Identification

A programmer's guide to the efficient use of EPL.

James F. Gimpel

Purpose

EPL contains a relatively rich assortment of programming tools, often leading to a variety of ways of encoding a given problem. This section is intended to provide a basis of choice when alternatives exist and to provide a ready source of information to the EPL programmer concerned with object code efficiency.

Discussion

Though one intent of writing systems in higher-level languages is to free the programmer from such picayune matters as code efficiency, recent developments have indicated that for the short run, at any rate, efficiency of object code cannot be ignored. For the long run there is a sometimes expressed, sometimes implied belief that, if programs are coded in the most reasonable and natural way at the source level, ultimately a compiler will exist which will provide reasonably efficient translations of these programs. To a large extent this is true; but there are certain features of the language which are inherently difficult to implement. Extensive use of these features, when alternative methods exist, should be avoided in most systems programming.

Scope

It would be unwise to base the discussion on the idiosynchrosies of any particular version of the compiler. Not only would the discussion be obsolete with the introduction of a new compiler, but the programs written using that advice would be sprinkled with obsolete glitches. Therefore, a somewhat broader view has been attempted. Some of the statements contained in this section carry an almost universal validity, i.e., they contain code-saving hints useful on virtually any reasonable PL/I compiler for any machine. Most, however, are Multics and/or 645 dependent and therefore have a somewhat more restricted applicability. But no material has been included which the author expects to be obsolete within a year and most of the material will be valid much longer than that.

Some of the statements contained in this section imply a version of the compiler which does not yet exist. Any such future modifications whose existence is implied are already in the design and/or check out phase and are expected to be available by mid-1967.

Contents