

See [SYSDOC. (CODE)] WORKNOTES.LAY

[What must we do?]

Copyright © 3L Ltd., 1987-1992

Fortran items in I-code

Peter S. Robertson
5th August 1986

The new items in I-code to handle FORTRAN are:

- ✓ ~~Init, Define~~
- ✓ Area-Size
- ✓ Convert
- ✓ Define (T=15)
- ✓ Define-At
- ✓ Fixup
- ✓ Position
- ✓ Promote
- * ~~G~~move

First, the easy ones

Convert <n>

This performs the fortran-defined type conversion on TOS. ?

Promote <n>

This removes the n'th item from the stack and pushed it back on the top of the stack. E.g. PROMOTE 1 is a no-op and PROMOTE 2 is SWOP.

The rest are more complicated so I will give a suggestion as to a possible implementation.

Define (T=15)

This defines a general (static) area. Fortran will invent some areas and the use DEFINE-AT to locate objects within them. These areas can be made external in the usual way to get the effect of COMMON. Note that it has been suggested that all common areas are treated as specs; if a common area is initialised then in addition to the spec the module will have an external definition for it.

My plan was that the definition of a general area would simply allocate space for a pointer in GLA.

Define-At

This defines a descriptor in the same way as DEFINE but specifies a general area and a byte displacement into that area where the object is to be located. This simply needs to mark the descriptor as being an area-based object and STACK VAR can convert references

Warning: no inputs into $\emptyset$