

AVSS

UNIVERSITY OF CALIFORNIA
AUTOMATED VITAL STATISTICS SYSTEM

REPORT GENERATOR MANUAL

COPYRIGHT NOTICE

Copyright 1990 by The Regents of the University of California. All rights reserved.

This document may not be copied without prior consent from a designee of the Regents of the University of California. However, in consideration of the local, State, and federal funding sources that assisted in the development of this product, it is the intent of the Regents to make this information readily available, and requests for permission to copy are likely to be granted. Please address all requests to:

AVSS Project
Community and Organization Research Institute
University of California
Santa Barbara, CA 93106
805-961-3143

TRADEMARKS

Automated Vital Statistics System and *AVSS* are trademarks of the Regents of the University of California.

TABLE OF CONTENTS

CHAPTER 1	INTRODUCTION.....	1-1
CHAPTER 2	GENERATE REPORT.....	2-1
2.1	Foreground Runs.....	2-1
2.2	Background Runs.....	2-2
CHAPTER 3	OUTPUT SAVED REPORT.....	3-1
3.1	Output Saved Report.....	3-1
3.2	Delete Saved Report.....	3-2
CHAPTER 4	QUERY DATABASE.....	4-1
CHAPTER 5	CONCEPTS.....	5-1
CHAPTER 6	REPORT EDITOR.....	6-1
6.1	Command Structure.....	6-1
6.2	Insert.....	6-2
6.3	List.....	6-3
6.4	Modify.....	6-3
6.5	Replace.....	6-5
6.6	Delete.....	6-5
6.7	Copy.....	6-6
6.8	Substitute.....	6-6
6.9	Print.....	6-7
6.10	Exiting.....	6-7
CHAPTER 7	REPORT COMMANDS.....	7-1
7.1	Introduction.....	7-1
7.2	Example One: Individual Report.....	7-1
7.3	Example Two: Listing Report.....	7-3
7.4	Example Three: Summary Table Report.....	7-5
APPENDIX	Naming Conventions.....	A-1
	California Standard Reports.....	A-2
	AVSS System Variables.....	A-3
	MUMPS Command Reference.....	A-4

The University of California's *Automated Vital Statistics System (AVSS)* is an online health information system designed to automate birth and death certificates as well as a variety of other paper forms. To make use of this information, a report generator has been developed to produce form letters, listings, cross-tabulations, and summary reports.

This manual was designed to help you learn to use the *AVSS* Report Generator. It describes how to generate standard reports, query the *AVSS* database and for more advanced users, how to use the Report Editor. Several examples will be used to illustrate specific features of the report generator. Each example will introduce you to more advanced features so that you can incrementally gain knowledge about the Report Generator.

The *AVSS* Report Generator has four sub-options:

- Generate Report
- Output Saved Report
- Query Database
- Report Editor

Generate Report is used to run *AVSS* standard reports or any customized report developed using the Report Editor. These reports may be run in the foreground (during the current session) and sent to the terminal, printer, or PC-DOS file. Alternately, the report may be run in the background and retrieved at a later time.

Output Save Report is used to retrieve output from reports that have been saved to disk. This output may then be sent to a terminal, printer, or PC-DOS file and will remain on the disk until they are deleted. This means that the reports may be viewed or printed many times. However, the saved reports will accumulate and consume valuable disk space until they are deleted.

Query Database is a relatively easy-to-use, interactive tool for *ad-hoc* queries. It may be used to search the database, using one or more variables (fields) from the desired form. Query Database produces one-dimensional tabulations, after locating records that satisfy the conditions specified in the query. The resulting report will indicate the records that met the final selection criteria. Query Database performs relatively rapid searches, depending of the size of the database and the number of conditions specified; its reporting format is, however, somewhat limited.

Report Editor is a more general-purpose tool, which allows the creation of customized reports. Reports may be written in a variety of formats, including: form letters, listings, cross-tabulations and summary reports. Due to the complexity of this option, a set of standard reports has been included in *AVSS* to meet many of the reporting needs for the birth certificate.

To start running standard reports, all you need to know is contained in Chapters 1, 2 and 3. Later, as you become more experienced, you can use the Query Database option for ad-hoc one-dimensional tables reporting your data as described in Chapter 4. The conceptual basis for creating customized reports is discussed in Chapter 5 and Chapter 6 describes how to use the Report Editor, with Chapter 7 and the Appendix serving as references.

We suggest that you use this manual while connected to an *AVSS* computer as described in the *AVSS User's Manual* and follow the dialogue contained herein. Where applicable, responses that you enter are indicated by **[BOLD CAPITAL LETTERS IN BRACKETS]**, while responses from *AVSS* will be in SMALLER-SIZED CAPITALS. Refer to Table 1-1 of the *AVSS User's Manual* for the list of conventions used herein.

2.1 FOREGROUND RUNS

AVSS has been designed to be easy to use for the most basic functions. This is especially true for running pre-programmed reports. Standard reports are ready to be run on your data and are prefixed by a punctuation character such as a “#”. See the Appendix for a list of standard reports. You need only specify the range of records that you wish to analyze. Several examples will be given demonstrating the available features of report generation.

In the first example, you will output the report to the screen and save it to an AVSS file as well.

REPORT GENERATOR OPTION > **[G]** **[ENTER]** ENERATE REPORT

REP. REPORT > **#[CAGE2X]** **[ENTER]** = DEMOGRAPHIC XTABS BY MOTHER’S AGE
*Use **[^L]** to display a list of available reports or **#[#]** for a list of standard reports.*

REPORT FROM DATE <1/1/2002> **[ENTER]** 1/1/2002
TO DATE <T> **[3/31]** **[ENTER]** 3/31/2002

RUN THIS REPORT IN THE BACKGROUND <N> **[ENTER]**

SAVE THE OUTPUT FOR THIS REPORT <N> **[Y]** **[ENTER]**

SOC. SAVE OUTPUT CODE > **[CAGE2X]** **[ENTER]**
This is an arbitrary name up to 8 characters used to store/retrieve a report.

ENTER A DESCRIPTION FOR THIS REPORT > **[CAGE2X REPORT: 1ST 3 MO 2002]**
[ENTER]
This is a textual description of the contents of the report that will be saved.

OUTPUT DEVICE <T> **[ENTER]**
PREPARING REPORT.....

The report is now running; please refer to Chapter 3 for instructions on retrieving the results. Note: when you run the report in the foreground, it will preclude the use of your terminal for other activities until the report is completed. In the next example, a form letter will be created. The AVSS person selector is used to locate the desired record; the output is then routed to a printer connected to the terminal.

REPORT GENERATOR OPTION > **[G]** **[ENTER]** ENERATE REPORT

REP. REPORT > **#[HVL1F]** **[ENTER]** = VERIFICATION LETTER
FILE# OR INDEX TO SEARCH <CHILD’S NAME> **[9876543]** **[ENTER]**

# INDEX VALUES	DATE	FILE#	NEXT SKIPPED
1) SMITH, JANE A	12/4/00	9876543	

IS THIS THE PERSON? <Y> **[ENTER]**

RUN THIS REPORT IN THE BACKGROUND <N> [ENTER]

SAVE THE OUTPUT FOR THIS REPORT <N> [ENTER]

OUTPUT DEVICE <T> [P] [ENTER]

PREPARING REPORT.....

Be sure that the printer is turned on, has the appropriate paper properly loaded and is online. After a short delay, a form letter will be printed.

2.2 BACKGROUND RUNS

In the next example, a standard report is run against the first three months of records in the *AVSS* database. After specifying the report to run, the range to search over is selected. In this case, the report is programmed to prompt for a date range, but filing number ranges, name ranges, cross-index or person selection may also be used in other reports.

SYSTEM OPTION > R [ENTER] EPORT GENERATOR

REPORT GENERATOR OPTION > [G] [ENTER] ENERATE REPORT

REP. REPORT > [#CA] [ENTER]

1 #CAGE1X = TEENAGE CROSS-TAB REPORT

2 #CAGE2X = DEMOGRAPHIC CROSS TABULATIONS BY MOTHER'S AGE

SELECTION # > [1] [ENTER] #CAGE1X = TEENAGE CROSS-TAB REPORT

REPORT FROM DATE <1/1/2002> [ENTER] 1/1/2002

TO DATE <T> [3 31] [ENTER] 3/31/2002

The next set of questions determines how and where the output is to be generated. In this example, the report will run in the background. This allows you to continue working on your terminal while the output of the report is being saved on the computer disk. You can display or print the results at a later time.

RUN THIS REPORT IN THE BACKGROUND <N> [Y] [ENTER]

SOC. SAVE OUTPUT CODE > [CAGE1X] [ENTER]

ENTER A DESCRIPTION FOR THIS REPORT > [CAGE1X REPORT: 1ST QTR 2002] [ENTER]

DELAY UNTIL <NOW> [ENTER]

Enter the 24-hour clock time (HHMM) here to delay the startup of your report.

PREPARING REPORT.....

THIS REPORT BEING RUN IN THE BACKGROUND.

EXIT

CHAPTER 3

OUTPUT SAVED REPORT

3.1 OUTPUT SAVED REPORT

If you run a report in the background or otherwise save the report output, an *AVSS* file will be created containing the results of the report. You may then use the OUTPUT SAVED REPORT sub option to display or print the out put to your terminal or printer.

SYSTEM OPTION > **[R]** **[ENTER]** EPORT GENERATOR

REPORT GENERATOR OPTION > **[O]** **[ENTER]** UTPUT SAVED REPORT

The following prompt will only occur for certain user classifications:

ID. ID CODE# <1> **[ENTER]** CHEN, PETER M

OUT. OUTPUT REPORT CODE > **[^L]** **[ENTER]**

ARCHIV01 = GENERAL BACKUP LOG [6/16/2002 05:24 PM]
CAGE1X = TEENAGE CROSS-TAB REPORT [8/21/2002 09:45 PM]
HNBL2 = NEWBORN LISTING, 2/2002 (10/6/2002 10:45 AM)
TEST = SAMPLE REPORT OUTPUT

OUT. OUTPUT REPORT CODE > **[H]** **[ENTER]** NB1L2 = NEWBORN LISTING, 2/2002
(2/6/2002 10:45 AM)

HNB1L2 = NEWBORN LISTING 2/2002 (2/6/2002 10:45 AM)
HNB1L2 = **[ENTER]**

OUTPUT DEVICE <T> **[^L]** **[ENTER]**

T = TERMINAL
P = PRINTER SLAVED TO TERMINAL
F = PC-DOS FILE
1 = CONSOLE
4 = MODEM

OUTPUT DEVICE <T> **[P]** **[ENTER]**

NEWBORN LISTING, BY HOSPITAL
DATE RANGE: 2/1/2002 TO 2/2/2002

BABY'S PF#	BABY'S NAME	BIRTH DATE	BRTH TIME	BRTH SEX	DLVY WT	DLVY TYPE	MOTHER'S NAME	DOCTOR'S NAME
4321	J FONDA	02/01	2000	F	3724	VAG	J SMITHSON	M BAILEY
1234	T HAYDEN	02/02	1976	M	3333	VAG	R CHAPLIN	D SPOCK

For certain user classification the report output is saved by the id code corresponding to the user who ran the job. Your own id code is the default value. Note that the list of possible output devices will reflect your particular system configuration.

3.2 DELETE SAVED REPORT

Because saved reports consume valuable disk space, they should be deleted after they have been printed. We recommend that you periodically purge old saved reports from your system using the following procedure:

SYSTEM OPTION > **[R]** **[ENTER]** EPORT GENERATOR

REPORT GENERATOR OPTION > **[O]** **[ENTER]** UTPUT SAVED REPORT

The following prompt will only occur for certain user classifications:

ID. ID CODE# <1> **[ENTER]** CHEN, PETER M

OUT. OUTPUT REPORT CODE > **[T]** **[ENTER]** TEST = SAMPLE REPORT OUTPUT

TEST = SAMPLE REPORT OUTPUT

TEST = **[K]** **[ENTER]** OK TO DELETE? **[Y]** **[ENTER]**

You can use this option you want to see values of one particular variable for a selected group of records. The following example will show how to find the values of birth weight (the variable) for teenage mothers (the selected group).

REPORT GENERATOR OPTION > **Q** [ENTER] UERY DATABASE
INITIALIZING.....

AVSS next prompts for the type of form to be searched, e.g. "HCA" for hospital birth record, "LCA" for legal birth certificate or use [^L] for the list of choices:

FORM. SELECTION > [**LCA**] [ENTER] BIRTH CERTIFICATE

Next, *AVSS* prompts for the site information. You may enter more than one item by separating each site with a comma or specifying a range by separating your limits with a double dash [--]. You may also enter your choices one after the other since *AVSS* will continue to prompt for XIND until you are done.

XIND. SITE CODES > [**XX**] [ENTER]

XIND. SITE CODES > [ENTER]

The next items of information are somewhat more complex. LEVEL keeps track of how many variables you have used in the selection criteria. FIELD NUMBER TO BE REPORTED ON is the variable identifier from the selected form. SPECIFIC VALUE(S) determines whether the variable is to be used in the =selection criteria or whether it is the variable to be reported on.

LEVEL 1: FIELD NUMBER TO BE REPORTED ON > [**MAGE**] [ENTER] (MOTHER'S AGE)

LEVEL 1: PLEASE ENTER SPECIFIC VALUE(S) > [**0-19**] [ENTER]

LEVEL 2: FIELD NUMBER TO BE REPORTED ON > [**26**] [ENTER] (BIRTHWEIGHT)

LEVEL 2: PLEASE ENTER SPECIFIC VALUE(S) > [ENTER]

Selection conditions under SPECIFIC VALUE(S) may be in the following order:

VAL1	Specific value.
LOWVAL--HIGHVAL	Range of values (note the two dashed).
*VAL1	Values containing VAL1 (keyword search).
(NULL)	Null values.
(UNDEF)	Undefined values.
C1;C2;...	Combination of any of the above conditions.
NOT;C1;C2;...	For all values except these.

In this example, item MAGE or AGE OF MOTHER is a selection variable with a LOWVAL of 0 and a HIGHVAL of 19. The reporting variable is item 26 or BIRTHWEIGHT. Note that items with range conditions should be numeric.

Next, AVSS prompts for various options pertaining to this report. The default responses are most commonly used.

INDEX TO SEARCH <DATE OF BIRTH> [ENTER]
BEGINNING DATE <1/1/2002> [ENTER] (1/1/2002)
ENDING DATE <T> [ENTER] (11/4/2002)
DO YOU WANT FILE VARIABLES ON ALL RECORDS FOUND DURING QUERY? <N>
[ENTER]

By answering "Y" to this prompt, AVSS will display identifying information for each record matching the selection criteria).

SAVE THESE COMMANDS FOR THE REPORT GENERATOR? <N> [ENTER]

RUN THIS REPORT IN THE BACKGROUND? <N> [ENTER]

SAVE THE OUTPUT FROM THIS REPORT? <N> [ENTER]

OUTPUT DEVICE <T> [ENTER]

SEARCHING DATABASE

XXXXXXXX

Each "X" represents 10 certificates that have been queried.

94 CERTIFICATES ARE REGISTERED FROM 1/1/2002 TO 11/4/2002

THE NUMBER OF CERTIFICATES SEARCHED IS 94

90 CERTIFICATES OF THOSE HAD OTHER THAN THE SPECIFIED VALUE(S) LEAVING
4

THE NUMBER OF CERTIFICATES REPORTED ON IS 4 4.26%

DO YOU WISH TO GROUP, SEE VALUES, CONTINUE OR QUIT (G/V/C/Q) [V]

[ENTER]

MOTHER'S AGE: 0—19 ... 4 SUCCESSES 90 FAILURES 4.26%

BIRTH WEIGHT:

2948 ... 1 ... 25.00% FILE# 59

3076 ... 1 ... 25.00% FILE# 87

3090 ... 1 ... 25.00% FILE# 98

3345 ... 1 ... 25.00% FILE# 83

DO YOU WISH TO GROUP, SEE VALUES, CONTINUE OR QUIT (G/V/C/Q) [G]

[ENTER]

LOW VALUE > [3000] [ENTER] HIGH VALUE > [9999] [ENTER]

FOR RANGE 3000 TO 9999 (INCLUSIVE)

THERE ARE 3 DISTINCT VALUES WITH 3 ENTRIES 75.00%

DO YOU WISH TO SEE THE 3 DISTINCT VALUES AND THEIR STATISTICS? [Y]

[ENTER]

MOTHER'S AGE: 0—19 ... 4 SUCCESSES 90 FAILURES 4.26%

BIRTH WEIGHT:

3076 ... 1 ... 25.00% FILE# 87

3090 ... 1 ... 25.00% FILE# 98

3345 ... 1 ... 25.00% FILE# 83

LOW VALUE > [ENTER]

DO YOU WISH TO GROUP, SEE VALUES, CONTINUE OR QUIT (G/V/C/Q) [Q]

[ENTER]

REPORT GENERATOR OPTION >

To create your own customized reports, you should have a general idea about how *AVSS* data and reports interact. Each *AVSS* record is stored accordingly to the specific site (hospital, registrar, etc.), the type of form (birth certificate, delivery record, etc.), and a record number (local file number, patient number, etc.). Thus, to retrieve data from the *AVSS* database, you must specify the site, the type of form and the specific record(s) that you wish to report on.

When you file data into *AVSS*, each record is stored by a unique identifier called a record number. In addition, cross-index files are created that allow you to find records by other means (e.g., by a name or by an event date). This is important to know if you wish to search a subset of your database, which usually results in saving time. In short, there can be many ways to search your data and the computer must be instructed on method to use.

After specific records have been selected, there are three different types of *AVSS* reports:

- **INDIVIDUAL**
- **LISTING**
- **SUMMARY**

INDIVIDUAL REPORTS: are used to display or print information from a single record. For example, the name, address and demographic or medical data can be used to generate a form letter.

LISTING REPORTS: are used to create line-by-line listings of selected data elements in columnar form. Each line corresponds to a specific record that was selected. For example, you may wish to list the names, addresses and birth weights of all deliveries to women under 20 years of age.

SUMMARY REPORTS: are used to create tabulations of counts for specific values of ranges of a variable. For example, you may wish to report on the distribution of ages in your hospital. You may also tabulate one variable against another to produce a “cross-tabulation.” For example, you may wish to report on the beginning month of prenatal care by mother’s age.

Reports begin with a set of instructions that tell *AVSS* how to print or display stored information. These instructions are called REPORT EDITOR commands. There are three distinct phases to report generation, each with its own set of REPORT EDITOR commands:

- **INITIALIZATION**
- **EXECUTION**
- **OUTPUT**

INITIALIZATION commands are performed before *AVSS* searches the database. As previously mentioned, the REPORT EDITOR must know the site(s) and the type of form(s) that you would like to report on. Or, you might want to print a header of textual information at the beginning of the report. Initialization commands are also used to specify the manner by which you wish to search the database.

EXECUTION commands are performed on each record as *AVSS* scans the database. It is here that you perform the computations to format your data, to list some variables from a particular record, to add up summary values, or to perform intermediate calculations.

OUTPUT commands are performed after *AVSS* has searched the database. Examples include the printing summary variables or tabulations, computing percentages from the final counts, or displaying headers for the final output.

6.1 COMMAND STRUCTURE

At this point, we will begin simulating a “hands-on” tutorial session using *AVSS*. It will be helpful to sign on to *AVSS* at this time and actually experience the interaction with the computer. This tutorial assumes that you are already familiar with the basic *AVSS* interactive method as discussed in the *AVSS User’s Manual*.

A simple example of an Individual Report will be used to demonstrate the use of the REPORT EDITOR.

SYSTEM OPTION > [R] [ENTER] EPORT GENERATOR

REPORT GENERATOR OPTION > [R] [ENTER] EPORT EDITOR

SRC. SAVE REPORT CODE > [TEST1] [ENTER]

REPORT NOT FOUND, OK TO ADD > [Y] [ENTER]

ENTER TITLE OF REPORT > [TUTORIAL EXAMPLE 1] [ENTER]

COPY FROM ANOTHER REPORT <N> [ENTER]

EDITOR COMMAND > [?] [ENTER]

(L)IST RANGE LIST A RANGE OF LINES

(M)ODIFY LINE MAKE CHANGES IN A PARTICULAR LINE

(R)EPLACE LINE REPLACE AN ENTIRE LINE

(I)NSERT LINE INSERT A LINE

(D)ELETE RANGE DELETE A RANGE OF LINES

(C)OPY RANGE COPY RANGE TO A SPECIFIED LINE

(S)UBSTITUTE CHANGE ONE SPECIFIED STRING TO ANOTHER FOR A GIVEN RANGE

(P)RINT RANGE PRINT A RANGE OF LINES TO A SPECIFIED DEVICE

RETURN HIT RETURN WHEN DONE EDITING

LINE # EDIT THIS PARTICULAR LINE

^V DISPLAY SYSTEM VARIABLES AVAILABLE FOR REPORTING

THERE ARE THREE SECTIONS TO A REPORT, NUMBERED AS FOLLOWS:

101-199 = INITIALIZATION, 201-299 = EXECUTION, 301-399 = OUTPUT

RANGE FORMATS: 'F/L', '100-300', '200:LAST', OR SINGLE LINE ('204')

IF THE RANGE OR LINE IS NOT GIVEN IN THE EDITOR COMMAND, AVSS WILL PROMPT YOU FOR IT. ALL (C)OMMANDS MAY BE ABBREVIATED TO THEIR FIRST LETTER.

As throughout *AVSS*, the [?] (question mark) command gives you tutorial help. Note that the [?] command in the example above lists eight REPORT EDITOR commands. You will usually enter one of these commands as follows:

- Insert the desired line number
- Enter the first letter of the report command
- Specify the arguments for the report command

After specifying the line number, *AVSS* expects you to enter one of the appropriate REPORT EDITOR commands. As shown in the example above, the [?] gives a list of these commands. You may specify REPORT EDITOR commands as follows:

- Enter the first letter of the command followed by [ENTER], or
- Enter the first two letters of the command, or
- Enter the first letter of the command and wait 2 seconds.

6.2 INSERT

When you begin to develop a new report from scratch, you will be using the INSERT command to add lines. Here are some of the ways that you can use this command:

```
EDITOR COMMAND > [INSERT] [ENTER] AT LINE > [101] [ENTER]
```

```
EDITOR COMMAND > [I] [ENTER] NSERT LINE > [101] [ENTER]
```

```
EDITOR COMMAND > [INSERT 101] [ENTER]
```

```
EDITOR COMMAND > [I 101] [ENTER]
```

Choosing the latter, we begin entering lines as follows:

```
EDITOR COMMAND > [I 101] [ENTER]
```

```
101) [X] [ENTER] IND      > [0] [ENTER]
102) [F] [ENTER] ORM      > [BC] [ENTER]
103) [S] [ENTER] EARCH    > [P] [ENTER] ERSON
104) [ENTER]
```

```
EDITOR COMMAND >
```

Note that you can continue to INSERT as many lines as you want. Just press [ENTER] when you have completed the information that you wish to insert and you will be returned to the EDITOR COMMAND > prompt. Then, if you wish to start inserting in another line number range, enter a new line number as follows:

```
EDITOR COMMANDS > [I 301] [ENTER]
```

```
301) [E] [ENTER] XECUTE    > [W !,"FIRST NAME: "{1A}] [ENTER]
302) [E] [ENTER] XECUTE    > [W !,"MIDDLE NAME: "{1B}] [ENTER]
303) [E] [ENTER] XECUTE    > [W !,"LAST NAME: "{1C}] [ENTER]
304) [E] [ENTER] XECUTE    > [W !,"BIRTH WEIGHT: "{26}] [ENTER]
```

There are a few restrictions on the line numbers that you specify. Recall from Chapter 5 that there are three phases of a report: Initialization, Execution and Output. If there are no entries for any of these phases, the first line that you are attempting to INSERT must be 101, 201 or 301. *AVSS* checks to see whether you are attempting to INSERT a line that is out of range. For example, in the example report above, trying to INSERT any line number greater than 104 would result in an error since 103 is the last initialization line and 104 is the next expected line number.

6.3 LIST

In the LIST command, like most other REPORT EDITOR commands, you may specify the desired range of lines after the command followed by a space. Report Editor commands that require a range specification are: LIST, DELETE, COPY and PRINT. You may list all the lines in a report by entering [F/L], where [F] stands for the first line and [L] stands for the last line. After typing [L], you specify line ranges by giving the beginning line number, then a range delimiter [-, :, /] or blank, followed by the ending line number. If the line number you specify does not appear in the report, then the next existing line number is used. Also, you can specify a single line number if the beginning and ending range numbers are the same. (i.e. [F], [206], or [L]). Some more examples of the LIST command are as follows:

```
EDITOR COMMAND > [L] [ENTER] IST
                  FROM <FIRST> [101] [ENTER] TO <LAST> [299] [ENTER]
```

```
EDITOR COMMAND > [LIST 301-400] [ENTER]
EDITOR COMMAND > [L] [ENTER] IST FROM <FIRST> [201 205] [ENTER]
```

```
EDITOR COMMAND > [L F/200] [ENTER]
```

Here is how one might use the LIST command on the example report:

```
EDITOR COMMAND > [L F/L] [ENTER]
```

INITIALIZATION:

```
101) XIND          > 0
102) FORM          > BC
103) SEARCH        > PERSON
```

EXECUTION:

OUTPUT:

```
301) EXECUTE       > W !,"FIRST NAME:",{1A}
302) EXECUTE       > W !,"MIDDLE NAME: ",{1B}
303) EXECUTE       > W !,"LAST NAME: ",{1C}
304) EXECUTE       > W !,"BIRTH WEIGHT:",{26}
```

6.4 MODIFY

There are two ways to invoke the MODIFY command. The first is by specifying the MODIFY command and then the line number you wish to edit. For example:

```
EDITOR COMMAND > [M] [ENTER] ODIFY LINE > [102] [ENTER]
```

```
EDITOR COMMAND > [MODIFY 102] [ENTER]
```

```
EDITOR COMMAND > [M 102] [ENTER]
```

The second way is by simply entering the desired line number by itself. For example:

```
EDITOR COMMAND > [102] [ENTER]
```


Having specified the desired line to edit, you now enter a line edit mode in which you can make changes on a character-by-character basis. Again, the question mark can be used to obtain help:

```
102  = FORM      > BC
      = [?] [ENTER]
```

B = BLANK D = DELETE I = INSERT R = REPLACE Q = QUIT (START OVER)
SPACE TO THE DESIRED POSITION AND ENTER THE EDIT CODE

Here we find some new commands, referred to in Table 2-3 of the *AVSS User's Manual* as the Edit commands (as contrasted to the REPORT EDITOR commands discussed in this section). If you are an experienced *AVSS* user you know that the Edit Commands require that you use the space bar to move the cursor directly under the spot you wish to edit. Then you enter the edit code and any new characters followed by an **[ENTER]**. For example, suppose you wish to change this form to use the hospital for code **HB**:

```
102  = FORM      > BC
      =          [RHB] [ENTER]
```

```
102  = FORM      > HB
      =
```

Since you are replacing two characters with two characters, you can use the REPLACE Edit command in this example. Most of the time, however, you will want to delete the bad and insert the good:

```
102  = FORM      > BC
      =          [DDIHB] [ENTER]
```

```
102  = FORM      > HB
      =
```

Having spaced to the desired position, you may only make one change there. That is, you may not make a change and then space to another position. Except for the DELETE/INSERT usage shown above, only one type of line edit is allowed at a time. You may continue editing the desired line until you're satisfied with your changes. You file these changes by pressing **[ENTER]** at the beginning of the edit prompt. In summary, here is a review of all the line edit commands (see the *AVSS User's Manual* for more information):

- [B]** *BLANK*: Change any character to a blank. This command may be repeated. This command may be repeated. Several **[B]**'s may be strung together to blank out an entire section. Please note that although only one type of line edit is normally allowed, the same command may be repeated in certain area.
- [D]** *DELETE*: Delete any character from the line. This command may be repeated. Like the BLANK edit command, sever **[D]**'s may be strung together to delete one section of the line.
- [IXXX]** *INSERT*: Insert the desired characters (**[XXX]** used as an example) at the specified cursor position. Only one INSERT edit command is allowed at a time. However, the DELETE edit command (which can be repeated) may be used before INSERT.
- [RXXX]** *REPLACE*: Replace those characters beginning at the specified cursor position with the given string (**[XXX]** used as an example). Only one REPLACE edit command is allowed at a time.
- [Q]** *QUIT*: If you have made changes, but found they were incorrect or you just want to start over again, the QUIT edit command will give you a fresh copy of the desired line to edit in its original state. Note that QUIT only applies to those commands entered before filing with **[ENTER]**.
- [ENTER]** *FILE*: Once you have completed all the desired changes, pressing **[ENTER]** will file them into the computer. You will then be returned to the EDITOR COMMAND > prompt.

6.5 REPLACE

When you want to replace an entire line with another, use the REPLACE command. You may include the line you wish to insert, or let *AVSS* prompt you for it. Examples of the REPLACE command are as follows:

```
EDITOR COMMAND > [R] [ENTER] EPLACE LINE > [102] [ENTER]
```

```
EDITOR COMMAND > [REPLACE 102] [ENTER]
```

```
EDITOR COMMAND > [R 102] [ENTER]
```

At this point, you specify report commands and parameters exactly as in the INSERT command.

6.6 DELETE

When you want to delete one or more lines from your report, use the DELETE command. This command uses the normal line number range specification, either prompted or unprompted. After specifying the lines to DELETE, *AVSS* will prompt you for verification of the deleted lines. This is just a safety procedure to make sure that you really want to delete those lines. If you find that you accidentally deleted some desired lines and are still editing an old report, you can retrieve the list lines. Just ABANDON the edits of this current session and start over with the original version of the report.

Examples of the DELETE command are as follows:

```
EDITOR COMMAND > [D] [ENTER] DELETE FROM > [102] [ENTER]  
TO > [103] [ENTER] OK TO DELETE <N> [Y] [ENTER]
```

```
EDITOR COMMAND > [DELETE 102-103] [ENTER] OK TO DELETE <N> [Y]  
[ENTER]
```

```
EDITOR COMMAND > [D 102/103] [ENTER] OK TO DELETE <N> [Y] [ENTER]
```

6.7 COPY

Sometimes, you can save entry time by copying lines of your report. This can be helpful when you need to change the order that certain items are computed or displayed. It is also handy when you have several similar statements to enter. *AVSS* will automatically handle numbering the lines for you. However, if you move report commands from one phase to another, it is up to you to make sure the appropriate commands are used. If you forget to do this, *AVSS* will display an error message just before you save the report.

Some examples of the COPY command are:

```
EDITOR COMMAND > [C] [ENTER] COPY FROM > [102] [ENTER] TO 103 [ENTER]  
COPY TO > [103] [ENTER]
```

```
EDITOR COMMAND > [C] [ENTER] COPY FROM > [102:103] [ENTER]  
COPY TO > [103] [ENTER]
```

```
EDITOR COMMAND > [COPY 102-103] [ENTER]  
COPY TO > [103] [ENTER]
```

```
EDITOR COMMAND > [C 102/103] [ENTER]  
COPY TO > [103] [ENTER]
```

6.8 SUBSTITUTE

The SUBSTITUTE command is used to change one character string into another within a specified range of line numbers. This command is useful when making global changes to a report, such as replacing one form variable with another. Also, when copying lines from one phase to another, the report command type will have to be changed.

SUBSTITUTE allows you to do this with one command instead of modifying each individual line. The following examples demonstrate how to use the SUBSTITUTE command to replace 'XXX' with 'YYY' for the entire report.

```
EDITOR COMMAND > [S] [ENTER] SUBSTITUTE  
CHANGE > [XXX] [ENTER] TO > [YYY] [ENTER]  
FROM <FIRST> [ENTER] TO > [306] [ENTER]
```

```
EDITOR COMMAND > [S] [ENTER] SUBSTITUTE  
CHANGE > [XXX] [ENTER] TO > [YYY] [ENTER]  
FROM <FIRST> [101/L] [ENTER]
```

6.9 PRINT

The PRINT command works just like the LIST command, except you are also prompted for the device where you want the report listing routed. Examples are as follows:

```
EDITOR COMMAND > [P] [ENTER]  
FROM <FIRST> [101] [ENTER] TO <LAST> [ENTER]
```

```
OUTPUT DEVICE <T> [P] [ENTER]
```

```
EDITOR COMMAND > [P 201/210] [ENTER]
```

```
OUTPUT DEVICE <T> [1] [ENTER]
```

6.10 EXITING THE REPORT EDITOR

Once you have completed entering and modifying your report, press [ENTER] at the EDITOR COMMAND > prompt. *AVSS* will ask you whether you want to (A)BANDON, (C)ONTINUE or (S)AVE. Enter the first character of the desired option. Here are some of the circumstances under which you would use these options:

ABANDON

- You made numerous changes, but decide against keeping them.
- You actually wanted to copy from this report and save the old one.
- You accidentally deleted the wrong lines.

CONTINUE

- You accidentally press [ENTER].
- You remember one last change you previously forgot.
- You want to take one last look before saving your report.

SAVE

- You are done with editing the report and you want to generate it.

7.1 INTRODUCTION

This chapter will focus on all the different types of report commands and how to use them to create the three basic report forms: Individual, Listing and Summary. For Individual reports, one record is selected from the database. For Listing reports, a group of records are selected and information for each one displayed. For Summary reports, a group of records are selected, but the data is counted and/or grouped into cross-tabulations or some custom format. Some of the REPORT GENERATOR commands use MUMPS syntax. Where appropriate, the syntax and usage of these commands will be explained. As previously discussed, there are three phases to report generation:

- Initialization Phase
- Execution Phase
- Output Phase

Initialization Phase consists of the entire program set up before the database is scanned. System and user variables are set to their default values. The method of record selection is specified; i.e. whether it will be based on the filing variable, other cross-filing index or the person selector. The user is prompted for any range information, if necessary, according to the search method. Also, any headers are printed at this time.

Execution Phase consists of all the tasks performed while the Report Generator is searching the database. This includes record selection, extracting information from individual records, computing cross-tabulation counts, database summary counts and other computations. All the commands in the execution phase are performed in a sequential manner for each record. However, once a record fails particular SEARCH criteria, it is discarded and not further commands are executed.

Output Phase begins after the database search is completed. At this time, summary values are computed and report results are printed. As mentioned earlier, Phase 1 (INITIALIZATION) consists of lines 101-199, Phase 2 (EXECUTION) consists of lines 201-299 and Phase 3 (OUTPUT) consists of lines 301-999. The three types of reporting functions are as follows:

Individual Report: Retrieve one record from the database and display it in another format (a form letter)

Listing Report: Retrieve a selected range of records from the database and display a certain set of variables from each record in a specified order (e.g. daily delivery reports, monthly low birth weight lists, alphabetically sorted listings).

Summary Report: Retrieve a selected range of records from the database and summarize the data for specific items. These summarizations may be organized into cross-tabulations (i.e. birth weight distributions, race tables, age groupings, etc) or into a customized format (i.e. form letter, standard report, etc).

7.2 EXAMPLE ONE: INDIVIDUAL REPORT

Let's review the example Individual Report from Chapter 6:

INITIALIZATION:

```
101) XIND          > 0
102) FORM          > LCA
103) SEARCH        > PERSON
```

EXECUTION:

OUTPUT:

```
301) EXECUTE       > W !,"FIRST NAME:",{1A}
302) EXECUTE       > W !,"MIDDLE NAME: ",{1B}
303) EXECUTE       > W !,"LAST NAME: ",{1C}
304) EXECUTE       > W !,"BIRTH WEIGHT:",{26}
```

This simple report extracts name and birth weight data for a selected patient and prints the information in a line by line fashion. We will now examine each of the commands in the example report to see how they work.

- XIND (Initialization Phase)

XIND is the system variable containing the site code you wish to examine when searching the *AVSS* database. See the Appendix for a definition of all the *AVSS* system variables. In our example program, XIND is '0' which corresponds to the test site. For multiple site specifications, you may use a comma-delimited list of codes, or "--" to specify a range of XIND, or use the MUMPS 'FOR' command structure to examine an incremental range of codes. More than one XIND command may be used within a single report. Some examples of the XIND command follow:

```
101) XIND          > 2005-2075          (All sites from 2005 to 2075, inclusive)

101) XIND          > 205,217,219       (The 3 sites: 205, 217 and 219)

101) XIND          > 317:1:322,405     From code 317 (inclusive), increment by 1 up to
                                         322 (inclusive) and 405
```

Please note that the upper and lower ranges of the '--' specification are inclusive. To see a list of site codes that contains data in your *AVSS* database, use [^L]. When specifying multiple forms, the most important thing to remember is that all the forms should have the same variable names.

```
102) FORM          > LCA,HCA           The 2 forms: LCA and HCA
102) FORM          > B1--B4           All forms from B1 to B4 (i.e. B1,B2,B3,B4)
```

- SEARCH (Initialization Phase)

This command selects the manner in which the database is searched. Only one option may be selected from the following:

```
ALL                Pass through all data for the given XINDs and FORMs.
INDEX              Prompt the user for a range of filing numbers.
PERSON             Prompt the user to select a person from the given XIND and FORM.
```

MASTER	Master indexing is the method <i>AVSS</i> uses to link several types of forms together. Pass through all data whose forms are included in master indexing.
DT	Prompt the user for beginning and ending dates (inclusive).
NM	Prompt the user for lower and upper values of the patient's last name.
cref	Prompt the user for lower and upper ranges of the specified cross-reference index. cref is usually a 2 or 3 letter mnemonic code.

Note in the example above, that only one person is being selected. In this case, record selection is performed using the person selector. Having done this, the database scan is complete and there is no meaning to any specified execution commands.

- OUTPUT (Output Phase)

The OUPUT command is used for customized printing or computation once the report has finished selecting and processing the desired records in the database. In the example, a simple MUMPS write statement is used. Please refer to the Appendix for a complete syntactical description of the WRITE command. The WRITE command is used to output information to your screen, to your printer or to an *AVSS* file. The four types of arguments are print controls, literals, Local variables and Database variables. They may be used in any order, separated by a comma. Print controls are used for pagination, line feed and tabs. Literals are just printed character strings. Local variables are created during the report process and are used to group data, to reformat data or to perform calculations. Database variables refer to stored data values. They are designated by a data field identifier enclosed in curly brackets.

7.3 EXAMPLE TWO: LISTING REPORT

Let's make some additional changes to the first example report and make a listing report out of it:

Initialization:

```

101) XIND          > 0
102) FORM          > LCA
103) SEARCH        > DT
104) INIT          > R !,"ENTER MAXIMUM BIRTHWEIGHT: ",BW
105) INIT          > S SUM=0,BWSUM=0
106) INIT          > W !,"NAME AND BIRTHWEIGHT LISTING, DATE:",TODAY,!

```

Execution:

```

201) EXECUTE      > S SUM=SUM+1
202) SELECT        > INCLUDE {26}<BW
203) EXECUTE      > S BXSUM=BWSUM+1
204) EXECUTE      > W !,"FIRST NAME: ",{1A}
205) EXECUTE      > W !,"MIDDLE NAME: ",{1B}
206) EXECUTE      > W !,"LAST NAME: ",{1C}
207) EXECUTE      > W !,"BIRTH WEIGHT: ",{26}

```

Output:

```

301) OUTPUT       > W !,"TOTAL NUMBER OF RECORDS: ",SUM
302) OUTPUT       > W !,"TOTAL NUMBER OF SELECTED RECORDS: ",BWSUM

```

Note that the SEARCH parameter is now 'DATE' in place of 'PERSON'. Instead of selecting a single record using the person selector, this report will ask the user to specify a date range by which to search the database. Line 104 asks the user to enter a maximum birth weight. *AVSS* System Variables are introduced in Line 106. This line prints a header for the report and shows an example of the use of an *AVSS* System Variable: DATE. These are local variables that you can use in your report that are already set up by the Report Generator. You can list these variables using '^V' at the EDIT COMMAND prompt. A complete listing of the *AVSS* System Variables can be found in the Appendix.

A discussion of the new report commands introduced in this report follows:

- INIT (Initialization Phase)

The INIT command allows execution of MUMPS code to your output before the execution phase. There are a variety of MUMPS commands you may use which will be introduced gradually as needed. With INIT code, you can display header information before the output of a listing report. Since the data is displayed during the execution phase, printing a header during the output phase will happen AFTER your report is done. INIT code is also used to initialize any summary variables or control variables (i.e. BW, BWSUM and SUM) you wish to use during the execution phase.

- EXECUTE (Execution Phase)

The EXECUTE command is general purpose, allowing for the execution of any MUMPS command. Primarily, it is used for the computation of user-defined local variables from the *AVSS* database. The MUMPS SET command is often used for this purpose. Please refer to the Appendix for a complete syntactical description of the SET command. The EXECUTE command is also used to display information about a particular record in a Listing Report, as shown in the above example with a WRITE command.

- SELECT (Execution Phase)

The SELECT command determines whether the current observation is to be included for subsequent listing and/or tabulation. The format of the SELECT command is:

INCLUDE CONDITION OR EXCLUDE CONDITION

where CONDITION is any conditional statement using MUMPS 'IF' syntax. In the example report, line 202 selects the current record only if birth weight is less than the user-specified maximum. Please refer to the Appendix for a complete syntactical description of the IF command.

7.4 EXAMPLE THREE: SUMMARY TABLE REPORT

Initialization:

```
101) FORM          > LCA
102) QUERY         > D INDEX^FSSU K ^T($J)
103) SEARCH       > DT
```

Execution:

```
201) EXECUTE      > S BWGRP=$S({26}?1>("E."UNK",{26}0:"LOBW",{26}4500:
                  "HIBW",1:"NORMAL")
202) EXECUTE      > S SEX=$S({2}?1("E."UNK",1:{2})
203) XTAB         > TAB1: BWGRP BY SEX
```

Output:

```
301) PAGE        > W !,$E(DASH,1,75)
302) PAGE        > W !,"CROSS-TABULATION OF BIRTHWEIGHT BY SEX"
303) PAGE        > W !,$E(DASH,1,75)
304) PAGE        > TAB1: SPCL=TP,V1LABEL="BW GROUP,"V2LABEL="SEX"
```

- QUERY (Initialization Phase)

QUERY is an advanced report command that allows you to execute MUMPS code before the report is prepared. Use the QUERY command to set the values of XIND and FORM in a customized manner. In the example, INDEX ^FSSU sets XIND by prompting the user for a site.

- XTAB (Execution Phase)

The XTAB (cross-tabulate) command allows you to generate 1, 2 or 3-way tables from the database. As the report generator scans the data, the XTAB command computes the values of the cross-tabulated variables and stores them in the NAMED table. These tables may then be output using the TABLE command at output time. The syntax formats for XTAB are:

```
XTAB              > NAME: VARIABLE1
XTAB              > NAME: VARIABLE1 BY VARIABLE2
XTAB              > NAME: VARIABLE1 BY VARIABLE2 BY VARIABLE3
```

VARIABLEn may be a computed local variable or a database variable of the form: {VARIABLE} or {FORM:VARIABLE}. You may use several XTAB commands so long as the NAME you assign to them are different. In one-way crosstabs, VARIABLE1 values comprise the rows. In two-way crosstabs, VARIABLE1 values comprise the rows and VARIABLE2 values comprise the columns. In three-way crosstabs, each VARIABLE1 value has a cross-tab of VARIABLE2 for the rows and VARIABLE3 for the columns.

- PAGE (Execution, Output Phase)

The PAGE command is a line of MUMPS code, which is executed when the report listing goes to the top of the next page. This command is mostly used for formatting headers or generating titles. Please note that PAGE commands begin with the second page, since OUPUT is assumed to be used for the initial titles and headers.

The TABLE command outputs the named cross-tabulation using any given options. The syntax for the TABLE command is as follows:

301) TABLE NAME: OPTION1=VALUE1 OPTION2=VALUE2 ...

Where OPTIONS are:

VSIZE=nnn : The number of spaces to use for the row values
CSIZE=nnn : The number of spaces to use for the column values
VILABEL=xxx : Text label to use for this variable as specified
XTAB > V1 BY V2 BY V3
LAYOUT=DUMP : Display the named table in array format
LAYOUT=TABLE : Display the named table by rows and columns
SPCL=[OPT] : A list of comma-delimited option codes as follows:

RP:Row Percent, CP:Column Percent, TP:Total Percent,
AR:Accumulate by Row, AC:Accumulate by Column,
ARP:Percent Accumulate by Row, ARC:Percent Accumulate by Column.

APPENDIX

NAMING CONVENTIONS

Note: The following information is available through the use of *AVSS* on-line help (using [?]):

THE FOLLOWING CONVENTIONS ARE USED FOR REPORT NAMES IN CALIFORNIA:

% = 1988 OR EARLIER RECORDS (FORMS = BC,HB)

= 1989 OR LATER RECORDS (FORMS = HCA,LCA)

C = COUNTY OF OCCURRENCE DATA (FORMS = BC,LCA)

F = FACILITY (1 OR MORE) OF OCCURRENCE DATA (FORMS = BC,LCA)

H = HOSPITAL OF OCCURRENCE DATA (FORMS = HB,HCA)

R = COUNTY OF RESIDENCE DATA (FORMS = BC,LCA)

L = LINE LISTING REPORT

T = ONE-WAY TABULATION REPORT

X = CROSS TABULATION REPORT

FOR EXAMPLE:

%HNB1L = LINE LISTING ON FORM HB BY HOSPITAL OF OCCURRENCE

%CSR1T = ONE-WAY TABULATION ON FORM BC BY COUNTY OF
RESIDENCE

#FAGE2X = CROSS TAB ON FORM LCA BY FACILITY(S) OF OCCURRENCE

#RPRE2X = CROSS TABULATION OF FORM LCA BY COUNTY OF RESIDENCE

#HMT1L = LINE LISTING ON FORM HCA BY HOSPITAL OF OCCURRENCE

TO SEE SUBSETS OF REPORTS, ENTER IN THE FIRST 1 OR 2 CHARACTERS,

%C = ALL 1988 OR EARLIER COUNTY OF OCCURRENCE REPORTS

#H = ALL 1989 OR LATER HOSPITAL OF OCCURRENCE REPORTS

%F = ALL 1988 OR EARLIER FACILITY(S) OF OCCURRENCE REPORTS

#R = ALL 1989 OR LATER COUNTY OF RESIDENCE REPORTS

APPENDIX CALIFORNIA STANDARD REPORTS

STANDARD REPORTS BY COUNTY (OR LRD) OF OCCURRENCE

The following list is obtained by responding with a [#C] when *AVSS* prompts for report name:

- 1 #CAGE1X=TEENAGE CROSS TABULATIONS BY OCCURRENCE
- 2 #CAGE2X=MATERNAL AGE CROSS TABULATIONS BY OCCURRENCE
- 3 #CANOMT=CONGENITAL ANOMALIES BY OCCURRENCE
- 4 #CBW1X=BIRTH WEIGHT CROSS TABULATIONS BY OCCURRENCE
- 5 #CDLVY1X=METHOD OF DELIVERY CROSS TABULATIONS BY OCCURRENCE
- 6 #CED1X=MATERNAL EDUCATION CROSSTABULATIONS BY OCCURRENCE
- 7 #CETH1X=ETHNICITY CROSS TABULATIONS BY OCCURRENCE
- 8 #CGABWX=GESTATIONAL AGE BY BIRTH WEIGHT BY OCCURRENCE
- 9 #CLDCT=LABOR/DELIVERY COMPLICATIONS BY OCCURRENCE
- 10 #CLOBW1L=LOW BIRTH WEIGHT LISTING BY OCCURRENCE
- 11 #CMDBW1X=MD LICENSE # BY BIRTH WEIGHT BY OCCURRENCE
- 12 #CMDBW2X=MD NAME BY BIRTH WEIGHT BY OCCURRENCE
- 13 #CMDCS1X=MD LICENSE # BY PRI/RPT CSECTION RATES BY OCCURRENCE
- 14 #CMDCS2X=MD NAME BY PRI/RPT CSECTION RATES BY OCCURRENCE
- 15 #CMDM1AX=MD LICENSE # BY MONTH (1ST 6 MONTHS) BY OCCURRENCE
- 16 #CMDM1BX=MD LICENSE # BY MONTH (2ND 6 MONTHS) BY OCCURRENCE
- 17 #CMDM2AX=MD NAME BY MONTH (1ST 6 MONTHS) BY OCCURRENCE
- 18 #CMDM2BX=MD NAME BY MONTH (2ND 6 MONTHS) BY OCCURRENCE
- 19 #CMT1L=MATERNAL TRANSPORT LISTING BY OCCURRENCE
- 20 #CMULT1L=MULTIPLE BIRTH LISTING BY OCCURRENCE
- 21 #CNB1L=NEWBORN LISTING BY OCCURRENCE
- 22 #CNB2L=NON-CONFIDENTIAL NEWBORN LISTING BY OCCURRENCE
- 23 #CPAY1X=SOURCE OF PAYMENT FOR PRENATAL CARE OF OCCURRENCE
- 24 #CPAY2X=SOURCE OF PAYMENT FOR DELIVERY XTABS BY OCCURRENCE
- 25 #CPGCT=PREGNANCY COMPLICATIONS BY OCCURRENCE
- 26 #CPRE1X=PRENATAL CARE CROSS TABULATIONS BY OCCURRENCE
- 27 #CPRE2X=KESSNER PRENATAL CARE INDEX XTABS BY OCCURRENCE
- 28 #CPROCT=CLINICAL PROCEDURES RELATED TO NEWBORN
- 29 #CPRV1X=PRENATAL VISIT CROSS TABULATIONS BY OCCURRENCE
- 30 #CSR1T=SUMMARY REPORT BY OCCURRENCE

APPENDIX

AVSS SYSTEM VARIABLES

BEGIN:	Date and time when the database search began.
BEGINH:	The \$H value when the database search began.
DASH:	String of dashes, use the \$E function for the desired length.
END:	Date and time when the database search finished.
ENDH:	The \$H value when the database search ended.
FOOT:	Number of lines for the bottom margin.
FORM:	Form(s) being searched within the database.
INDEX:	Current filing number being processed.
PAGESIZE:	Number of lines per outage page.
SITE:	Name of the site where the computer is located.
START:	Date and time when this report started.
TIME:	Elapsed time (in seconds) to scan the database.
TODAY:	Today's date.
USER:	Name of the person running the report.
VILO:	Lower range limit of cross-referenced index in database search.
VIHI:	Upper range limit of cross-referenced index in database search.
XIND:	Site code(s) being searched within the database.

APPENDIX MUMPS COMMAND REFERENCE

MUMPS COMMAND: WRITE

SYNTAX: W pc,"OUTPUT STRING",lvar,{var}

where the four arguments used in the command have the following meaning:

pc = Print Control:
= new page
! = new line
?nn = tab to column nn
"OUTPUT STRING" = literal string text
lvar = Local variable
{var} = Database variable from an *AVSS* form

EXAMPLES: W !,?10,"START AT COL 10, DATE:",TODAY,?50,"BW:",{26}
W #,!,"REPORT HEADER LINE",?50,"PAGE: ",PAGE

MUMPS COMMAND: SET

SYNTAX: S var1=expression1,var2=expression2, ...

where varN = Local variable to use in this report
expressionN = Any valid combination of literal strings or numbers, local variables, database variables, operators and/or functions expressing the desired value of varN.

EXAMPLES: S LVAR={26} + 300\250
S TEMP=\$E(LVAR,3,4)_ {2}_ \$P({4A},"A",2)

MUMPS COMMAND: READ

SYNTAX: R pc,"INPUT STRING",var:to

where pc = Print Control:
= new page
! = new line
?nn = tab to column nn
"INPUT STRING" = literal free text
var = the local variable whose value you want to set
to = time-out value (seconds) for the read operation

EXAMPLES: R !,"ENTER INPUT > ",INP
R "ARE YOU SURE?",YN:60

MUMPS COMMAND: IF

SYNTAX: IF Condition
 IF Condition1,Condition2, ...,ConditionN

with the following form for ConditionN:

- 1) ExpressionA
- 2) ExpressionA Compare ExpressionB

ExpressionA and ExpressionB may be of the following form:

- 1) A numeric or string literal, (i.e. 2500,"XYZ",etc.)
- 2) A single variable (local or database), evaluating to TRUE if the value is non-zero.
 Note: non-numeric strings evaluate to zero.
- 3) A valid MUMPS function value: \$X(arguments), where \$x is the desired MUMPS function.
- 4) The negation of the given expression made by prepending an apostrophe: 'expression.
- 5) Two or more expressions combined using valid MUMPS operators, evaluating to TRUE if the value is non-zero.

Note: that these expression forms are the same as those in the MUMPS SET command.

Compare is one of the following MUMPS comparison operators:

<	: Less than.
>	: Greater than.
'<	: Not less than (Greater than or equal to).
'>	: Not greater than (Less than or equal to).
=	: Equal to.
'=	: Not equal to.
[: Contains (does ExpressionA contain ExpressionB).
'[: Does not contain.
]	: Follows (does Expressiona follow Expressionb in the ASCII collating order). In general, this order is as follows: Punctuation marks, numbers, upper-case letters, then lower-case letters.
']	: Does not follow.
'?	: Does not pattern match.
?	: Pattern match with. This is a powerful comparison operator that can be used to determine whether certain patterns exist in ExpressionA. ExpressionB must be in the following format: n1 p2 n2 p2 ...

where n1=Number of occurrences of p1 an '.' May be used to mean "any number of occurrences".

p1=A literal string enclosed in quotes (i.e. "XXX") or one of the following pattern codes:

E=Everything
N=Numeric digit
A=Alphabetic character
U=Upper case character
L=Lower case character
P=Punctuation

For example:

1N.N = Any number
3N1”-“2N1”-“4N = A social security number
3N”-“3N”-“4N = A telephone number
1A.A = Any word
1.A1”-“1A.A = Any hyphenated name
1”JOHN”.E = Any string beginning with “JOHN”

Other MUMPS operators are:

+ : Plus
- (dash) : Minus
* : Multiplied by
/ (slash) : Divided by
\ (back slash) : Integer divided by
: Modulo
& : Logical AND
! : Logical OR
_ (underscore) : Concatenate with

The following MUMPS functions are useful in the Report Generator:

\$E(STRING,P1,P2)

EXTRACT Function

Extracts the substring in STRING beginning at position P! and ending at position P2. If P1 and P2 are missing, then both are assumed to be 1. If P2 is missing, then it is assumed to have the value of P1. This function is useful for truncating long data items in listing reports and ensuring uniform columns.

\$J(NUMBER,LENGTH,D)

JUSTIFY Function

Evaluates NUMBER by rounding to D decimal places and then right-justifies it into a field of LENGTH number of characters. If LENGTH is 0, leading blanks are omitted. This function is useful in formatting percentage or rate values.

\$P(STRING,DL,P1,P2)

PIECE Function

Extract the substring within STRING that is between the P1 and P2 occurrences of DL, the delimiting string. If P1 and P2 are missing, then both are assumed to be 1. If P2 is missing, then P2 is assumed to have the value of P1. This function is useful in extracting information from data items with standard formats. For example, use \$P(DATE,”/”) to derive the first “/” piece (namely month) from DATE or \$P(NAME,””) to derive the first name.

\$S(C1:VALUE1,C2:VALUE2, ... ,1:VALUEn)

SELECT Function

If condition C1 is true, then return VALUE1. Otherwise, check condition C2 and if it's true, return VALUE2. Continue in a similar manner until all the conditions have been checked. Although not required, it is a good practice to specify the last condition as 1 (always true) to prevent select errors. This function is most useful in the EXECUTE statement for recoding form variables into more desirable categories.

\$ZD(DATE)

DATE Function

Not available on all systems, this function converts internal date values (i.e. I4A for date of birth) into MM/DD/YY format. This can be useful in list reports and cross-tabulations by date.

EXAMPLES: SELECT {2}="MALE"
 SELECT {26}<1500!({26}>4000)
 SELECT {1C}'["JR",{1C}]["SR"
 SELECT \$E({29},1,1)="07", \$P({30},'^',2)="YES"

Evaluation Notes:

MUMPS evaluates expressions in a left to right manner, regardless of operation, although you can use parentheses for logical AND (&) and OR (!) conditions.

Numeric values beginning with '0' are considered to be strings, since the leading zero is extraneous.