Menu-driven Reporting
James M. Watts, CIBA-GEIGY Corporation

I. INTRODUCTION

Menu-driven Reporting (MDR) is an on-line ad-hoc report generator provided to the users of the Dyestuffs and Chemicals Division of CIBA-GEIGY located in Greensboro North Carolina. MDR is designed to access any SAS® dataset in the computer. MDR employs a simple operating philosophy. To get a report, the user fills in one screen pushes a button and back to the screen comes a report. If the report is not acceptable, the user may easily return to that screen, change it and generate a new report for viewing.

The purpose of this paper is to discuss the components of MDR and their use. Section II explains a bit of the background of MDR and its value to an organization. Section III illustrates how a user chooses MDR during an interactive session at the terminal, leading the user to the standard report selection screen. Section IV explains the heart of the system, the standard report selection screen and its use. Section V describes the actual generation of an MDR report. Section VI discusses some of the internal processing of MDR. Section VI concludes by summarizing the strengths and weaknesses of MDR.

II. BACKGROUND

The organization of the Dyestuffs and Chemicals Division (D&C) is like many other organizations. There are several product managers each with their own set of products to manage. Similarly, many sales managers have their own set of customers that they support. Department heads have different budgets that they monitor.

Each of these groups of people want to look at the same kinds of information. For example, department managers want to look at expenses. However, a given department manager wants to look only at his expenses. It is easier to keep and maintain the information about all departments in one dataset rather than keep and maintain separate datasets for each manager. The trick is to let a department manager access only the information he wants from a single dataset.

The Information Center (IC) of D&C is composed of several SAS datasets and various methods to access or look at the data contained in these datasets. MDR is only one of the methods an IC user may elect to use to get data. Mr. H. Edwards of CIBA-GEIGY presents an overview of the IC in his paper, "Information Center Using SAS Software--A Success Story" elsewhere in these proceedings.

In order to use MDR, a user must be familiar with the structure of a SAS dataset. D&C users are taught the meaning of an observation and a variable and the convention used to name SAS datasets. The major benefit of MDR is that a user can select a subset of the observations in a given SAS dataset that are of interest to him. In addition, the user may easily retrieve only that subset of variables he wants to see on a report.

III. Flow of an MDR Session

A D&C user types "INFOCNTR" in the TSO command list panel to access the IC. This action brings to the screen the menu shown in Figure 2. As mentioned earlier, MDR is one of several methods users can use to obtain information from the IC. In this section, we will follow the steps of an MDR session (select option A).

---

Figure 1. INFOCNTR Command List Execution

THE DYESENTS AND CHEMICALS DIVISION INFORMATION CENTER MENU

1-PRODUCT MASTER FILE      5-MENU DRIVEN REPORTING
2-CUSTOMER MASTER FILE      6-MENU DRIVEN REPORTING WITH SAS
3-ACCOUNT/BUDGET/FORECAST FILE 7-DISTRIBUTIONS
4-PRODUCT FILE             8-PC FILE TRANSFER
5-ORDER FILE               9-MORE FEATURES AND INFORMATION
X-EXIT THE INFORMATION CENTER

ENTER YOUR CHOICE BELOW:

---

Figure 2. Information Center Primary Menu

Once choice A is entered the user, is prompted for an employee number (See figure 3). MDR uses this number to uniquely identify the user's set of report selection screens. MDR automatically allocates a new dataset if one is not found. The employee number prompting screen is seen only once during an MDR session.

---

Figure 3. Employee Number Prompt Screen
Once MDR successfully allocates the private dataset of report selection screens, the choice of modes of operation is presented as shown in Figure 4. This choice is presented only once during an MDR session. The difference between the two modes of operation occurs after the user completes a report selection screen and activates it for report generation. These differences will be discussed later in Section V.

The heart of MDR is the report selection screen. Each user with enough space to store as many as 35 report selection screens. The standard report selection screen shown in Figure 5 is presented to the user through the terminal (See Figure 5).

The standard report selection screen shown in Figure 5 is presented to the user through the terminal (See Figure 5). This choice is presented only once during an MDR session. The difference between the two modes of operation occurs after the user completes a report selection screen and activates it for report generation. These differences will be discussed later in Section V.

Two modes of operation are presented as shown in Figure 4. The choice is presented only once during an MDR session. The difference between the two modes of operation occurs after the user completes a report selection screen and activates it for report generation. These differences will be discussed later in Section V.

Let's talk about each of these as we build up our first example of reporting under MDR.

1. Dataset Identification

All of the datasets comprising the basic IC for D&C are stored in the computer under the input file name 'G13DYPN.ICSAS.MASTERS'. One of the datasets in this group frequently used is the expenses dataset (EXP). Figure 6 illustrates the entries necessary to specify that EXP is the dataset to be accessed in this particular report.

2. Subsetting Variables

MDR allows the user to report as many as 12 variables. These variables are usually those that are stored in the referenced dataset but they may include temporary variables created or computed at the time the report is generated. (See Computing or Creating Variables). Figure 7 illustrates that this report will have 8 columns of information. Each column will consist of values of the variable specified under the heading VARIABLE. The first column will show a code indicating the budget or cost center (CENTER). As you read down the list under the heading VARIABLE, MDR arranges the columns of the report from left to right on the printed page.
4. Subsetting Observations

As stated above, the primary benefit of MDR is the ability to select only that subset of observations which are of interest to a particular user. To accomplish this the user enters a set of selection criteria. These entries follow a simple set of rules. When the criteria are met by a particular observation in the dataset that observation is captured in a separate working dataset. In this manner the user reduces the IC dataset (in this case EXP) to those observations of interest to him.

Let's look at the rules used by MDR and the syntax needed to carry out these rules. First, each selection is formed by referencing a particular variable and the values that the user is looking for. A selection takes the form:

\[ \text{Variable name} \langle \text{CRITERIA, CRITERIA, ..., CRITERIA} \rangle \]

When parentheses are used, the rule is to select the observation if the referenced variable possesses any one of the specified CRITERIA.

or

\[ \text{Variable name} \langle \text{CRITERIA, CRITERIA, ..., CRITERIA} \rangle \]

When brackets are used, the rule is the reverse of the one above, select the observation if the referenced variable fails all the CRITERIA contained inside the brackets.

CRITERIA takes one of three forms:

a. Single: value or 'value'

b. Range: value-value or 'value'-'value'

c. Special: operator/value or operator/'value'

where operator may be

- GE: greater than or equal to
- GT: greater than
- LE: less than or equal to
- LT: less than
- EQ: equal to
- NE: not equal to
- STR: select if value appears anywhere in the character variable

In this context, value is a string of characters which represents either a numeric comparison value for numbers or a matching value for character variable comparisons.

Another point to remember is that MDR knows the lengths and types of all variables in the specified dataset. This allows MDR to handle selections which specify a smaller number of characters than the actual variable has. If a smaller number of characters is used, MDR assumes that the comparison is to begin with the leftmost character in the actual value of
the referenced variable. Figure 9 illustrates this best. In the example, the user wants all budget centers from EXP for which the budget center (CENTER) begins with the value OE62. In addition, the user wants only expense accounts (ACCOUNT) that begin with 24 or 25.

Figure 9. Subsetting Observations

A range of values would do the job that the list of values did above:

ACCOUNT(24-25)

However, the specification

ACCOUNT(STR/24,STR/25)

would have selected observations for which the value 24 or the value 25 appears embedded anywhere within the number. Obviously, this would be nonsense when applied to an expense number. However, using the STR feature to find a particular string of characters in, say, a product description could be useful.

The CRITERIA do not have to be the same length. For example, the specification

ACCOUNT(24-25,230)

would not only select the observations mentioned above but also any other observations in which the expense number begins with the value 230.

MDR uses an implied AND between separate selection specifications. However, an implied OR is used for the list of CRITERIA within a given selection specification.

There is a special specification which may be used in the selection process. Often the users of MDR are interested in selecting observations based on a ranking according to a particular variable. Two special keywords of the form TOPnmn and LOWnmn can be used to select a set of observations if they have the n highest or the n lowest values of a referenced variable. For example, if the user wants the top 10 accounts from EXP according to year-to-date expenses he would specify:

YTD_EXP(TOP10)

These keywords can be used only once in the selection section of a given report screen. It implies a sorting of the data according to YTD_EXP. This sort step takes place after the other selection criteria are met.

5. Computing or Creating Variables

The selection criteria section is also used to construct a new variable for this report. To do this, the user enters a new name and writes the rule by which that new value is created and encloses it in braces {}.

The user must specify whether the new variable is numeric or character. This is done as follows:

Variable name {arithmetic expression}

Variable name {character expression}

Figure 10 shows how the user creates a new character variable for purposes of reporting.

The new variable (GROUP) is assigned the first two characters of the account number. The difference between this year's expenses and last year's expenses is computed and given the name VARIANCE. Since new variables are created prior to selection criteria being tested, a new variable can be used to select observations. Remember the variables created at this time are not stored in the IC datasets. The user is unable to change the data stored in an IC dataset through the use of MDR!

Figure 10. Computing or Creating Variables

6. Totaling and Subtotaling.

Once a subset of observations and variables has been extracted from the chosen dataset, MDR can be used to simply report these data or to compute useful totals and subtotals and display them on the report. To do this, the user must include in the report screen an indication of those variables to be totaled and the levels at which totals and subtotals are computed. Figure 11 shows asterisks by the variables YTD_EXP and PYT_EXP. This indicates to MDR that only these variables are to be totaled or subtotaled.

615
MDR always computes a grand total for all variables marked with an asterisk whether or not a list of variables is shown on the line "COMPUTE SUBTOTALS FOR:"

7. Report Sequencing

In order for MDR to compute subtotals while printing a report, the data must be in sequence according to the variables under "COMPUTE SUBTOTALS FOR:". Why then is there a place to specify "COMPUTE SUBTOTALS FOR:"? There are three cases when a report sequence would be needed. First, if the user were not interested in subtotalling but only wanted a listing of data in a particular sequence. Second, when the user wants to sequence the report by several variables, e.g. CENTER GROUP ACCOUNT but only wants to compute subtotals at the highest level (CENTER). Third, after the data are collapsed in a no detail request the user may want to rearrange the data for purposes of reporting.

It should be remembered that when "SHOW DETAIL (Y or N):" is set to "Y", the summation is done during a PRINT procedure execution. In this case, the report sequence list and the subtotalling list must bear a special relationship to each other. The computer cannot be printing data which is in order by, say, the variables CENTER and ACCOUNT and at the same time compute subtotal for changes in subsidiary (COMPANY).

MDR looks at the two lists to determine if the last variable in the subtotalling list is one of the variables in the report sequence list. MDR assumes that the user wishes to show subtotals at a level no lower than the last variable in the subtotalling list. If the last variable is not one of the report sequence variables an error condition is signaled and the appropriate error report is produced at the screen. Figure 12 shows a proper specification involving the two lists.

Now that each section of the screen has been explained in some detail let's pull together the parts as shown in Figure 13. Since the user is operating under the control of EDIT, he can move the cursor about the screen to make any changes, deletions or additions he desires.

This screen is complete. However, this screen is not activated for report generation because the entry for "SELECT REPORT (Y or N):" is set to "N". In the next section we discuss the activation of this screen so that MDR will attempt to generate the report.

V. REPORT GENERATION

When given the signal to start (PFZ is pressed), MDR searches through all report selection screens looking for those screens which have been marked as activated ("SELECT REPORT (Y or N):""). Unless this indicator is set to "Y", MDR will ignore the report screen. If MDR finds no activated report selection screens, it will notify the user with the options shown in Figure 14.
Although this can be a nuisance at times, it is better than forgetting to turn off those reports you do not want. MDR automatically sets this indicator to "N" on all screens each time a user enters MDR and each time you return to view the screens during an MDR session.

Figure 14. No Reports Requested Options

Figure 15 shows the report in activated status. This report is ready to go. Press program function key 2 (PF2). After a while the report shown in Figure 16 is presented to you at the terminal screen.

Figure 15. An Activated Report Screen

Figure 16. View Generated Report Under FSLIST.
Notice that there are 37 observations and 8 variables. EXP contains over 4,000 observations and 80 variables. Note that MDR computed and/or created two variables "GROUP" and "VARIANCE" reported under the headings "GROUP CODE" and "CURRENT YEAR LESS PRIOR" respectively. These two variables do not exist on EXP. This report illustrates that MDR has done its job.

The user is now viewing the report under FSLIST. He may browse the report as he desires to determine if it is the report he wants.

The user now presses PF3 and the control menu of MDR shown in Figure 17 is presented.

The user chooses either to route the report to a printer, generate and add new reports to the current set of reports or delete (purge) the current set of reports and generate a new set. If the user selects to route the report to a printer, MDR will initiate the printing process. Figure 17 shows that the report has been queued for printing. MDR will then display the Control Menu again. Again, the user has the same choices.

Suppose the user wishes to activate another report screen. Figure 18 shows that this user has elected to add the next report that's generated to the current one.

Once the user selects the option G or P, MDR will return to FSEDIT and display the first report selection screen from his set of report selection screens. The user then picks the report selection screen he desires, modifies it, activates it and MDR generates a whole new report. In this manner, the user interactively "debugs" a report selection screen. The user responds with an "X" to the MDR Control Menu in order to terminate an interactive MDR session.

Batch Mode of Operation

If the user had selected this mode of operation (See Figure 4), the session proceeds similarly to an interactive session until the report screen is activated by the PF2 key. At this time a job identification prompt (See Figure 19) is presented. After the user responds to the prompt, a batch job is submitted and the primary IC menu is displayed (See Figure 2). The batch job will process the activated screen(s) to produce the desired report(s). This mode of operation is useful for those instances when a user knows that the report works but he does not need the report immediately and, therefore, doesn't wish to wait at the terminal for its generation.

VI. MDR Internal Processing

This section describes, in a general way, what happens behind the scenes to generate a report once the user presses PF2. Three major processes take place. First, MDR builds an internal dictionary of variable names, types and lengths. Second, the entries in the report selection screen are converted to appropriate values and stored in a set of SAS macro variables. Third, assuming no errors are detected, the macro variables are used to expand the SAS code which, when executed, generate the desired report.
The dictionary is created on-line so that any SAS dataset in the computer system can be accessed by MDR. This is accomplished by processing the header records of the chosen dataset to collect all variable names, types and lengths of the actual variables. With this information, variables names on the report selection screen may be checked for correctness. More importantly, the lengths and types of selection criteria may be compared to the lengths and types of the actual variables on the dataset so that the correct logical comparisons may be constructed for subsetting the observations.

The entries on the screen convert to both single and lists of macro variables. For example, only one macro variable is needed to hold the fully qualified name of the chosen SAS dataset. However, the subtotalling list converts very nicely to a list of macro variables. If errors are detected during this step an error macro variable is set to trigger the appropriate error report.

Assuming no errors are detected, MDR uses the macro variables to expand the necessary SAS code to subset the chosen dataset and generate the report. There are at least two SAS steps that are expanded. One of these is a DATA step which accesses the chosen dataset, creates temporary variables and applies the criteria to extract desired observations. For selected "key" variables in the IC datasets D&C has constructed an indexing structure. If a user references one of the key variables, a binary search routine is used to directly access the appropriate observations. If no key variables are found, access to the chosen dataset is sequential.

A PROC PRINT step is always expanded to produce the actual report. The report is stored in a print file (FSLIST is used to browse the print file). If other steps such as sorting and summarizations are needed to produce the desired report, MDR will also expand the SAS code to do these jobs.

The user always presses PF3 to terminate the browsing of the report. The MDR Control Menu of Figure 17 is presented. If the user wants to route the report to a local printer, MDR generates a request to DSPRINT. If the user chooses to continue with MDR the report selection screens are redisplayed under FSEDIT. If the user has routed the report to a printer, he may elect to type "X" in response to the control menu. At this time, MDR de-allocates all transient files and returns to the primary IC menu.

VI. Conclusion

In conclusion, let's look at some of the strengths and weaknesses of MDR and some possible enhancements.

Strengths

* Report generation with simple syntax.
* Automatic access to report requests.
* Easy summarization.
* Easy inclusion and exclusion of variables.
* Interactive report debugging.
* Minimal training is required.

Weaknesses

* Limited formatting.
* Limited space for selection criteria.
* No computations between rows.
* Report from one and only one dataset.
* Limited to 300 or less variables.

Possible Enhancements

* Better error diagnostics.
* Allow multiple datasets to be accessed.
* Increase space for selection criteria.
* Modify dictionary build to accept more than 300 variables.

MDR is used regularly by about 40 people in D&C and an equal number in a couple of other divisions of CIBA-GEIGY. To this point, MDR has served well as an ad-hoc report generator.

*SAS and SAS/FSP are registered trademarks of SAS Institute, Inc., Cary, NC, USA.