdar[l]; cdaar[l]]]];~~

~~T → aa = cons[car[l]; aa]]]];~~

NEW

~~y = sublis1[map[aa~~

```
SPECIAL ((G, SS))

DEFINE ((

(SUBLIS (LAMBDA (A, S) (PRØG (G, Y, SS)
        (SETQ SS S)
        (SETQ G (GENSYM))
        (MAP A (FUNCTIØN (LAMBDA (L) (CØND
              ( (ATØM (CAAR L)) (RPLACD (CAAR L)
                         (CØNS G (CØNS (CDAR L) (CDAAR L))) ))
              (   T    (SETQ SS (SUBST (CDAR L) (CAAR L) SS)))
                                                  ))))
       (SETQ Y (SUBLIS1 SS))
       (MAP A (FUNCTIØN (LAMBDA (L) (AND
              (ATØM (CAAR L)) (RPLACD (CAAR L) (CDDDR (CAAR L)))
                                                    ))))
       (RETURN Y) )))
(SUBLIS1   AS   BEFORE
))
UNSPECIAL ((G, SS))
```

(3) RE LITER: THE L1 CODE IS

```
FLAG ((A, B, ..., Z) LITER)
EVAL ((FLAG (LIST PLUSS DASH SLASH STAR EQSIGN) ØPCHAR)
                    NIL)
```

```
DEFINE((

(LITER (LAMBDA (Ø) (AND (GET Ø (QUØTE LITER)))))
(ØPCHAR (LAMBDA (Ø) (AND (GET Ø (QUØTE ØPCHAR)))))

))
```

(4) BESIDES LITER & OPCHAR, THE FOLLOWING SUBRS COULD BE MOVED BACK TO EXPRS

(A) SEARCH 35
(B) SUBST 70 COULD USE SUBLIS TRICK TO AVOID SUPERFLUOUS CONSING
(C) MAP 73
(D) MAPCON 73
(E) REMPRP 74
(F) ATTRIB 18

F

```
**********************************************************************************************************************
T0312          3047      FENFEN          FAP FOR   T0312          3047        01/05       0
      INSERT   $IOMACS       TSSX,TSSX1,CALLIO,OPEN                               00010
      INSERT   $IOCNGF       FILE ROUTINES                                        00020
      INSERT   $ZERRRF                                                            00030
      INSERT   $SETCNF                                                            00040
      INSERT   $GCOLEF                                                            00050
      INSERT   $TEMLIF                                                            00060
      INSERT   $CONSZF                                                            00070
      INSERT   $BLCKF2 → BLCKMY, SEE p 29                                         00080
      INSERT   DECONF                                                             00090
      INSERT   $EQULZF                                                            00100
      INSERT   $PRNTZF                                                            00110
      INSERT   $PUNCHF                                                            00120
      INSERT   $FLONMF                                                            00130
      INSERT   $REDZ1F                                                            00140
      INSERT   $GTGCDF                                                            00150
      INSERT   $SUBSTF                                                            00160
      INSERT   $APENDF                                                            00170
      INSERT   $EVQTZF                                                            00180
      INSERT   $CARXZF                                                            00190
      INSERT   $ADV1FE → ADVIMY, SEE pp 94-96, IF COMP DOESN'T USE LOG, LSHIFT    00200
      INSERT   ~~$ARAY1F~~ OUT                                                    00210
      INSERT   $FIXVLF                                                            00220
      INSERT   $ARAY2F  ARYMAK  BSS  0                                            00230
      INSERT   $ARTH1F                                                            00240
      INSERT   $EXPTFE                                                            00250
      INSERT   $ARTHMF  ARTHMY, SEE p 119                                         00260
      INSERT   $APPL1F                                                            00270
      INSERT   $ATOMS                                                             00280
      INSERT   $SYNS                                                              00290
      INSERT   $CNSFWF                                                            00300
      INSERT   $LAPZFE                                                            00310
```