6. Implementing EGS4

The flow of control and data when a user-written program is using the EGS4 code is illustrated in Fig.1 (detailed information needed to write User Codes is given in Appendix 2 of SLAC-265)[1].

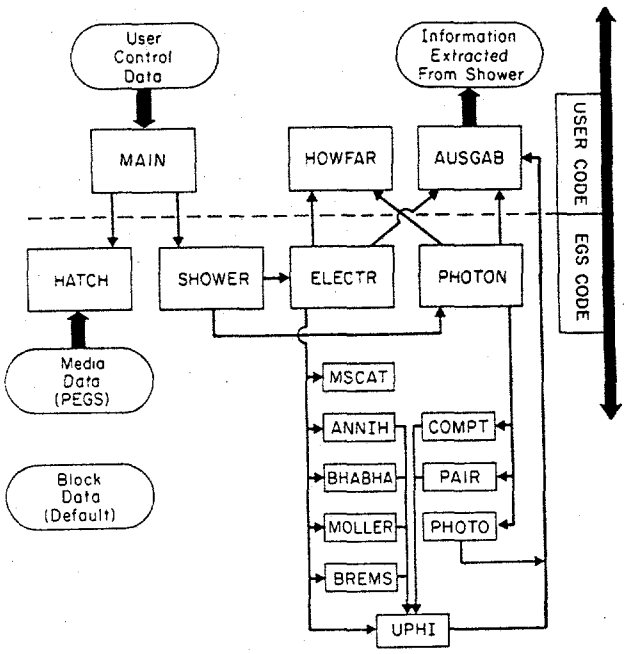

Figure 1. Flow control with user using EGS4.

The EGS4 code itself consists of two *user-callable* subroutines, HATCH and SHOWER, which in turn call the other subroutines in the EGS4 code, some of which call two *user-written* subroutines, HOWFAR and AUSGAB. The latter determine the geometry and output (scoring), respectively.

The user communicates with EGS4 by means of various COMMON variables. To use EGS4, the user must write a MAIN program and the subroutines HOWFAR and AUSGAB. In general, MAIN will perform any initialization needed for the geometry routine, HOWFAR, and set the values of certain EGS4 COMMON variables which specify such things as names of the media to be used, the desired cutoff energies, and the unit of distance (*e.g.*, centimeters, radiation lengths, etc.). MAIN then calls the HATCH subroutine, which "hatches" EGS by doing once-only initialization and by reading from the data sets prepared by PEGS for the materials requested.

With this initialization completed, MAIN may then call SHOWER when desired. Each call to SHOWER results in the generation of one EGS history. The arguments to SHOWER specify the parameters of the incident particle. Therefore, the user has the freedom to use any source distribution desired.

Also included as an argument is the *particle weight*, normally assumed to be unity. This allows for the implementation of *importance sampling* and other variance reduction techniques—*e.g.*, splitting, path-length biasing, Russian roulette, leading-particle biasing, *etc.*.