

Using COBOL, VIEW and IMAGE A Practical Structured Interface for the Programmer

Peter Somers
Cape Data, Inc.

INTRODUCTION

VIEW or V/3000, Hewlett-Packard's screen handler offers a convenient and versatile method of data collection. To fully utilize the capabilities of VIEW requires the application programmer to go beyond the routines available using the ENTRY program. Ideally the data entry routine will include complete editing including IMAGE data base checking and comprehensive error messages. The routine should allow the programmer to quickly "plug in" new applications and easily perform maintenance. Additionally the program will provide utility routines for data confirmation, screen refreshing, paging, etc.

At our shop, Cape Data, we developed a general purpose VIEW and IMAGE interface program. This program written in structured COBOL allows new applications to go up, with custom editing, in a fraction of the time previously required. The following discussion will cover this interface routine and its application. I will assume that the user has basic knowledge of both VIEW and COBOL.

TABLE OF CONTENTS

1. Screen Design Tips
 - A. Error Messages
 - B. VIEW Editing
 - C. Screen Titles
2. Function Keys
3. COBOL Application Program
 - A. General
 - B. Data Division Considerations
 - C. Main Program Loop
 - D. Program Text
4. SPL Forced Read Subroutine

1. Screen Design

When designing your VIEW input screens using FORMSPEC, the following techniques will help you get the most out of VIEW.

A. Error Message Fields: Add error message fields during form design wherever needed. Place the error message field next to or under the corresponding data field. The error message fields will remain invisible un-

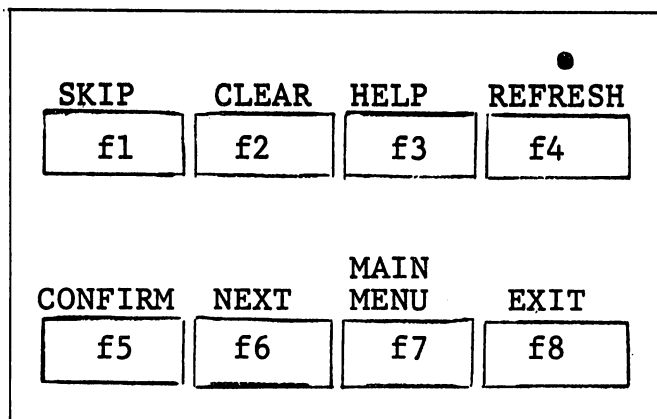
less your program writes a message to the field. Create the field with an enhancement of B (blink) and a field type of D (display) and an initial value of spaces (Fig. 1, Field 3). The last 24th line of the screen is reserved for program error messages.

B. VIEW Editing: As a general rule let VIEW do as much editing as possible. On numeric fields let VIEW zero fill and test for numeric input during the FIELD portion of VIEW's editing. On alpha-numeric input fields allow VIEW to left justify the data and optionally upshift lower case characters (Fig. 1, Field 2).

C. Title: We reserve a Title area on all forms using Field #1. The title field is initialized by VIEW to a save field value. The VIEW forms file can contain the title and any other constants in SAVE FIELDS.

Function Keys

When using a formatted screen program with a Hewlett-Packard 2640-2645 type terminal, a special set of function keys are used by the programs. The 8 function keys are located on the upper right hand side of the terminal's keyboard. They are labeled with blue letters f1 through f8 in 2 rows. A blank Hewlett-Packard template labels the function keys (#7120-5525). On the 2620 family of terminals, the keys are labeled programmatically.



Function Keys

The function keys are used as follows in the interface program: **f1 SKIP** — This key will cause the cursor to

skip to the next block of data. This is useful if you have a number of fields to skip. The tab key only skips a field at a time where the SKIP key will skip to the next block of data.

f2 CLEAR (RESET) — This key causes the screen to clear all fields and set the initial field values (usually blanks). This key is useful if you have created a mess on the screen and want to start over again.

f3 HELP — This key will cause the program to display an instruction screen. A special set of instructions can be displayed relating to the particular form on the screen when the HELP key was pushed. When you have finished reading the HELP instructions, push the ENTER key to return to the last form or the MENU (f7) KEY to return to the MAIN MENU.

f4 REFRESH — This key resets the terminal, erases the screen and brings up a fresh copy of the form. If you loose your form due to a power or line failure or the terminal hangs up, the REFRESH KEY will restore the terminal to normal operation.

f5 CONFIRM — This key is used when you have changed a record in the EDIT mode, or if the program wants confirmation that the data on the screen is acceptable. The program will prompt with a message at the bottom of the screen when a CONFIRM is desired. Before a CONFIRM is requested the data must pass all normal program edits.

Note:

The terminal normally does not read data when the function keys are pushed. If you need to read the screen contents after a function key has been pushed call the IMMVREADFIELD Subroutine to force a read.

f6 NEXT — This key will cause the program to go to the NEXT form or next step.

f7 MAIN MENU — This key causes the program to display the MAIN MENU SELECTION form. Use this key to change from one program mode to another.

f8 EXIT — This key ends the program.

3. COBOL Application Program

A. General: The attached COBOL program has sufficient structure to allow the programmer to readily plug in applications without having to spend additional time coding VIEW procedures. At Cape Data, I have used this program layout to do extensive data entry routines which edit against the IMAGE data base, and provide detailed error messages and help routines. The program procedure division consists of 3 parts:

1. Opening
2. Main Loop
3. Closing

The program contains routines which call the VIEW procedures listed in Fig. 7. The program also contains special VIEW data fields in working storage.

1. Opening — The program opens the terminal, forms file, the data base and displays the MENU screen.

2. Main Loop — After opening the program performs the Main Loop until either the f8 (exit) function key is pushed, or the program encounters a fatal error. The Main Loop consists of 3 parts:

- a. Read Keys and Screen
- b. Edit Input
- c. Process Input (if valid) and Refresh Screen

3. Closing — The program closes the terminal file, forms file and data base.

B. Program Data Division Considerations: The working storage area contains the buffers needed by the various VIEW procedures used. Every VIEW procedure called uses the VIEW-COM buffer (Fig. 2). Most of the fields useful to the programmer have self-explanatory names. Remember the V-Language field must be set to zero for a COBOL program.

The data area passed between the program and VIEW (using VGETBUFFER and VPUTBUFFER calls) is defined as DATA-BUF (Fig. 3). This Buffer is redefined for each screen layout. Note the title field and error message fields.

The program uses a forms table which contains the MENU selection character, form number, next form number, and help form number for each routine. When the user makes a selection from the Screen Menu, the program scans this table to find the corresponding screen references (Fig. 4). The program picks up the Form Name corresponding to the Screen Number from the Form Name Table (Fig. 6). Additional routines and forms can quickly be added by making additional entries in the tables.

3. Program Main Loop

The program goes to the MAIN LOOP and remains there until the user pushes the f8 (exit) function key. The program performs a terminal read (VREAD-FIELDS) each time either a function key or the enter key is depressed. The program then tests to see if any function keys were pushed (VIEW returns the key number pushed into the last key field of the VIEW-COM buffer).

If the ENTER key was pushed (key zero) the program will perform the edit routine corresponding to the screen routine selected. Within each edit routine the program does the following steps (Fig. 5):

1. Zeros the field error array and set the field count.
2. Performs VIEW edits (VFIELDEDITS).
3. Get the data buffer from VIEW (VGETBUFFER)
4. Clear the error message fields.
5. Performs any user defined edits (if an error is detected, a flag is set in the field error array and a message is moved to the appropriate error field)
6. The data area is sent back to VIEW (VPUTBUFFER)
7. Perform VIEW edits again (VFIELDEDITS). This zero fills and justifies data.

8. The field error array is scanned and any error fields are set to blink (VSETERROR).
9. The screen is updated and displayed (VSHOW-FORM).

If the routine passes all edits, the program then goes to the corresponding valid record routine. If an error exists or a function key was depressed, the program will do the appropriate error and screen enhancing routines.

4. SPL Forced Read Subroutine

```
1      $CONTROL SUBPROGRAM
2      << KEPT AS IMMREAD >>
3      << THIS ROUTINE IS CALLED TO FORCE          >>
4      << AN IMMEDIATE READ (RE-READ FOR DATA) >>
5      << IN PARTICULAR CASES WHERE THE USER   >>
6      << USED A SOFT KEY TO INDICATE ACTIONS    >>
7      << HE/SHE WANTS PERFORMED WITH THE DATA >>
8      << THAT HAS BEEN ENTERED ON THE SCREEN    >>
9      << SINCE THE HITTING OF A SOFT KEY DOES  >>
10     << NOT TRANSFER THE ACTUAL BUFFER DATA  >>
11     << A CALL TO THIS ROUTINE OR ONE LIKE    >>
12     << IT IS NEEDED TO GET THE SCREEN DATA  >>
13     << INTO THE PROGRAM WHERE IT CAN BE      >>
14     << WORKED UPON                           >>
15     << IN COBOL, THE CALL WOULD BE           >>
16     << CALL "IMMVREADFIELDS" USING VCONT     >>
17     << WITH VCONT BEING THE V/3000 CONTROL  >>
18     << AREA                                   >>
19     << IN THE USER PROGRAM, THIS SHOULD BE  >>
20     << TREATED EXACTLY AS IF IT HAD BEEN A  >>
21     << CALL TO "VREADFIELDS"                 >>
22     BEGIN PROCEDURE IMMVREADFIELDS(CONT);
23     INTEGER ARRAY CONT;
24     BEGIN
25     PROCEDURE VREADFIELDS(C);
26     INTEGER ARRAY C;
27     OPTION EXTERNAL;
28     CONT(55).(13:2):=X1;
29     VREADFIELDS(CONT);
30     CONT(55).(13:2):=0;
31     END;
32     END.
```

FORMSPEC VERSION A.00.01
FORMS FILE: BUDGFORM.DEVELOP.FDEVELOP

FORM: VENDOR_DATA
REPEAT OPTION: N

NEXT FORM OPTION: C
NEXT FORM: BUDMAINT_HELP

VENDOR MASTER SPECIFICATIONS, INPUT & EDITING

TITLE

VENDOR - MASTER INFORMATION

VENDOR NUMBER V_NBR VEND_ERR
VENDOR'S NAME VEND_NAME
VENDOR ADDRESS VEND_ADDR
VENDOR'S CITY VEND_CITY STATE: SI ZIP: V_ZIP

PAYMENT ADDRESS P_ADDRESS
P_ADDRESS2
PAYMENT CITY P_CITY STATE: PS ZIP: P_ZIP

(PAYMENT ADDRESS USED ONLY IF YOU WANT PAYMENTS GO TO A DIFFERENT ADDRESS)

VENDOR'S PHONE NUMBER: (AC_) EXC-PHON

1099 CODE IN VENDOR CODE VC VENDOR STATUS VS
CODE_ERR STATUS_ERR

FIELD: TITLE
NUM: 1 LEN: 69 NAME: TITLE ENH: NONE FTYPE: D DTYPE: CHAR
INIT VALUE:
*** PROCESSING SPECIFICATIONS ***
INIT
SET TO SFTITLE

FIELD: V_NBR
NUM: 2 LEN: 6 NAME: V_NBR ENH: 1 FTYPE: R DTYPE: DIG
INIT VALUE:
*** PROCESSING SPECIFICATIONS ***
FIELD
JUSTIFY RIGHT
FILL LEADING "0"

FIELD: VEND_ERR
NUM: 3 LEN: 26 NAME: VEND_ERR ENH: B FTYPE: D DTYPE: CHAR
INIT VALUE:

FIELD: VEND_NAME
NUM: 4 LEN: 30 NAME: VEND_NAME ENH: 1 FTYPE: K DTYPE: CHAR
INIT VALUE:

Figure 1

6.4					
6.5	01	VIEW-COM.			
6.6		05	V-STATUS	PIC S9(4)	COMP VALUE ZERO.
6.7		05	V-LANGUAGE	PIC S9(4)	COMP VALUE ZERO.
6.8		05	V-COM-AREA-LEN	PIC S9(4)	COMP VALUE 60.
6.9		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
7		05	VIEW-MODE	PIC S9(4)	COMP VALUE ZERO.
7.1		05	LAST-KEY	PIC S9(4)	COMP VALUE ZERO.
7.2		05	V-NUM-ERRS	PIC S9(4)	COMP VALUE ZERO.
7.3		05	V-WINDOW-ENH	PIC S9(4)	COMP VALUE ZERO.
7.4		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
7.5		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
7.6		05	V-CFNAME	PIC X(15)	VALUE SPACES.
7.7		05	FILLER	PIC X	VALUE SPACES.
7.8		05	V-NFNAME	PIC X(15)	VALUE SPACES.
7.9		05	FILLER	PIC X	VALUE SPACES.
8		05	V-REPEAT-OPT	PIC S9(4)	COMP VALUE ZERO.
8.1		05	V-NF-OPT	PIC S9(4)	COMP VALUE ZERO.
8.2		05	V-NBR-LINES	PIC S9(4)	COMP VALUE ZERO.
8.3		05	V-DBUF-LEN	PIC S9(4)	COMP VALUE ZERO.
8.4		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
8.5		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
8.6		05	V-DELETE-FLAG	PIC S9(4)	COMP VALUE ZERO.
8.7		05	V-SHOW-CONTROL	PIC S9(4)	COMP VALUE ZERO.
8.8		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
8.9		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9.1		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9.2		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9.3		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9.4		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9.5		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9.6		05	V-NUM-RECS	PIC S9(6)	COMP VALUE ZERO.
9.7		05	V-REC-NBR	PIC S9(6)	COMP VALUE ZERO.
9.8		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
9.9		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10		05	V-TERM-FILE-NBR	PIC S9(4)	COMP VALUE ZERO.
10.1		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.2		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.3		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.4		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.5		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.6		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.7		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.8		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
10.9		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
11		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
11.1		05	FILLER	PIC S9(4)	COMP VALUE ZERO.
11.2					

Figure 2

13.3	01	DATA-BUF.	
13.4		05 FILLER	PIC X(69).
13.5		05 DATA-IN	PIC X(443).
13.6			
13.7	01	MENU-DATA	REDEFINES DATA-BUF.
13.8		05 FILLER	PIC X(69).
13.9		05 SELECT-IN	PIC X.
14		05 SELECT-ERR	PIC X(26).
14.1			
14.2	01	VEND-IN	REDEFINES DATA-BUF.
14.3		05 FILLER	PIC X(69).
14.4		05 VEND-NBR	PIC X(6).
14.5		05 VEND-NBR-ERR	PIC X(26).
14.6		05 VEND-NAME	PIC X(30).
14.7		05 S-ADDRESS	PIC X(30).
14.8		05 S-ADDRESS2	PIC X(30).
14.9		05 S-CITY	PIC X(20).
15		05 S-STATE	PIC X(2).
15.1		05 S-ZIP	PIC X(10).
15.2		05 P-ADDRESS	PIC X(30).
15.3		05 P-ADDRESS2	PIC X(30).
15.4		05 P-CITY	PIC X(20).
15.5		05 P-STATE	PIC X(2).
15.6		05 P-ZIP	PIC X(10).
15.7		05 PHONE-NBR.	
15.8		10 PHONE-AC	PIC X(3).
15.9		10 PHONE-EX	PIC X(3).
16		10 PHONE-NO	PIC X(4).
16.1		05 FLAG-1099	PIC X(2).
16.2		05 VEND-CODE	PIC X(2).
16.3		05 VENDOR-STATUS	PIC X(2).
16.4		05 VEND-CODE-ERR	PIC X(26).
16.5		05 VEND-STATUS-ERR	PIC X(26).
16.6			
16.7	01	BUDG-IN	REDEFINES DATA-BUF.
16.8		05 FILLER	PIC X(69).
16.9		05 ACCT-NBR	PIC X(20).
17		05 BUDG-NBR-ERR	PIC X(26).
17.1		05 BUDG-NY-AMT	PIC X(13).
17.2			

Figure 3

19	01 FORM-TABLE.		
19.1	05 FORM-1.	PIC X	VALUE "Z".
19.2	10 FILLER	PIC 9(4)	COMP VALUE 1.
19.3	10 FILLER	PIC 9(4)	COMP VALUE 1.
19.4	10 FILLER	PIC 9(4)	COMP VALUE 14.
19.5	10 FILLER		
19.6	05 FORM-2.	PIC X	VALUE "A".
19.7	10 FILLER	PIC 9(4)	COMP VALUE 2.
19.8	10 FILLER	PIC 9(4)	COMP VALUE 1.
19.9	10 FILLER	PIC 9(4)	COMP VALUE 13.
20	10 FILLER		
20.1	05 FORM-3.	PIC X	VALUE "B".
20.2	10 FILLER	PIC 9(4)	COMP VALUE 3.
20.3	10 FILLER	PIC 9(4)	COMP VALUE 1.
20.4	10 FILLER	PIC 9(4)	COMP VALUE 13.
20.5	10 FILLER		
20.6	05 FORM-4.	PIC X	VALUE "C".
20.7	10 FILLER	PIC 9(4)	COMP VALUE 4.
20.8	10 FILLER	PIC 9(4)	COMP VALUE 1.
20.9	10 FILLER	PIC 9(4)	COMP VALUE 11.
21	10 FILLER		
21.1	05 FORM-5.	PIC X	VALUE "D".
21.2	10 FILLER	PIC 9(4)	COMP VALUE 2.
21.3	10 FILLER	PIC 9(4)	COMP VALUE 1.
21.4	10 FILLER	PIC 9(4)	COMP VALUE 13.
21.5	10 FILLER		
21.6	05 FORM-6.	PIC X	VALUE "E".
21.7	10 FILLER	PIC 9(4)	COMP VALUE 5.
21.8	10 FILLER	PIC 9(4)	COMP VALUE 1.
21.9	10 FILLER	PIC 9(4)	COMP VALUE 12.
22	10 FILLER		
22.1	05 FORM-7.	PIC X	VALUE "Z".
22.2	10 FILLER	PIC 9(4)	COMP VALUE 1.
22.3	10 FILLER	PIC 9(4)	COMP VALUE 1.
22.4	10 FILLER	PIC 9(4)	COMP VALUE 12.
22.5	10 FILLER		
22.6	05 FORM-8.	PIC X	VALUE "Z".
22.7	10 FILLER	PIC 9(4)	COMP VALUE 1.
22.8	10 FILLER	PIC 9(4)	COMP VALUE 1.
22.9	10 FILLER	PIC 9(4)	COMP VALUE 14.
23	10 FILLER		
23.1	05 FORM-9.	PIC X	VALUE "Z".
23.2	10 FILLER	PIC 9(4)	COMP VALUE 1.
23.3	10 FILLER	PIC 9(4)	COMP VALUE 1.
23.4	10 FILLER	PIC 9(4)	COMP VALUE 14.
23.5	10 FILLER		
23.6	05 FORM-10.	PIC X	VALUE "Z".
23.7	10 FILLER	PIC 9(4)	COMP VALUE 1.
23.8	10 FILLER	PIC 9(4)	COMP VALUE 1.
23.9	10 FILLER	PIC 9(4)	COMP VALUE 14.
24	10 FILLER		
24.1			
24.2	01 FORM-SPEC-ARRAY	REDEFINES FORM-TABLE.	
24.3	05 FORM-SPECS	OCCURS 10 TIMES.	
24.4	10 FORM-ID	PIC X.	
24.5	10 FORM-NBR	PIC 9(4) COMP.	
24.6	10 FORM-NEXT	PIC 9(4) COMP.	
24.7	10 FORM-HELP	PIC 9(4) COMP.	

Figure 4

```

24.9 01 FORM-NAME-TABLE.
25      05 FORM-1.
25.1      10 FILLER      PIC X(15) VALUE "BUDMAINT_MENU  ".
25.2      10 FILLER      PIC 9(4)  COMP VALUE 96.
25.3      05 FORM-2.
25.4      10 FILLER      PIC X(15) VALUE "VENDOR_DATA    ".
25.5      10 FILLER      PIC 9(4)  COMP VALUE 357.
25.6      05 FORM-3.
25.7      10 FILLER      PIC X(15) VALUE "BANK_MSTR_DATA  ".
25.8      10 FILLER      PIC 9(4)  COMP VALUE 325.
25.9      05 FORM-4.
26      10 FILLER      PIC X(15) VALUE "BUDGET_LOAD    ".
26.1      10 FILLER      PIC 9(4)  COMP VALUE 128.
26.2      05 FORM-5.
26.3      10 FILLER      PIC X(15) VALUE "                ".
26.4      10 FILLER      PIC 9(4)  COMP VALUE 96.
26.5      05 FORM-6.
26.6      10 FILLER      PIC X(15) VALUE "                ".
26.7      10 FILLER      PIC 9(4)  COMP VALUE 96.
26.8      05 FORM-7.
26.9      10 FILLER      PIC X(15) VALUE "                ".
27      10 FILLER      PIC 9(4)  COMP VALUE 96.
27.1      05 FORM-8.
27.2      10 FILLER      PIC X(15) VALUE "                ".
27.3      10 FILLER      PIC 9(4)  COMP VALUE 96.
27.4      05 FORM-9.
27.5      10 FILLER      PIC X(15) VALUE "                ".
27.6      10 FILLER      PIC 9(4)  COMP VALUE 96.
27.7      05 FORM-10.
27.8      10 FILLER      PIC X(15) VALUE "                ".
27.9      10 FILLER      PIC 9(4)  COMP VALUE 96.
28      05 FORM-11.
28.1      10 FILLER      PIC X(15) VALUE "HELP_BANK_DATA  ".
28.2      10 FILLER      PIC 9(4)  COMP VALUE 69.
28.3      05 FORM-12.
28.4      10 FILLER      PIC X(15) VALUE "HELP_BUDG_LOAD  ".
28.5      10 FILLER      PIC 9(4)  COMP VALUE 69.
28.6      05 FORM-13.
28.7      10 FILLER      PIC X(15) VALUE "HELP_VENDOR    ".
28.8      10 FILLER      PIC 9(4)  COMP VALUE 69.
28.9      05 FORM-14.
29      10 FILLER      PIC X(15) VALUE "BUDMAINT_HELP  ".
29.1      10 FILLER      PIC 9(4)  COMP VALUE 69.
29.2      05 FORM-15.
29.3      10 FILLER      PIC X(15) VALUE "HELP_CREDIT_SUP".
29.4      10 FILLER      PIC 9(4)  COMP VALUE 69.
29.5
29.6 01 FORM-NAME-ARRAY REDEFINES FORM-NAME-TABLE.
29.7      05 FORM-NAME-INFO OCCURS 15 TIMES.
29.8      10 FORM-NAME      PIC X(15).
29.9      10 FORM-DATA-LEN  PIC 9(4) COMP.
30

```

Figure 4a


```

48.3      102000-EDIT-A.
48.4          MOVE ZERO TO CHECK-RESULT.
48.5          MOVE ZERO TO FIELD-ZERO.
48.6          MOVE 16 TO FIELD-CNT.
48.7          PERFORM 805000-VIEW-EDIT.
48.8          PERFORM 807000-GET-BUFFER.
48.9          MOVE SPACES TO VEND-NBR-ERR, VEND-STATUS-ERR.
49          PERFORM 102100-CK-VEND-NBR.
49.1        PERFORM 102200-CK-VEND-CODE.
49.2        PERFORM 102300-CK-VEND-STATUS.
49.3        PERFORM 102400-CK-VEND-1099.
49.4
49.5          PERFORM 809000-PUT-BUFFER.
49.6          PERFORM 805000-VIEW-EDIT.
49.7          IF V-NUM-ERRS NOT = ZERO MOVE 1 TO CHECK-RESULT.
49.8          MOVE ZERO TO FIELD-LOC.
49.9          PERFORM 813000-SET-ERROR-FIELDS FIELD-CNT TIMES.
50
50.1        102100-CK-VEND-NBR.
50.2          MOVE VEND-NBR OF VEND-IN TO ARGUMENT.
50.3          PERFORM 831000-GET-VEND-MSTR.
50.4          IF COND-WORD = 17 NEXT SENTENCE
50.5          ELSE MOVE "INVALID! DUPLICATE NUMBER" TO VEND-NBR-ERR
50.6              MOVE 1 TO FIELD-ERR (2)
50.7              MOVE 1 TO CHECK-RESULT.
50.8
50.9        102200-CK-VEND-CODE.
51          IF VEND-CODE OF VEND-IN = "VN" OR = "VM" OR = "DP"
51.1          NEXT SENTENCE
51.2          ELSE MOVE "INVALID VENDOR CODE!" TO VEND-CODE-ERR
51.3              MOVE 1 TO FIELD-ERR (19)
51.4              MOVE 1 TO CHECK-RESULT.
51.5
51.6        102300-CK-VEND-STATUS.
51.7          IF VENDOR-STATUS OF VEND-IN = "CR" OR = "XX"
51.8          NEXT SENTENCE
51.9          ELSE MOVE "INVALID STATUS CODE!" TO VEND-STATUS-ERR
52              MOVE 1 TO FIELD-ERR (20)
52.1          MOVE 1 TO CHECK-RESULT.
52.2
52.3        102400-CK-VEND-1099.
52.4          IF FLAG-1099 OF VEND-IN = SPACES OR = "Y "
52.5          NEXT SENTENCE
52.6          ELSE
52.7              MOVE 1 TO FIELD-ERR (18)
52.8              MOVE 1 TO CHECK-RESULT.
52.9
53

```

Figure 5

Summary of VIEW Procedures

PROCEDURE	FUNCTION
VCLOSEBATCH	Closes batch file.
VCLOSEFORMF	Closes forms file.
VCLOSETERM	Closes terminal file.
VERRMSG	Returns message associated with error code.
VFIELDEDITS	Edits field data and performs other field processing.
VFINISHFORM	Performs final processing specified for form.
VGETBUFFER	Reads contents of data buffer into user program.
VGETFIELD	Reads field from data buffer into user program.
VGETNEXTFORM	Reads next form into form definition area of memory; window and data buffer are not affected.
VGETtype	Reads field from data buffer to user program, converting data to specified type.
VINITFORM	Sets data buffer to initial values for form.
VOPENBATCH	Opens batch file for processing.
VOPENFORMF	Opens forms file for processing.
VOPENTERM	Opens terminal file for processing.
VPRINTFORM	Prints current form and data on offline list device.
VPUTBUFFER	Writes data from user program to data buffer.
VPUTFIELD	Writes data from user program to field in data buffer.
VPUTtype	Writes data of specified type from user program to data buffer, converting data to ASCII.
VPUTWINDOW	Writes message from user program to window area in memory for later display.
VREADBATCH	Reads record from batch file into data buffer.
VREADFIELDS	Reads input from terminal into data buffer.
VSETERROR	Sets error flag for data field in error; and moves error message to window area.
VSHOWFORM	Updates terminal screen, merging the current form, any data in buffer, and any message in window.
VWRITEBATCH	Writes data from data buffer to batch file.

Figure 6

COBOL VIEW Application Source Program

```
1
1.1 $CONTROL LIST,NOSOURCE,USLINIT,BOUNDS
1.2 IDENTIFICATION DIVISION.
1.3 PROGRAM-ID.
1.4 LAYOUT.
1.5 **THIS PROGRAM PROVIDES A LAYOUT FOR USING VIEW FORMS WITH COBOL.
1.6 ** THE PROGRAM DISPLAYS, EDITS, AND UPDATES MANY FORMS.
1.7 *** VERS 0.00 APRIL 9,1980.
1.8 AUTHOR.
1.9 P SOMERS.
2 INSTALLATION.
2.1 CAPE DATA INC.
2.2 *** (C)COPYRIGHT 1980 CAPE DATA INC. CAPE MAY,NEW JERSEY 08204
2.3 DATE-COMPILED.
2.4 ENVIRONMENT DIVISION.
2.5 CONFIGURATION SECTION.
2.6 SOURCE-COMPUTER. HP-3000.
2.7 OBJECT-COMPUTER. HP-3000.
2.8 INPUT-OUTPUT SECTION.
2.9
3 DATA DIVISION.
3.1
3.2 WORKING-STORAGE SECTION.
3.3 77 FBASE PIC X(13) VALUE " BUDGET.PUB;".
3.4 77 PASSWORD PIC X(8) VALUE "ABC1234;".
3.5 77 DSET-NAME PIC X(16) VALUE SPACES.
3.6 77 NO-ITEM PIC X(2) VALUE "0;".
3.7 77 ITEM PIC X(16) VALUE SPACES.
3.8 77 LIST PIC X(30) VALUE SPACES.
3.9 77 ALL-ITEMS PIC X(2) VALUE "@;".
4 77 SAME-ITEMS PIC X(2) VALUE ";".
4.1 77 ARGUMENT PIC X(20) VALUE SPACES.
4.2 77 MODE1 PIC 9(4) COMP VALUE 1.
4.3 77 MODE2 PIC 9(4) COMP VALUE 2.
4.4 77 MODE3 PIC 9(4) COMP VALUE 3.
4.5 77 MODE5 PIC 9(4) COMP VALUE 5.
4.6 77 MODE7 PIC 9(4) COMP VALUE 7.
4.7 77 MODE-FLAG PIC X.
4.8 77 LOC-FORM PIC 9(4) COMP.
4.9 77 LOC-FORM-NAME PIC 9(4) COMP.
5 77 LOC-FIND PIC 9(4) COMP.
5.1 77 FUTURE-FLAG PIC 9.
5.2 77 NEW-FLAG PIC 9.
5.3 77 LAST-RESULT PIC 9.
5.4 77 CHECK-RESULT PIC S9(4) COMP.
5.5 01 BELL PIC X VALUE " ".
5.6
5.7 01 STATUS-AREA.
5.8 05 COND-WORD PIC S9(4) COMP.
5.9 05 D-L PIC S9(4) COMP.
6 05 R-N PIC S9(9) COMP.
6.1 05 C-L PIC S9(9) COMP.
6.2 05 B-A PIC S9(9) COMP.
6.3 05 F-A PIC S9(9) COMP.
6.4
6.5 01 VIEW-COM.
6.6 05 V-STATUS PIC S9(4) COMP VALUE ZERO.
```

6.7	05 V-LANGUAGE	PIC S9(4)	COMP	VALUE ZERO.
6.8	05 V-COM-AREA-LEN	PIC S9(4)	COMP	VALUE 60.
6.9	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
7	05 VIEW-MODE	PIC S9(4)	COMP	VALUE ZERO.
7.1	05 LAST-KEY	PIC S9(4)	COMP	VALUE ZERO.
7.2	05 V-NUM-ERRS	PIC S9(4)	COMP	VALUE ZERO.
7.3	05 V-WINDOW-ENH	PIC S9(4)	COMP	VALUE ZERO.
7.4	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
7.5	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
7.6	05 V-CFNAME	PIC X(15)	VALUE	SPACES.
7.7	05 FILLER	PIC X	VALUE	SPACES.
7.8	05 V-NFNAME	PIC X(15)	VALUE	SPACES.
7.9	05 FILLER	PIC X	VALUE	SPACES.
8	05 V-REPEAT-OPT	PIC S9(4)	COMP	VALUE ZERO.
8.1	05 V-NF-OPT	PIC S9(4)	COMP	VALUE ZERO.
8.2	05 V-NBR-LINES	PIC S9(4)	COMP	VALUE ZERO.
8.3	05 V-DBUF-LEN	PIC S9(4)	COMP	VALUE ZERO.
8.4	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
8.5	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
8.6	05 V-DELETE-FLAG	PIC S9(4)	COMP	VALUE ZERO.
8.7	05 V-SHOW-CONTROL	PIC S9(4)	COMP	VALUE ZERO.
8.8	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
8.9	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9.1	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9.2	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9.3	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9.4	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9.5	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9.6	05 V-NUM-RECS	PIC S9(6)	COMP	VALUE ZERO.
9.7	05 V-REC-NBR	PIC S9(6)	COMP	VALUE ZERO.
9.8	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
9.9	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10	05 V-TERM-FILE-NBR	PIC S9(4)	COMP	VALUE ZERO.
10.1	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.2	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.3	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.4	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.5	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.6	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.7	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.8	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
10.9	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
11	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
11.1	05 FILLER	PIC S9(4)	COMP	VALUE ZERO.
11.2				
11.3	01 V-FILE-NAME	PIC X(36).		
11.4				
11.5	01 ERR-MES-BUF	PIC X(76).		
11.6	01 LEN-ERR-BUF	PIC S9(4)	COMP	VALUE 76.
11.7	01 LEN-ERR-MES	PIC S9(4)	COMP	VALUE ZERO.
11.8	01 TERM-FILE	PIC X(8)	VALUE	SPACES.
11.9	01 DATA-LEN	PIC S9(4)	COMP	.
12	01 FIELD-NBR	PIC S9(4)	COMP	VALUE ZERO.
12.1	01 ACT-FIELD-LEN	PIC S9(4)	COMP	VALUE ZERO.
12.2	01 NEXT-FIELD-NBR	PIC S9(4)	COMP	VALUE ZERO.
12.3	01 NO-MESSAGE	PIC S9(4)	COMP	VALUE -1.

12.4	01 NUM-IN	PIC X(13).
12.5	01 NUM-OUT	PIC S9(9)V99.
12.6	01 NUM-DISP-13	PIC -----9.99.
12.7		
12.8		
12.9	01 ACCT-MSTR	COPY ACCTMSTR.
13		
13.1	01 VEND-MSTR	COPY VENDMSTR.
13.2		
13.3	01 DATA-BUF.	
13.4	05 FILLER	PIC X(69).
13.5	05 DATA-IN	PIC X(443).
13.6		
13.7	01 MENU-DATA	REDEFINES DATA-BUF.
13.8	05 FILLER	PIC X(69).
13.9	05 SELECT-IN	PIC X.
14	05 SELECT-ERR	PIC X(26).
14.1		
14.2	01 VEND-IN	REDEFINES DATA-BUF.
14.3	05 FILLER	PIC X(69).
14.4	05 VEND-NBR	PIC X(6).
14.5	05 VEND-NBR-ERR	PIC X(26).
14.6	05 VEND-NAME	PIC X(30).
14.7	05 S-ADDRESS	PIC X(30).
14.8	05 S-ADDRESS2	PIC X(30).
14.9	05 S-CITY	PIC X(20).
15	05 S-STATE	PIC X(2).
15.1	05 S-ZIP	PIC X(10).
15.2	05 P-ADDRESS	PIC X(30).
15.3	05 P-ADDRESS2	PIC X(30).
15.4	05 P-CITY	PIC X(20).
15.5	05 P-STATE	PIC X(2).
15.6	05 P-ZIP	PIC X(10).
15.7	05 PHONE-NBR.	
15.8	10 PHONE-AC	PIC X(3).
15.9	10 PHONE-EX	PIC X(3).
16	10 PHONE-VO	PIC X(4).
16.1	05 FLAG-1099	PIC X(2).
16.2	05 VEND-CODE	PIC X(2).
16.3	05 VENDOR-STATUS	PIC X(2).
16.4	05 VEND-CODE-ERR	PIC X(26).
16.5	05 VEND-STATUS-ERR	PIC X(26).
16.6		
16.7	01 BUDG-IN	REDEFINES DATA-BUF.
16.8	05 FILLER	PIC X(69).
16.9	05 ACCT-NBR	PIC X(20).
17	05 BUDG-NBR-ERR	PIC X(26).
17.1	05 BUDG-NY-AMT	PIC X(13).
17.2		
17.3	*****	
17.4	***	FORM TABLE LAYOUT:
17.5	***	SELECTION CHARACTER
17.6	***	FORM NUMBER
17.7	***	NEXT FORM NUMBER
17.8	***	HELP FORM NUMBER
17.9	***	
18	*****	

```

18.1
18.2 *****
18.3 *** FORM NAME TABLE LAYOUT:
18.4 *** FORM NAME (VIEW FORM NAME)
18.5 *** LENGTH OF DATA FIELDS
18.6 ***
18.7 *****
18.8
18.9
19 01 FORM-TABLE.
19.1 05 FORM-1.
19.2 10 FILLER PIC X VALUE "Z".
19.3 10 FILLER PIC 9(4) COMP VALUE 1.
19.4 10 FILLER PIC 9(4) COMP VALUE 1.
19.5 10 FILLER PIC 9(4) COMP VALUE 14.
19.6 05 FORM-2.
19.7 10 FILLER PIC X VALUE "A".
19.8 10 FILLER PIC 9(4) COMP VALUE 2.
19.9 10 FILLER PIC 9(4) COMP VALUE 1.
20 10 FILLER PIC 9(4) COMP VALUE 13.
20.1 05 FORM-3.
20.2 10 FILLER PIC X VALUE "B".
20.3 10 FILLER PIC 9(4) COMP VALUE 3.
20.4 10 FILLER PIC 9(4) COMP VALUE 1.
20.5 10 FILLER PIC 9(4) COMP VALUE 13.
20.6 05 FORM-4.
20.7 10 FILLER PIC X VALUE "C".
20.8 10 FILLER PIC 9(4) COMP VALUE 4.
20.9 10 FILLER PIC 9(4) COMP VALUE 1.
21 10 FILLER PIC 9(4) COMP VALUE 11.
21.1 05 FORM-5.
21.2 10 FILLER PIC X VALUE "D".
21.3 10 FILLER PIC 9(4) COMP VALUE 2.
21.4 10 FILLER PIC 9(4) COMP VALUE 1.
21.5 10 FILLER PIC 9(4) COMP VALUE 13.
21.6 05 FORM-6.
21.7 10 FILLER PIC X VALUE "E".
21.8 10 FILLER PIC 9(4) COMP VALUE 5.
21.9 10 FILLER PIC 9(4) COMP VALUE 1.
22 10 FILLER PIC 9(4) COMP VALUE 12.
22.1 05 FORM-7.
22.2 10 FILLER PIC X VALUE "F".
22.3 10 FILLER PIC 9(4) COMP VALUE 1.
22.4 10 FILLER PIC 9(4) COMP VALUE 1.
22.5 10 FILLER PIC 9(4) COMP VALUE 12.
22.6 05 FORM-8.
22.7 10 FILLER PIC X VALUE "G".
22.8 10 FILLER PIC 9(4) COMP VALUE 1.
22.9 10 FILLER PIC 9(4) COMP VALUE 1.
23 10 FILLER PIC 9(4) COMP VALUE 14.
23.1 05 FORM-9.
23.2 10 FILLER PIC X VALUE "H".
23.3 10 FILLER PIC 9(4) COMP VALUE 1.
23.4 10 FILLER PIC 9(4) COMP VALUE 1.
23.5 10 FILLER PIC 9(4) COMP VALUE 14.
23.6 05 FORM-10.
23.7 10 FILLER PIC X VALUE "I".

```

23.8	10 FILLER	PIC 9(4) COMP VALUE 1.
23.9	10 FILLER	PIC 9(4) COMP VALUE 1.
24	10 FILLER	PIC 9(4) COMP VALUE 14.
24.1		
24.2	01 FORM-SPEC-ARRAY	REDEFINES FORM-TABLE.
24.3	05 FORM-SPECS	OCCURS 10 TIMES.
24.4	10 FORM-ID	PIC X.
24.5	10 FORM-NBR	PIC 9(4) COMP.
24.6	10 FORM-NEXT	PIC 9(4) COMP.
24.7	10 FORM-HELP	PIC 9(4) COMP.
24.8		
24.9	01 FORM-NAME-TABLE.	
25	05 FORM-1.	
25.1	10 FILLER	PIC X(15) VALUE "BUDMAINT_MENU ".
25.2	10 FILLER	PIC 9(4) COMP VALUE 96.
25.3	05 FORM-2.	
25.4	10 FILLER	PIC X(15) VALUE "VENDOR_DATA ".
25.5	10 FILLER	PIC 9(4) COMP VALUE 357.
25.6	05 FORM-3.	
25.7	10 FILLER	PIC X(15) VALUE "BANK_MSTR_DATA ".
25.8	10 FILLER	PIC 9(4) COMP VALUE 325.
25.9	05 FORM-4.	
26	10 FILLER	PIC X(15) VALUE "BUDGET_LOAD ".
26.1	10 FILLER	PIC 9(4) COMP VALUE 128.
26.2	05 FORM-5.	
26.3	10 FILLER	PIC X(15) VALUE " ".
26.4	10 FILLER	PIC 9(4) COMP VALUE 96.
26.5	05 FORM-6.	
26.6	10 FILLER	PIC X(15) VALUE " ".
26.7	10 FILLER	PIC 9(4) COMP VALUE 96.
26.8	05 FORM-7.	
26.9	10 FILLER	PIC X(15) VALUE " ".
27	10 FILLER	PIC 9(4) COMP VALUE 96.
27.1	05 FORM-8.	
27.2	10 FILLER	PIC X(15) VALUE " ".
27.3	10 FILLER	PIC 9(4) COMP VALUE 96.
27.4	05 FORM-9.	
27.5	10 FILLER	PIC X(15) VALUE " ".
27.6	10 FILLER	PIC 9(4) COMP VALUE 96.
27.7	05 FORM-10.	
27.8	10 FILLER	PIC X(15) VALUE " ".
27.9	10 FILLER	PIC 9(4) COMP VALUE 96.
28	05 FORM-11.	
28.1	10 FILLER	PIC X(15) VALUE "HELP_BANK_DATA ".
28.2	10 FILLER	PIC 9(4) COMP VALUE 69.
28.3	05 FORM-12.	
28.4	10 FILLER	PIC X(15) VALUE "HELP_BUDG_LOAD ".
28.5	10 FILLER	PIC 9(4) COMP VALUE 69.
28.6	05 FORM-13.	
28.7	10 FILLER	PIC X(15) VALUE "HELP_VENDOR ".
28.8	10 FILLER	PIC 9(4) COMP VALUE 69.
28.9	05 FORM-14.	
29	10 FILLER	PIC X(15) VALUE "BUDMAINT_HELP ".
29.1	10 FILLER	PIC 9(4) COMP VALUE 69.
29.2	05 FORM-15.	
29.3	10 FILLER	PIC X(15) VALUE "HELP_CREDIT_SUP".
29.4	10 FILLER	PIC 9(4) COMP VALUE 69.

```

29.5
29.6 01 FORM-NAME-ARRAY REDEFINES FORM-NAME-TABLE.
29.7     05 FORM-NAME-INFO OCCURS 15 TIMES.
29.8     10 FORM-NAME PIC X(15).
29.9     10 FORM-DATA-LEN PIC 9(4) COMP.
30
30.1
30.2
30.3
30.4
30.5
30.6
30.7 01 FORMAT-CNIL COPY FORMTCTL.
30.8
30.9 01 FORMAT-13.
31     05 FILLER PIC S9(4) COMP VALUE 1.
31.1    05 FILLER PIC S9(4) COMP VALUE 13.
31.2    05 FILLER PIC S9(4) COMP VALUE 0.
31.3    05 FILLER PIC X VALUE " ".
31.4    05 FILLER PIC X VALUE SPACES.
31.5    05 D13-FORMAT.
31.6      10 FILLER PIC S9(4) COMP VALUE 11.
31.7      10 FILLER PIC S9(4) COMP VALUE ZERO.
31.8      10 FILLER PIC X VALUE SPACES.
31.9      10 FILLER PIC X VALUE "N".
32       10 FILLER PIC X VALUE "1".
32.1      10 FILLER PIC X VALUE "2".
32.2
32.3
32.4 01 ERROR-ARRAY.
32.5     05 FIELD-ERR OCCURS 56 TIMES PIC 9.
32.6 01 FIELD-ZERO REDEFINES ERROR-ARRAY PIC X(56).
32.7
32.8 01 FIELD-CNT PIC 99 COMP.
32.9 01 FIELD-LOC PIC 99 COMP.
33
33.1 *****
33.2 $PAGE
33.3
33.4
33.5 PROCEDURE DIVISION .
33.6 000000-MAIN-PART SECTION 01.
33.7 000000-PROGRAM-LOGIC.
33.8     PERFORM 900000-OPEN-PROGRAM.
33.9     PERFORM 100000-READ-LOOP UNTIL LAST-KEY = 8.
34       PERFORM 970000-CLOSE-PROGRAM.
34.1     STOP RUN.
34.2
34.3 100000-READ-LOOP.
34.4     PERFORM 801000-READ-TERM.
34.5     IF LAST-KEY = 0
34.6         NEXT SENTENCE
34.7     ELSE IF LAST-KEY = 1
34.8         PERFORM 881000-KEY-1
34.9     ELSE IF LAST-KEY = 2
35         PERFORM 882000-KEY-2
35.1     ELSE IF LAST-KEY = 3

```



```

35.2      PERFORM 883000-KEY-3
35.3      ELSE IF LAST-KEY = 4
35.4      PERFORM 884000-KEY-4
35.5      ELSE IF LAST-KEY = 5
35.6      PERFORM 885000-KEY-5
35.7      ELSE IF LAST-KEY = 6
35.8      PERFORM 886000-KEY-6
35.9      ELSE IF LAST-KEY = 7
36        PERFORM 887000-KEY-7
36.1      ELSE
36.2      PERFORM 888000-KEY-8.
36.3
36.4      IF CHECK-RESULT NOT = 0 OR LAST-KEY = 5
36.5      NEXT SENTENCE
36.6      ELSE IF LOC-FORM = 1
36.7      PERFORM 101000-EDIT-MENU
36.8      ELSE IF LOC-FORM = 2
36.9      PERFORM 102000-EDIT-A
37        ELSE IF LOC-FORM = 3
37.1      PERFORM 103000-EDIT-B
37.2      ELSE IF LOC-FORM = 4
37.3      PERFORM 104000-EDIT-C
37.4      ELSE IF LOC-FORM = 5
37.5      PERFORM 105000-EDIT-D
37.6      ELSE IF LOC-FORM = 6
37.7      PERFORM 106000-EDIT-E
37.8      ELSE
37.9      MOVE 4 TO CHECK-RESULT.
38
38.1      IF CHECK-RESULT NOT = ZERO
38.2      NEXT SENTENCE
38.3      ELSE IF LOC-FORM = 1
38.4      PERFORM 150000-VALID-MENU
38.5      ELSE IF LOC-FORM = 2
38.6      PERFORM 151000-VALID-A
38.7      ELSE IF LOC-FORM = 3
38.8      PERFORM 152000-VALID-B
38.9      ELSE IF LOC-FORM = 4
39        PERFORM 153000-VALID-C
39.1      ELSE IF LOC-FORM = 6
39.2      PERFORM 155000-VALID-E
39.3      ELSE IF LAST-KEY NOT = 5
39.4      PERFORM 889000-ASK-CONFIRM
39.5      ELSE IF LOC-FORM = 5
39.6      PERFORM 154000-VALID-D
39.7      ELSE
39.8      MOVE 4 TO CHECK-RESULT.
39.9
40
40.1      IF CHECK-RESULT = 3 PERFORM 851000-REFRESH-TERM.
40.2      IF CHECK-RESULT = 4 MOVE 3 TO V-SHOW-CONTROL
40.3      MOVE "Z" TO MODE-FLAG.
40.4      IF CHECK-RESULT = 0 PERFORM 811000-FORM-INITIALIZE.
40.5      IF CHECK-RESULT = 2 OR = 4 OR = 6 PERFORM 852000-NEXT-FORM.
40.6      IF CHECK-RESULT NOT = 9
40.7      PERFORM 804000-SHOW-FORM
40.8      PERFORM 803000-CLEAR-WINDOW.

```

```

40.9      MOVE ZERO TO CHECK-RESULT, V-SHOW-CONTROL.
41
41.1
41.2      101000-EDIT-MENU.
41.3          MOVE ZERO TO CHECK-RESULT.
41.4          PERFORM 805000-VIEW-EDIT.
41.5          PERFORM 807000-GET-BUFFER.
41.6          MOVE SPACES TO SELECT-ERR.
41.7          IF SELECT-IN = "X" MOVE 8 TO LAST-KEY
41.8              MOVE 9 TO CHECK-RESULT
41.9          ELSE
42              PERFORM 101100-EDIT-MENU-DATA.
42.1
42.2      101100-EDIT-MENU-DATA.
42.3          PERFORM 101110-MENU-SCAN VARYING LOC-FIND FROM 1 BY 1
42.4          UNTIL LOC-FIND > 15
42.5              OR SELECT-IN = FORM-ID (LOC-FIND).
42.6          IF LOC-FIND > 15 PERFORM 101120-MENU-ERROR
42.7              ELSE PERFORM 803000-CLEAR-WINDOW.
42.8
42.9      101110-MENU-SCAN.
43          EXIT.
43.1
43.2      101120-MENU-ERROR.
43.3          MOVE
43.4              "PLEASE SELECT ONE OF THE ABOVE LETTERS AND RE-ENTER"
43.5              TO ERR-MES-BUF.
43.6          MOVE 51 TO LEN-ERR-BUF.
43.7          MOVE 2 TO FIELD-NBR.
43.8          PERFORM 812000-SET-ERROR.
43.9          ADD 1 TO V-NUM-ERRS.
44          MOVE 1 TO CHECK-RESULT.
44.1
44.2
44.3      *****
44.4      *              MODE SELECTIONS              *
44.5      *      A = ADD VENDOR MASTER              N = *
44.6      *      B = ADD BANK MASTER              O = *
44.7      *      C = ADD NEXT YR.BUDGET              P = *
44.8      *      D = EDIT VENDOR MASTER              Q = *
44.9      *      E = LOAD BUDGET              R = *
45      *      F =              S = *
45.1      *      G =              T = *
45.2      *      H =              U = *
45.3      *      I =              V = *
45.4      *      J =              W = *
45.5      *      K =              X = EXIT PROGRAM *
45.6      *      L =              Y = *
45.7      *      M =              Z = MAIN MENU *
45.8      * * *
45.9      * * *
46      *****
46.1
46.2
46.3      *****
46.4      ***              CHECK RESULT CODES              ***
46.5      ***      0 = NO ERRORS              5 = ***

```

46.6	***	1 = ERRORS, RE-READ	6 = NEXT FORM	***
46.7	***	2 = BRING UP NEW FORM	7 =	***
46.8	***	3 = REFRESH TERMINAL	8 =	***
46.9	***	4 = RETURN TO MAIN MENU	9 = EXIT PROGRAM	***
47	***			***

47.1 *****

47.2
47.3

47.4 *****

47.5	***	FUNCTION KEY CODES		***
47.6	***	F1 = SKIP	F5 = CONFIRM ENTRY	***
47.7	***	F2 = CLEAR (INITIALIZE)	F6 = NEXT FORM	***
47.8	***	F3 = HELP	F7 = MAIN MENU	***
47.9	***	F4 = REFRESH SCREEN	F8 = EXIT	***

48 *****

48.1
48.2

48.3 102000-EDIT-A.

48.4 MOVE ZERO TO CHECK-RESULT.
48.5 MOVE ZERO TO FIELD-ZERO.
48.6 MOVE 16 TO FIELD-CNT.
48.7 PERFORM 805000-VIEW-EDIT.
48.8 PERFORM 807000-GET-BUFFER.
48.9 MOVE SPACES TO VEND-NBR-ERR, VEND-STATUS-ERR.
49 PERFORM 102100-CK-VEND-NBR.
49.1 PERFORM 102200-CK-VEND-CODE.
49.2 PERFORM 102300-CK-VEND-STATUS.
49.3 PERFORM 102400-CK-VEND-1099.
49.4
49.5 PERFORM 809000-PUT-BUFFER.
49.6 PERFORM 805000-VIEW-EDIT.
49.7 IF V-NUM-ERRS NOT = ZERO MOVE 1 TO CHECK-RESULT.
49.8 MOVE ZERO TO FIELD-LOC.
49.9 PERFORM 813000-SET-ERROR-FIELDS FIELD-CNT TIMES.

50

50.1 102100-CK-VEND-NBR.

50.2 MOVE VEND-NBR OF VEND-IN TO ARGUMENT.
50.3 PERFORM 831000-GET-VEND-MSTR.
50.4 IF COND-WORD = 17 NEXT SENTENCE
50.5 ELSE MOVE "INVALID! DUPLICATE NUMBER" TO VEND-NBR-ERR
50.6 MOVE 1 TO FIELD-ERR (2)
50.7 MOVE 1 TO CHECK-RESULT.
50.8

50.9 102200-CK-VEND-CODE.

51 IF VEND-CODE OF VEND-IN = "VN" OR = "VM" OR = "DP"
51.1 NEXT SENTENCE
51.2 ELSE MOVE "INVALID VENDOR CODE!" TO VEND-CODE-ERR
51.3 MOVE 1 TO FIELD-ERR (19)
51.4 MOVE 1 TO CHECK-RESULT.
51.5

51.6 102300-CK-VEND-STATUS.

51.7 IF VENDOR-STATUS OF VEND-IN = "CR" OR = "XX"
51.8 NEXT SENTENCE
51.9 ELSE MOVE "INVALID STATUS CODE!" TO VEND-STATUS-ERR
52 MOVE 1 TO FIELD-ERR (20)
52.1 MOVE 1 TO CHECK-RESULT.
52.2

```

52.3 102400-CK-VEND-1099.
52.4 IF FLAG-1099 OF VEND-IN = SPACES OR = "Y "
52.5 NEXT SENTENCE
52.6 ELSE
52.7 MOVE 1 TO FIELD-ERR (18)
52.8 MOVE 1 TO CHECK-RESULT.
52.9
53
53.1
53.2
53.3 103000-EDIT-B.
53.4 MOVE ZERO TO CHECK-RESULT.
53.5 MOVE ZERO TO FIELD-ZERO.
53.6 MOVE 3 TO FIELD-CNT.
53.7 PERFORM 805000-VIEW-EDIT.
53.8 PERFORM 807000-GET-BUFFER.
53.9 PERFORM 103100-CK-VEND-NBR.
54 PERFORM 102200-CK-VEND-CODE.
54.1 PERFORM 102300-CK-VEND-STATUS.
54.2 PERFORM 102400-CK-VEND-1099.
54.3 PERFORM 809000-PUT-BUFFER.
54.4 PERFORM 805000-VIEW-EDIT.
54.5 IF V-NUM-ERRS NOT = ZERO
54.6 MOVE 1 TO CHECK-RESULT.
54.7 MOVE ZERO TO FIELD-LOC.
54.8 PERFORM 813000-SET-ERROR-FIELDS FIELD-CNT TIMES.
54.9
55 103100-CK-VEND-NBR.
55.1 IF NEW-FLAG = ZERO NEXT SENTENCE
55.2 ELSE IF VEND-NBR OF VEND-IN = VEND-NBR OF VEND-MSTR
55.3 MOVE 1 TO NEW-FLAG
55.4 ELSE MOVE ZERO TO NEW-FLAG.
55.5 MOVE VEND-NBR OF VEND-IN TO ARGUMENT.
55.6 PERFORM 831000-GET-VEND-MSTR.
55.7 IF COND-WORD = ZERO AND NEW-FLAG = 1
55.8 NEXT SENTENCE
55.9 ELSE IF COND-WORD = ZERO
56 PERFORM 103110-SETUP-VENDOR
56.1 ELSE
56.2 MOVE "NON-EXISTENT VENDOR NUMBER!" TO VEND-NBR-ERR
56.3 MOVE 1 TO FIELD-ERR (2)
56.4 MOVE 1 TO CHECK-RESULT.
56.5
56.6 103110-SETUP-VENDOR.
56.7 MOVE CORR VEND-MSTR TO VEND-IN.
56.8 MOVE 1 TO NEW-FLAG, CHECK-RESULT.
56.9
57 104000-EDIT-C.
57.1 MOVE ZERO TO CHECK-RESULT.
57.2 MOVE ZERO TO FIELD-ZERO.
57.3 MOVE 3 TO FIELD-CNT.
57.4 PERFORM 805000-VIEW-EDIT.
57.5 PERFORM 807000-GET-BUFFER.
57.6 * MOVE SPACES TO ERROR DISPLAY FIELDS
57.7 * PERFORM EDIT ROUTINES
57.8 PERFORM 809000-PUT-BUFFER.
57.9 PERFORM 805000-VIEW-EDIT.

```

58 IF V-NUM-ERRS NOT = ZERO
 58.1 MOVE 1 TO CHECK-RESULT.
 58.2 MOVE ZERO TO FIELD-LOC.
 58.3 PERFORM 813000-SET-ERROR-FIELDS FIELD-CNT TIMES.
 58.4
 58.5 105000-EDIT-D.
 58.6 MOVE ZERO TO CHECK-RESULT.
 58.7 MOVE ZERO TO FIELD-ZERO.
 58.8 MOVE 3 TO FIELD-CNT.
 58.9 PERFORM 805000-VIEW-EDIT.
 59 PERFORM 807000-GET-BUFFER.
 59.1 * MOVE SPACES TO ERROR DISPLAY FIELDS
 59.2 * PERFORM EDIT ROUTINES
 59.3 PERFORM 809000-PUT-BUFFER.
 59.4 PERFORM 805000-VIEW-EDIT.
 59.5 IF V-NUM-ERRS NOT = ZERO
 59.6 MOVE 1 TO CHECK-RESULT.
 59.7 MOVE ZERO TO FIELD-LOC.
 59.8 PERFORM 813000-SET-ERROR-FIELDS FIELD-CNT TIMES.
 59.9
 60 106000-EDIT-E.
 60.1 MOVE ZERO TO CHECK-RESULT.
 60.2 MOVE ZERO TO FIELD-ZERO.
 60.3 MOVE 3 TO FIELD-CNT.
 60.4 PERFORM 805000-VIEW-EDIT.
 60.5 PERFORM 807000-GET-BUFFER.
 60.6 * MOVE SPACES TO ERROR DISPLAY FIELDS
 60.7 * PERFORM EDIT ROUTINES
 60.8 PERFORM 809000-PUT-BUFFER.
 60.9 PERFORM 805000-VIEW-EDIT.
 61 IF V-NUM-ERRS NOT = ZERO
 61.1 MOVE 1 TO CHECK-RESULT.
 61.2 MOVE ZERO TO FIELD-LOC.
 61.3 PERFORM 813000-SET-ERROR-FIELDS FIELD-CNT TIMES.
 61.4
 61.5
 61.6 150000-VALID-RECORD SECTION 03.
 61.7 150000-VALID-MENU.
 61.8 MOVE 2 TO CHECK-RESULT.
 61.9 MOVE LOC-FIND TO LOC-FORM.
 62 MOVE ZERO TO NEW-FLAG.
 62.1
 62.2 151000-VALID-A.
 62.3
 62.4 152000-VALID-B.
 62.5
 62.6 153000-VALID-C.
 62.7
 62.8 154000-VALID-D.
 62.9
 63 155000-VALID-E.
 63.1
 63.2
 63.3
 63.4
 63.5 800000-UTILITIES SECTION 02.
 63.6

63.7 801000-READ-TERM.
63.8 CALL "VREADFIELDS" USING VIEW-COM.
63.9 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
64
64.1 802000-PUT-WINDOW.
64.2 CALL "VPUTWINDOW" USING VIEW-COM ERR-MES-BUF LEN-ERR-BUF.
64.3
64.4 803000-CLEAR-WINDOW.
64.5 MOVE SPACES TO ERR-MES-BUF.
64.6 PERFORM 802000-PUT-WINDOW.
64.7
64.8
64.9 804000-SHOW-FORM.
65 CALL "VSHOWFORM" USING VIEW-COM.
65.1 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
65.2
65.3
65.4 805000-VIEW-EDIT.
65.5 CALL "VFIELDDEITS" USING VIEW-COM.
65.6
65.7 806000-FINISH-FORM.
65.8 CALL "VFINISHFORM" USING VIEW-COM.
65.9
66 807000-GET-BUFFER.
66.1 CALL "VGETBUFFER" USING VIEW-COM DATA-BUF V-DBUF-LEN.
66.2 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
66.3
66.4 809000-PUT-BUFFER.
66.5 CALL "VPUTBUFFER" USING VIEW-COM DATA-BUF V-DBUF-LEN.
66.6 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
66.7
66.8 810000-GET-FORM-FILE.
66.9 MOVE 0 TO V-REPEAT-OPT.
67 MOVE 0 TO V-NF-OPT.
67.1 CALL "VGETNEXTFORM" USING VIEW-COM.
67.2 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
67.3
67.4 811000-FORM-INITIALIZE.
67.5 MOVE 65 TO V-WINDOW-ENH.
67.6 CALL "VINITFORM" USING VIEW-COM.
67.7 * ADD INDIVIDUAL FORMS INITIALIZATION HERE AS REQ.
67.8 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
67.9 PERFORM 803000-CLEAR-WINDOW.
68 MOVE 1 TO LAST-RESULT.
68.1
68.2 812000-SET-ERROR.
68.3 CALL "VSETERROR" USING VIEW-COM FIELD-NBR ERR-MES-BUF
68.4 LEN-ERR-BUF.
68.5 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
68.6
68.7 813000-SET-ERROR-FIELDS.
68.8 ADD 1 TO FIELD-LOC.
68.9 IF FIELD-ERR (FIELD-LOC) = 1
69 CALL "VSETERROR" USING VIEW-COM FIELD-LOC ERR-MES-BUF
69.1 NO-MESSAGE
69.2 IF V-STATUS NOT = ZERO
69.3 PERFORM 992000-VIEW-ERROR

```

69.4         ELSE
69.5             NEXT SENTENCE
69.6     ELSE
69.7         NEXT SENTENCE.
69.8
69.9     814000-CONFIRM-READ.
70         CALL "IMMVREADFIELDS" USING VIEW-COM.
70.1         IF V-STATUS NOT = ZERO PERFORM 992000-VIEW-ERROR.
70.2
70.3
70.4     820000-SPACE-NUMBER.
70.5         MOVE FORMAT-13 TO FORMAT-CNTL.
70.6         CALL "CAPE'ENTRY" USING FORMAT-CNTL NUM-IN NUM-OUT.
70.7         IF CFIELD-ERR (1) = ZERO AND CENTRY-ERR = ZERO
70.8             MOVE NUM-OUT TO NUM-DISP-13
70.9             ELSE MOVE 1 TO CHECK-RESULT.
71
71.1     830000-GET-ACCT-MSTR.
71.2         MOVE "ACCOUNT-MSTR;" TO DSET-NAME.
71.3         CALL "DBGET" USING FBASE DSET-NAME MODE7 STATUS-AREA
71.4         ALL-ITEMS ACCT-MSTR ARGUMENT.
71.5         IF COND-WORD NOT = ZERO AND NOT = 17
71.6             PERFORM 991000-STATUS-CK
71.7         ELSE
71.8             NEXT SENTENCE.
71.9
72     831000-GET-VEND-MSTR.
72.1         MOVE "VENDOR-MSTR;" TO DSET-NAME.
72.2         CALL "DBGET" USING FBASE DSET-NAME MODE7 STATUS-AREA
72.3         ALL-ITEMS VEND-MSTR ARGUMENT.
72.4         IF COND-WORD = ZERO OR = 17
72.5             NEXT SENTENCE
72.6         ELSE
72.7             PERFORM 991000-STATUS-CK.
72.8
72.9     841000-DB-LOCK.
73         CALL "DBLOCK" USING FBASE DSET-NAME MODE3 STATUS-AREA.
73.1         IF COND-WORD NOT = ZERO PERFORM 991000-STATUS-CK.
73.2
73.3     842000-DB-UNLOCK.
73.4         CALL "DBUNLOCK" USING FBASE DSET-NAME MODE1 STATUS-AREA.
73.5         IF COND-WORD NOT = ZERO PERFORM 991000-STATUS-CK.
73.6
73.7
73.8
73.9     851000-REFRESH-TERM.
74         PERFORM 980000-CLOSE-TERM.
74.1         PERFORM 902000-OPEN-TERM.
74.2         MOVE 3 TO V-SHOW-CONTROL.
74.3         PERFORM 811000-FORM-INITIALIZE.
74.4
74.5     852000-NEXT-FORM.
74.6         IF CHECK-RESULT = 4 MOVE 1 TO LOC-FORM.
74.7         MOVE FORM-NHR (LOC-FORM) TO LOC-FORM-NAME.
74.8         MOVE FORM-NAME (LOC-FORM-NAME) TO V-NFNAME.
74.9         PERFORM 810000-GET-FORM-FILE.
75         PERFORM 811000-FORM-INITIALIZE.

```

75.1
75.2 853000-EDIT-ERROR.
75.3 CALL "VERRMSG" USING VIEW-COM ERR-MES-BUF LEN-ERR-BUF
75.4 LEN-ERR-MES.
75.5 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
75.6 CALL "VPUTWINDOW" USING VIEW-COM ERR-MES-BUF LEN-ERR-BUF.
75.7
75.8 854000-PUT-TITLE.
75.9 * MOVE 1 TO FIELD-NBR.
76 * CALL "VPUTFIELD" USING VIEW-COM FIELD-NBR TITLE-BUF
76.1 * TITLE-LEN ACT-FIELD-LEN NEXT-FIELD-NBR.
76.2 * IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
76.3
76.4
76.5
76.6 860000-START-HELP.
76.7 MOVE FORM-HELP (LOC-FORM) TO LOC-FORM-NAME.
76.8 MOVE FORM-NAME (LOC-FORM-NAME) TO V-NFNAME.
76.9
77 861000-HELP-DISPLAY.
77.1 PERFORM 810000-GET-FORM-FILE.
77.2 PERFORM 854000-PUT-TITLE.
77.3 PERFORM 804000-SHOW-FORM.
77.4 CALL "VREADFIELDS" USING VIEW-COM.
77.5 IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
77.6 IF LAST-KEY = A
77.7 MOVE 9 TO CHECK-RESULT
77.8 ELSE IF LAST-KEY = 7
77.9 MOVE 4 TO CHECK-RESULT
78 ELSE IF LAST-KEY = 4
78.1 MOVE 3 TO CHECK-RESULT
78.2 ELSE
78.3 MOVE 2 TO CHECK-RESULT.
78.4
78.5
78.6 865000-INITIAL-VENDOR.
78.7 * MOVE VEND-NBR OF VEND-MSTR TO VEND-NBR OF VEND-IN.
78.8 * MOVE VEND-NAME OF VEND-MSTR TO VEND-NAME OF VEND-IN.
78.9 * PERFORM 809000-PUT-BUFFER.
79 *
79.1
79.2 870000-UPDATE-ACCT-MSTR.
79.3 CALL "DUPDATE" USING FBASE DSET-NAME MODE1
79.4 STATUS-AREA ALL-ITEMS ACCT-MSTR.
79.5 IF COND-WORD = ZERO NEXT SENTENCE
79.6 ELSE PERFORM 991000-STATUS-CK.
79.7
79.8
79.9
80 881000-KEY-1.
80.1 MOVE 1 TO CHECK-RESULT.
80.2 MOVE "INVALID KEY SELECTED, IGNORED" TO ERR-MES-BUF.
80.3 MOVE 29 TO LEN-ERR-BUF.
80.4 PERFORM 802000-PUT-WINDOW.
80.5
80.6 882000-KEY-2.
80.7 MOVE 1 TO CHECK-RESULT.


```

80.8      PERFORM 811000-FORM-INITIALIZE.
80.9
81      883000-KEY-3.
81.1      PERFORM 860000-START-HELP THRU 861000-HELP-DISPLAY.
81.2
81.3      884000-KEY-4.
81.4      MOVE 3 TO CHECK-RESULT.
81.5
81.6      885000-KEY-5.
81.7      IF LAST-RESULT = ZERO
81.8          PERFORM 814000-CONFIRM-READ
81.9          MOVE 5 TO LAST-KEY
82          MOVE ZERO TO CHECK-RESULT
82.1      ELSE
82.2          PERFORM 890000-INVALID-CONFIRM.
82.3
82.4      886000-KEY-6.
82.5      MOVE 6 TO CHECK-RESULT.
82.6      MOVE FORM-NEXT (LOC-FORM ) TO LOC-FORM.
82.7
82.8
82.9      887000-KEY-7.
83          MOVE "Z" TO MODE-FLAG.
83.1      MOVE 4 TO CHECK-RESULT.
83.2
83.3      888000-KEY-8.
83.4      MOVE 9 TO CHECK-RESULT.
83.5
83.6      889000-ASK-CONFIRM.
83.7      MOVE
83.8          "VALID RECORD, PUSH CONFIRM KEY (F5) TO POST AS SHOWN"
83.9          TO ERR-MES-BUF.
84          MOVE 52 TO LEN-ERR-BUF.
84.1      PERFORM 802000-PUT-WINDOW.
84.2      MOVE 1 TO CHECK-RESULT.
84.3      MOVE ZERO TO LAST-RESULT.
84.4
84.5      890000-INVALID-CONFIRM.
84.6      MOVE 1 TO CHECK-RESULT.
84.7      MOVE
84.8          "INVALID USE OF CONFIRM KEY! CONFIRM NOT REQUESTED!"
84.9          TO ERR-MES-BUF.
85          MOVE 52 TO LEN-ERR-BUF.
85.1      PERFORM 802000-PUT-WINDOW.
85.2
85.3      *****
85.4
85.5      900000-START-STOP SECTION 51.
85.6      900000-OPEN-PROGRAM.
85.7          DISPLAY "VIEW/COBOL LAYOUT PROGRAM VERS. 0.01".
85.8          PERFORM 901000-OPEN-DATA-BASE.
85.9          PERFORM 902000-OPEN-TERM.
86          PERFORM 903000-OPEN-VFORM.
86.1      PERFORM 904000-START-MENU.
86.2
86.3      901000-OPEN-DATA-BASE.
86.4      CALL "DOPEN" USING FBASE PASSWORD MODE1 STATUS-AREA.

```

```

86.5      IF COND-WORD = ZERO
86.6          NEXT SENTENCE
86.7      ELSE
86.8          PERFORM 991000-STATUS-CK
86.9          STOP RUN.
87
87.1      902000-OPEN-TERM.
87.2          MOVE ZERO TO V-STATUS, V-LANGUAGE,
87.3          VIEW-MODE, LAST-KEY, V-NUM-ERRS, V-REPEAT-OPT
87.4          V-NF-OPT.
87.5          CALL "VOPENTERM" USING VIEW-COM TERM-FILE.
87.6          IF V-STATUS = ZERO
87.7              NEXT SENTENCE
87.8          ELSE
87.9              PERFORM 992000-VIEW-ERROR
88              CALL "VCLOSEFORMF" USING VIEW-COM
88.1              DISPLAY ERR-MES-BUF " STOPPING RUN! "
88.2              PERFORM 990000-STOP-PAR.
88.3
88.4      903000-OPEN-VFORM.
88.5          MOVE "BUDGFORM.BUDGET.PROGLIB" TO V-FILE-NAME.
88.6          CALL "VOPENFORMF" USING VIEW-COM V-FILE-NAME.
88.7          IF V-STATUS = ZERO
88.8              NEXT SENTENCE
88.9          ELSE
89              PERFORM 992000-VIEW-ERROR
89.1          CALL "VCLOSETERM" USING VIEW-COM
89.2          DISPLAY ERR-MES-BUF " STOPPING RUN! "
89.3          PERFORM 990000-STOP-PAR.
89.4
89.5      904000-START-MENU.
89.6          MOVE 1 TO LOC-FORM.
89.7          PERFORM 852000-NEXT-FORM.
89.8          PERFORM 804000-SHOW-FORM.
89.9          MOVE ZERO TO CHECK-RESULT.
90
90.1
90.2      970000-CLOSE-PROGRAM.
90.3          PERFORM 980000-CLOSE-TERM.
90.4          PERFORM 981000-CLOSE-VFORM.
90.5          PERFORM 990000-STOP-PAR.
90.6
90.7      980000-CLOSE-TERM.
90.8          CALL "VCLOSETERM" USING VIEW-COM.
90.9          IF V-STATUS NOT = 0 PERFORM 992000-VIEW-ERROR.
91
91.1      981000-CLOSE-VFORM.
91.2          CALL "VCLOSEFORMF" USING VIEW-COM.
91.3          IF V-STATUS NOT = ZERO PERFORM 992000-VIEW-ERROR.
91.4
91.5
91.6      990000-STOP-PAR.
91.7          CALL "DBCLOSE" USING FBASE DSET-NAME MODE1 STATUS-AREA.
91.8          STOP RUN.
91.9
92
92.1      991000-STATUS-CK.

```

92.2 PERFORM 980000-CLOSE-TERM.
92.3 PERFORM 981000-CLOSE-VFORM.
92.4 CALL "DBEXPLAIN" USING STATUS-AREA.
92.5 PERFORM 990000-STOP-PAR.
92.6
92.7 992000-VIEW-ERROR.
92.8
92.9 CALL "VERRMSG" USING VIEW-COM ERR-MES-BUF LEN-ERR-BUF
93 LEN-ERR-MES.
93.1 DISPLAY BELL "VIEW ERROR!!!".
93.2 DISPLAY ERR-MES-BUF.
93.3 DISPLAY BELL "PROGRAM TERMINATED DUE TO ABOVE ERROR!!!".
93.4 PERFORM 990000-STOP-PAR.
93.5

