To: Distribution

From: Steve Webber

Subject: Proposed new form of Link

Date: 1/19/76

This MTB proposes adding a new form of link, called a *system link, to the system and that this link be used for (at least) PL/I external variables with no embedded dollar signs and FORTRAN common blocks. The new links would reference a single component name (i.e., no dollar signs and in particular, no reference name) and the system linker would know how to resolve the link. In conjunction with this proposal, it is also proposed that type 6 (create if not found) links be treated the same as type 4 links, i.e. complain if not found. This latter proposal is incompatible and must be modified slightly for a smoother transition. In particular, the type 6 links that specify "stat_" as the reference name will be mapped into *system links and the type 6 links that have a reference name ending in ".com" will also be mapped into *system links. (This mapping will be done only by the system linker, not by the binder.)

There are two other (possibly common) uses of type 6 links that must be considered. These are:

1. use of PL/I external variables of the form a$b, and

2. use of common blocks with a dollar sign in the name.

One reason for wanting PL/I external variables of the form a$b to be created dynamically is that stat_ is not large enough to handle more than 256K worth of variables. This restriction would be removed with *system links and hence this reason for type 6 links goes away. Another reason for wanting PL/I external variables of the form a$b is to partition such variables into classes via the reference names. That is, a user may put a class of variables in the a$ pool, another class in the foo$ pool, etc. This can now be done (albeit incompatibly) with something of the form "a_" or "foo_" as a prefix to the variable name.

The use of common blocks with a dollar sign in the name allows users to have permanent common blocks. This will continue to work although, as I understand it, this is nonstandard FORTRAN.

Details

The *system link would be implemented as a type 5 (*|<ext>+exp,m) link with a new class. The currently defined classes are:

0 relative to text
1 relative to link
2 relative to symbol
3 <special>
4 relative to static

It is proposed that 5 be defined as relative to the outside world, the system. The linker would keep a table of all <ext> names found in *system links and would allocate the storage for the variables in system storage (the combined linkage regions). This provides for the possibility of a more efficient search mechanism than the current walk of a linear list of definitions. It also uses many fewer segments in that FORTRAN common blocks will in general require 0 (as opposed to today's 2) segments. Further, it makes it possible for PL/I programs to easily reference FORTRAN common blocks and vice-versa. This is because the new FORTRAN compiler would not add the ".com" suffix to common block names.

There are some other points to be noted. First, the *system links will not be able to have a trap-before-link option. This is because the the trap field (see below) will be used to hold initialization information for the storage associated with the link. Another note is that FORTRAN blank common, currently specified via a link of the form <b_.com>|0, must be special cased. This is because the size of blank common can not be known (in Multics) when blank common is first referenced. Hence, the linker will allocate an entire segment for links of this form.

The use of *system links gives the system a useful check on the allocation and use of PL/I external variables. That is, the size of the PL/1 variables can be checked (when the links are snapped) against the allocated size. This is not possible with today's type 6 links because the size is not retained.

Since the linker will have its own manager of *system links, it will be an easy task to write a program that lists the locations (and possibly values) of all *system variables. This would be convenient for debugging.

Mappings

The following compatibility mappings are proposed:

OLD	NEW
<stat_>¦[any-name]	*system¦[any-name]
<any-name.com>¦0	*system¦[any-name]
<b_.com>¦0	*system¦special-for-blank-common

The attached diagram shows the actual format for the proposed new links.

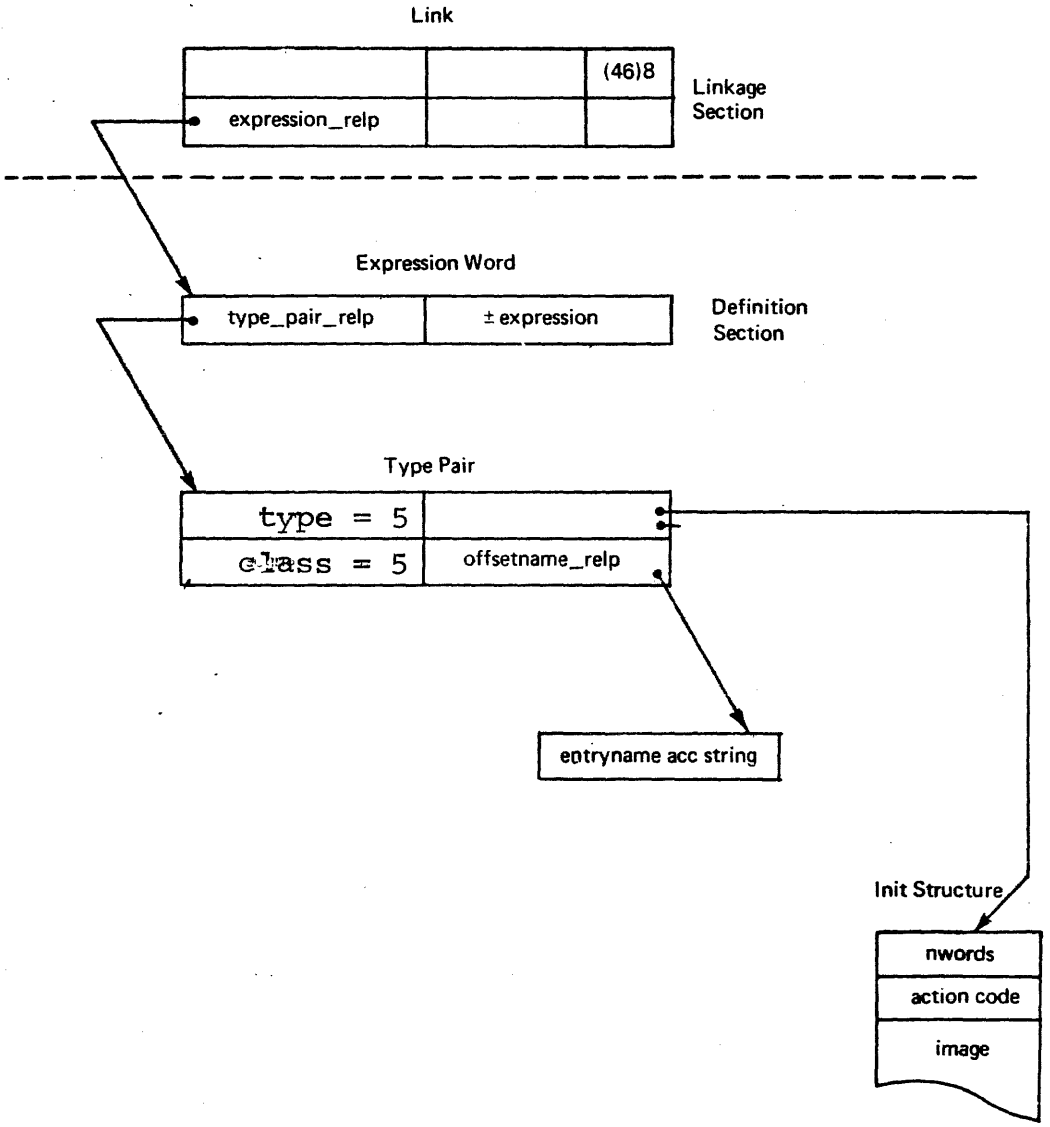

Figure 1-2. Structure of a Link
*system only