

## Identification

Standard Call Brackets  
M. A. Padlipsky

## References

This section assumes a thorough knowledge of BD.9.00, and a reading knowledge of BD.9.01-03.

## Purpose

A common problem in sub-system implementation lies in the fact that it is not easy for the sub-system (which nominally resides in ring 32) to interpose itself between its user procedures (which nominally reside in ring 33) and Multics system procedures. That is, a ring 33 procedure could, unless somehow prevented from so doing, call into ring 1 or even ring 0 without the sub-system it is executing under being aware of the call, and possibly interfere disastrously with the workings of the sub-system. For the sub-system to prevent this sort of thing requires herculean labors. The system, however, can achieve the desired end in a rather elegant fashion, simply by adhering to the following convention.

## Convention

The highest generally allowable ring number in the call bracket of a system routine is 48.

## Discussion

If no system routine can be called from a ring "outside" of ring 48, procedures executing under a (ring 32 through ring 47) subsystem simply cannot sneak around their sub-system. They must, instead, call upon the sub-system to perform the calls for them. By means of a transfer vector-like arrangement of some sort, the sub-system is able to exercise whatever degree of control it needs to over inward calls, at the minimum cost of an extra call and return. It can, of course, do argument validation itself, when necessary. Subsystems which do not need the ability to interpose themselves between the system and their subordinate procedures can place the subordinate procedures in rings 33 - 48; subsystems which do need the ability can place theirs in ring 49 or above.

Exceptions

Library routines and compiler run-time routines are not subject to the convention, as subjecting them to the convention would introduce needless overhead. However, library routines and compiler run-time routines should have a maximum call bracket limit of 56. This allows rings 57-63 to be used for "restricted users" in the old CTSS sense of the term, and is also, for example, of use to instructors who do not want their classes to employ the library or run-time routines freely (if at all).

The convention is not intended to suggest that there are no conceivable exceptions to the general rule other than the one just stated. There may well be system routines which are innocuously-callable from any ring, but the convention should be applied in general and only violated after careful thought. The basic point is that it is far easier for a sub-system to pass on calls than it is for the sub-system to intercept calls, so calls on which the system programmer is sure that interception will not be needed can be allowed by means of an upper limit on the call bracket greater than 48, but less than 57.