The data provided by the United States Bureau of Census in the form of census summary tape files can be of great use to planners and administrators and to researchers in many areas. However, access to this information by the "unsophisticated" computer user is difficult because of the large number of data elements and the many different record formats and sizes used to record the data. Attempts have been made to write computer programs that provide convenient access to the census tapes. In particular, MOD3 allows the user to refer to data elements by a set of data descriptors. The user can indicate, in a high-level command language, which files to access, which records to retrieve, and which data elements to display. The MOD3 program, via a table lookup, handles the mechanics of locating the data in the files.

The MOD3 program is distributed by National Data Use and Access Laboratories (DLULabs), Suite 900, 1601 North Kent Street, Rosslyn, Virginia 22209.

The following is an example of the use of the MOD3 program.

```
// JOB
// EXEC MOD3
// DOTAPE DD DSNAME=3RCHT,UNIT=2400,
// DISP=OLD,LABEL=(3.SL),VOL=SER=3001303
// CENSUS DD DSNAME=\'VAST.3\',UNIT=2400,
// VOL=SER=200818,DISP=OLD
// SYSIN DD *
START 03
RETR 01 G04001 EQ 0001
STOP 01 G04001 NE 0001
LIST 01 005001 COUNT: TOTAL MALE
LIST 02 005002 COUNT: TOTAL FEMALE
LIST 03 005003 COUNT: BLACK MALE
LIST 04 005004 COUNT: BLACK FEMALE
END
/*
```

The START command specifies to MOD3 that the Third Count census summary tapes are to be accessed as input.

The RETR (retrieve) command specifies which input records are to be selected. This is done by specifying a condition which must be met in order for the input record to be selected. In the example, the condition is

\[ G04001 = 0001 \]

G04001 is an example of a data descriptor. The data item corresponding to this descriptor is the census tract number. Effectively, the RETR command in the example specifies that any input record for which the census summary tract number equals 001 is to be selected.

In general, a sequence of RETR commands may be used. Each RETR command must have a number. RETR commands with the same number are connected with a logical AND, while RETR commands with different numbers are connected with a logical OR.

Thus, the following sequence:

```
RETR 01 condition 1
RETR 01 condition 2
RETR 02 condition 3
```

specifies that an input record be selected if condition 1 and condition 2 are both met or if condition 3 is met.

Conditions are specified in the following form:

```
data descriptor运算符value`
```

The following are the operations which may be specified as part of a condition and their meanings.

- **EQ** Equal to
- **NE** Not equal to
- **GE** Greater than or equal to
- **LE** Less than or equal to
- **GT** Greater than
- **LT** Less than
- **OR** Or equal to

The STOP command is similar in form to the RETR command. However, the STOP command specifies a condition which, when met, terminates the retrieval process. If the user is familiar with the sequence of the census summary tapes, he can use the STOP command to avoid the expense of searching the entire tape.

A LIST command, again via a data descriptor, specifies which data item is to be retrieved and displayed. The LIST command also has a number which determines the position, from left to right, of the data item in the printed output. Our version of the MOD3 program allows a maximum of 10 data items for display. The LIST command also allows the user to specify labeling information for each data item.

In summary, the MOD3 program provides a unified access to the census summary...
tapes in a high-level command language with a flexible record selection facility. A disadvantage of the MOD3 program is that, in general, it serves only to display in columnar form the retrieved information. There are limited facilities for statistical analyses of the information.

In order both to provide a convenient access to the census tapes, particularly to the analyst who is not computer oriented, and to facilitate statistical analysis of the retrieved information, we wrote a local SAS procedure called MOD3. The SAS/MOD3 procedure is modeled closely after the MOD3 program. The following is an example of the use of SAS/MOD3.

```sas
// JOB
// EXEC SAS
// ODYTAP DD DSN=PDRTAPE,UNIT=2400,
// DISP=OLD, LABEL=(3, SL), VOL SER=901393
// OCENSUST DD DSN='INAST3', UNIT=2400,
// VOL SER=900018, DISP=OLD
// SYSIN DD *
// PROC MOD3;
// START OS;
// RETR 01 004001 EN '0001';
// STOP 01 004001 '0001';
// LIST 01 005001 'TOTAL MALE' THALE;
// LIST 02 005002 'TOTAL FEMALE' TFEFAL;
// LIST 03 005003 'BLACK MALE' BMALE;
// LIST 04 005004 'BLACK FEMALE' BFEFAL;
// *
// SAS/MOD3 uses the same databases as does MOD3. It accepts high-level language commands, similar to MOD3's, except that SAS/MOD3's are free-format. He hoped to achieve a transfer of learning by users already familiar with MOD3, but alleviate the problem of wasted runs caused by violating fixed-format specifications.

The major differences in the command languages are the following. On the RETR or STOP statements in SAS/MOD3, values are specified as character literals. The LIST statement requires an additional specification. Since SAS/MOD3 produces a SAS dataset as output, it requires the user to specify a different SAS variable name at the end of each LIST statement.

The MOD3 procedure is fully integrated into SAS in the sense that it uses SAS's parsing facilities to scan the procedure statements for syntax and uses SAS's diagnostic facilities to report errors.

The MOD3 procedure flags as errors (1) invalid census summary tape identifiers, (2) invalid operations on RETR or STOP statements, (3) statement numbers that are not numeric, (4) specifications that are incomplete, and (5) duplicate uses of SAS variable names.

Some errors are detected "too late" to be flagged with a numbered error. For example, the MOD3 procedure checks to insure there is only one START statement and at least one LIST statement associated with the current use of the procedure. In addition, the MOD3 procedure insures that all data descriptors specified by the user appear in a file of data descriptors associated with the census summary tape being accessed. Exceptions are reported as errors and identified by their appearance (line number and column) in the procedure statements.

An example of these diagnostic messages follows:

```
1 PROC MOD3;
2 START OS;
3 RETR 01 004001 EN '0001';
4 RETR 02 008001 EN '0001';
5 LIST 01 00777 'Houses' SASH;
6 PROC MOD3;
7 START OS;
8 RETR 01 00899 EN '002';
9 RETR 02 00888 EN '001';
10 LIST 01 00777 'Houses OCC' SASH;
11 LIST 02 00778 'Houses RNT' SASH;
12 PROC MOD3;
13 START OS;
14 RETR 02 00777 EN '002';
15 ERROR: 21
21 LIST OR RETR NUMBER NOT IN RANGE
16 PROC MOD3;
17 RETR 01 004001 EN '001';
18 RETR 02 009023 'FE' '001';
19 LIST 01 005001 'TOTAL MALE' TH;
20 LIST 02 005029 'TOTAL FEMALE' TF;
```

ERROR: DATA DESCRIPTORS REFERENCED AT LINE 00100 (LICE COL) WERE NOT FOUND
009023(189)
005099(20:8)

88
We feel that these diagnostic facilities alone make the SAS/MOD3 user more self-sufficient than our present MOD3 users, who are already very independent.

The most important difference between the MOD3 stand-alone program and the SAS/MOD3 procedure is that the output from SAS/MOD3 is a SAS dataset. This dataset can be analyzed and displayed using any facility provided by SAS. The use of PROC PRINT is sufficient to duplicate the output of MOD3. However, the ability also to perform a variety of statistical analyses or to produce formatted reports is readily available in SAS.