The Build Program

Build pulls together equations estimated with *G*7 or *G* into a model. It does so by writing a C++ program, HEART.CPP, which must then be compiled by a C++ compiler and linked with two object programs, RUN.OBJ and UTILITY.OBJ, to produce an executable file, RUN.EXE, which is model ready to run. Basic documentation will be found in *The Craft of Economic Modeling*.

The HEART.CPP program is now written for the Borland C++ compiler. We have had success using every Borland compiler from Turbo C++ 2.0 to Borland C++ 5.5, which is now available for free at www.borland.com/bcppbuilder/freecompiler. In case you have some other C++ compiler, the source code for the RUN.CPP and UTILITY.CPP files has been included, so that you can recompile them with your compiler. Various other changes may have to be made, in the make file or in headers, depending on your compiler.

The output of *Build* is:

- BWS.IND and BWS.BNK : A G data bank containing the historical values of all variables in the model.

- HEART.CPP: The C++ program to compute the model. Now written for the the Borland C++ compiler.

- RUN.LAG: A binary file necessary for the running of the model.

- RUN.NAM: The names and some information about each variable in the model. In front of each variable name are three numbers. The first is the sequential number of the variable, the second is the maximum number of lags with with it occurs, and the third is the number of times it is defined in the model. If this last is zero, the variable is exogenous. Watch out for variables defined more than once! While not necessarily an error, this condition can indicate that the same name has been used for more than one concept.

- RUN.GR : A file for graphing all the variables in the model. It also indicates whether the variable is exogenous or endogenous.

When *Build* starts, it reads the BUILD.CFG file for its basic configuration. Read this file and insure that it is appropriate for you computer's setup. Similarly, RUN.EXE, the model itself, reads RUN.CFG.