



VisualAge Pacbase 2.5

**UNISYS 2200 SERIES OLSD
REFERENCE MANUAL**

DDOU1000021A

Note

Before using this document, read the general information under "Notices" on the next page.

According to your license agreement, you may consult or download the complete up-to-date collection of the VisualAge Pacbase documentation from the VisualAge Pacbase Support Center at:

<http://www.software.ibm.com/ad/vapacbase/support.htm>

Consult the Catalog section in the Documentation home page to make sure you have the most recent edition of this document.

First Edition (April 1994)

This edition applies to the following licensed programs:

- VisualAge Pacbase Version 2.0
- VisualAge Pacbase Version 2.5

Comments on publications (including document reference number) should be sent electronically through the Support Center Web site at:

<http://www.software.ibm.com/ad/vapacbase/support.htm>

or to the following postal address:

IBM Paris Laboratory
VisualAge Pacbase Support
30, rue du Château des Rentiers
75640 PARIS Cedex 13
FRANCE

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1983, 1999. All rights reserved.

Note to U.S. Government Users – Documentation related to restricted rights – Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

NOTICES

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Intellectual Property and Licensing
International Business Machines Corporation
North Castle Drive, Armonk, New-York 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of information which has been exchanged, should contact:

IBM Paris Laboratory
SMC Department
30, rue du Château des Rentiers
75640 PARIS Cedex 13
FRANCE

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

IBM may change this publication, the product described herein, or both.

TRADEMARKS

IBM is a trademark of International Business Machines Corporation, Inc. AIX, AS/400, CICS, CICS/MVS, CICS/VSE, COBOL/2, DB2, IMS, MQSeries, OS/2, PACBASE, RACF, RS/6000, SQL/DS, TeamConnection, and VisualAge are trademarks of International Business Machines Corporation, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

All other company, product, and service names may be trademarks of their respective owners.

TABLE OF CONTENTS

1. INTRODUCTION	7
1.1. PURPOSE OF THE MANUAL	8
1.2. MANUAL'S CONTENTS & REMINDERS ON THE OLSD FUNCTION	9
1.3. UNISYS 2200 SCREEN GENERATION - OPERATING MODE.....	12
2. PRESENTATION OF THE EXAMPLE	14
2.1. THE 'DO' DIALOGUE.....	15
2.2. THE 'DO0030' SCREEN	18
3. GENERATED PROGRAM: DATA DIVISION	38
3.1. BEGINNING OF PROGRAM	39
3.2. BEGINNING OF WORKING-STORAGE	41
3.3. SEGMENT DESCRIPTION	48
3.4. FORM DESCRIPTION	50
3.5. DESCRIPTION OF VALIDATION AREAS.....	59
3.6. TABLE-OF-ATTRIBUTES AND SEGMENT VARIABLES	67
3.7. COMMUNICATION AREA DESCRIPTION.....	71
4. GENERATED PROGRAM: PROCEDURE DIVISION	74
4.1. STRUCTURE OF THE PROCEDURE DIVISION.....	75
4.2. INITIALIZATIONS (F01)	77
4.3. RECEPTION AND OPERATION CODE (F05)	79
4.4. CATEGORY PROCESSING LOOP (F10).....	81
4.5. VALIDATION OF TRANSACTION CODE (F15).....	83
4.6. DATA ELEMENT VALIDATION (F20).....	85
4.7. SEGMENT ACCESS FOR VALIDATION (F25)	90
4.8. DATA ELEMENT TRANSFER (F30)	94
4.9. SEGMENT ACCESS FOR UPDATE (F35).....	96
4.10. END OF RECEPTION (F40).....	99
4.11. DISPLAY PREPARATION (F50).....	102
4.12. CATEGORY PROCESSING LOOP (F55).....	104
4.13. SEGMENT ACCESS FOR DISPLAY (F60).....	106
4.14. DATA ELEMENT TRANSFER (F65).....	108
4.15. ERROR PROCESSING (F70).....	111
4.16. DISPLAY AND END OF PROGRAM (F8Z).....	116
4.17. PHYSICAL SEGMENT ACCESS ROUTINES (F80).....	118
4.18. PERFORMED VALIDATION FUNCTIONS (F81).....	123
4.19. CALLED USER FUNCTIONS	129
5. HELP FUNCTION	131
5.1. INTRODUCTION	132
5.2. GENERATED 'HELP' PROGRAM	137
6. CHART OF VARIABLES AND CONSTANTS	151

1. INTRODUCTION

INTRODUCTION	PAGE	8
PURPOSE OF THE MANUAL		1
		1

1.1. PURPOSE OF THE MANUAL

PURPOSE OF THE MANUAL

The purpose of the UNISYS 2000 ON-LINE SYSTEMS DEVELOPMENT Reference Manual is to present a Screen generated by the OLSO function. This manual only provides specific information on the description and generation of dialogues which will operate under UNISYS 2200.

The basic rules and general characteristics of dialogue management are fully described in the ON-LINE SYSTEMS DEVELOPMENT (OLSO) Reference Manual, which is common to all on-line monitors.

The Screen example features accesses to a DMS 1100 Database.

INTRODUCTION	PAGE	9
MANUAL'S CONTENTS & REMINDERS ON THE OLSD FUNCTION		1
		2

1.2. MANUAL'S CONTENTS & REMINDERS ON THE OLSD FUNCTION

BRIEF DESCRIPTION OF THIS MANUAL'S CONTENTS

This manual presents a Screen described in and generated by the OLSD function. It is a complement to the ON-LINE SYSTEMS DEVELOPMENT (OLSD) Reference Manual, which is common to all on-line monitors.

This manual first shows the coding and then the organization of the generated programs.

The structure of a generated program is also detailed and commented upon so as to help users insert their own specific procedures that may be needed in the Screen.

It illustrates the following:

- . The coding of Data Names,
- . Descriptions of segments, screen, work areas, and communication areas,
- . A complete lexicon of variables, indexes and fields used by the automatic functions,
- . A description of the automatic functions, including their generation conditions. (Refer to Chapter "GENERATED PROGRAM: PROCEDURE DIVISION".)

NOTE: The Screen example described in this manual does not illustrate all generation possibilities provided by the OLSD function: segment accesses, cross-references between segments, access conditions, etc.

This manual does NOT contain an exhaustive presentation of the specific information on the use of the OLSD function.

INTRODUCTION	PAGE	10
MANUAL'S CONTENTS & REMINDERS ON THE OLSD FUNCTION		1
		2

REMINDERS ON THE OLSD FUNCTION

Based on the Screen descriptions, the OLSD function ensures the following:

- The automatic generation of the Screen map description from layout-type information. (Adaptation to the hardware and on-line monitor is based on an option specified at the Screen level.)
- The automatic generation of the Screen data processing from process-type information:
 - . Screen Call of Elements (-CE) -> Screen data processing
 - . Screen Call of Segments (-CS) -> External data processing
 - . Dialogue Complement (-O) and Dialogue and Screen General Documentation (-G) -> Generation Options
 - . Structured Code (-P) -> Specific processing

All processing is generated in a program structured in "Reception" and "Display", thus ensuring the complete processing of the Screen data.

The program is generated in COBOL. Adaptation to the hardware and the on-line Monitor is based on the options specified at the Screen level.

REMINDERS ON THE OLSD FUNCTION - Cont'd

It may be necessary to use complementary description lines in order to generate on-line programs:

- . Screen General Documentation (-G),
- . Screen Call of Macro-Structures (-CP),
- . Beginning Insertions (-B),
- . Screen Work Areas (-W).

SCREEN GENERAL DOCUMENTATION

The General Documentation (-G) lines of the screen or dialogue can be used to override the value of some generated constants. For more details, refer to Chapter "DESCRIPTION OF A TRANSACTION", Subchapter "SCREEN GENERAL DOCUMENTATION (-G)" in the OLSD Reference Manual.

WORK AREAS

On Work Areas (-W) screens, 'AA' is a reserved value for the CODE FOR COBOL PLACEMENT; it is used internally by the OLSD function.

The automatically generated lines are identified in the COBOL code by the '*AAnn' character string from columns 72 to 80. They can be overridden on the Work Areas (-W) screen on 'AAnn'-numbered lines.

1.3. UNISYS 2200 SCREEN GENERATION - OPERATING MODE

UNISYS 2200 SCREEN GENERATION - OPERATING MODE

Generated forms must be compiled with the FLDP utility, before the compilation of generated screens.

Generated screens include a redefinition of the table of attributes (FCA), after the call by the COPY cobol command, and before the description of screen data (DATA), also called by COPY.

The utility produces only one COPY block. That is why this block must be split into two blocks.

Example of control cards used for generated forms:

```
Lines inserted before stream: (ex: PCDM)
A . Copy a source element in a library
AED,IQ  SCREENLIB.%                                P
Lines inserted after stream: (ex: PCFM)
AEOF
A . Compilation of screen description
AFLDP,L  SCREENLIB.%,,SCREENFILE                  P
AEOF
A . Generation of standard copy block
AFLMU,G  SCREENFILE
COB
%                                                P
AEOF
A . Copy the COPY block in a cobol copy library
ACOPY,I  TPF$.SCREEN-%/COBP,COPYLIB.             P
A . Editor commands for technical status SB3
AED,UN   COPYLIB.SCREEN-%/COBP                  P
F        01  SCREEN-%-%-DATA                    SP
IB  END
IB SCREEN-%-%-DATA PROC                          SP
EXIT
A . Editor commands for technical status SB4
AED,UN   COPYLIB.SCREEN-%/COBP                  P
CH /     02  SCREEN/01  SCREEN/ALL
0
F        01  SCREEN-%-%-HEADER                   SP
IB  END
IB SCREEN-%-%-DATA PROC                          SP
EXIT
A . Create COPY entry points
APDP,C   COPYLIB.SCREEN-%/COBP                  P
AEOF
```

NOTES: Character '%' is replaced, during generation, by symbolic parameters:

- . P Form external name
- . S Screen clear name in the Database

Any text editor other than 'ED' (used in the above example) may be used, with the corresponding Find, Insert, and Change commands.

Since the system does not ensure consistency between the requests on a screen, the user must make sure that there is no generation of the FLDP and of the COBOL program associated with the screen in the same run. If this were the case, the execution of the generated program would end with the message:
ABORT DPS STATUS 015 (check-number not compatible)

2. PRESENTATION OF THE EXAMPLE

2.1. THE 'DO' DIALOGUE

```
-----  
!                APPLICATION UNISYS 2200                *PDSG.NDOC.AU1.9!  
! ON-LINE DIALOGUE DEFINITION.....: DO                !  
!                !  
! DIALOGUE NAME.....: DOCUMENTATION MANAGER            !  
!                !  
! SCREEN SIZE (LINES, COLUMNS) .....: 24      080      !  
! LABEL TYPE, TABS, INITIALIZATION...: L        01      _  !  
! HELP CHARACTER SCREEN, DATA ELEMENT: $        $      !  
!                !  
!                LABELS  DISPLAY  INPUT  ER.MESS.  ER.FLD!  
! INTENSITY ATTRIBUTE .....: N        N        N        N        N  !  
! PRESENTATION ATTRIBUTE .....: N        N        N        N        N  !  
! COLOR ATTRIBUTE .....: W        W        W        W        W  !  
!                !  
! TYPE OF COBOL AND MAP TO GENERATE...: U    0      UNISYS 2200      !  
! CONTROL CARD OPTIONS FRONT & BACK...:          (PROGRAM)  $$      (MAP)!  
! EXTERNAL NAMES .....:          (PROGRAM)      (MAP)!  
! TRANSACTION CODE.....:                !  
!                !  
!                !  
! EXPLICIT KEYWORDS...: DOC                !  
! SESSION NUMBER.....: 0010      LIBRARY.....: AU1      LOCK.....:  !  
!                !  
! O: C1 CH: Odo                ACTION:                !  
-----
```

PRESENTATION OF THE EXAMPLE
THE 'DO' DIALOGUE

PAGE

16

2
1

```
-----  
!                APPLICATION UNISYS 2200                *PDSG.NDOC.AU1.9!  
! DIALOGUE COMPLEMENT....: DO DOCUMENTATION MANAGEMENT !  
! ! ! ! !  
! COMMON AREA-DATA STRUCTURE CODE.....: CA !  
! ! ! ! !  
! ERROR MESSAGE FILE CHARACTERISTICS !  
!             ORGANIZATION....: W !  
!             EXTERNAL NAME...: EM !  
! ! ! ! !  
! FIRST SCREEN CODE OF THE DIALOGUE.....: 0060 !  
! ! ! ! !  
! COMPLEMENTARY COMMON AREA LENGTH.....: 700 !  
! ! ! ! !  
! CODE OF PSB OR SUB-SCHEMA.....: !  
! ! ! ! !  
! ! ! ! !  
! OPTIONS : OCF F10 NOSDERR !  
! ! ! ! !  
! ! ! ! !  
! SESSION NUMBER      : 0007  LIBRARY      : AU1 !  
! ! ! ! !  
! O: C1 CH: Odo O                ACTION: !  
-----
```


2.2. THE 'DO0030' SCREEN

```
-----  
!                APPLICATION UNISYS 2200                *PDSG.NDOC.AU1.9!  
! ON-LINE SCREEN DEFINITION.....: DO0030                !  
!                !  
! SCREEN NAME.....: *** ORDER INPUT SCREEN ***        !  
!                !  
! SCREEN SIZE (LINES, COLUMNS) .....: 24      080      !  
! LABEL TYPE, TABS, INITIALIZATION...: L        01      * -    !  
! HELP CHARACTER SCREEN, DATA ELEMENT: !          $      !  
!                !  
!                LABELS  DISPLAY  INPUT  ER.MESS.  ER.FLD!  
! INTENSITY ATTRIBUTE .....: N        N        N        N        N !  
! PRESENTATION ATTRIBUTE .....: N        N        N        N        N !  
! COLOR ATTRIBUTE .....: W        W        W        W        W !  
!                !  
! TYPE OF COBOL AND MAP TO GENERATE...: U  0      UNISYS 2200      !  
! CONTROL CARD OPTIONS FRONT & BACK...:          (PROGRAM)      $$      (MAP) !  
! EXTERNAL NAMES .....: DOP030  (PROGRAM)      130      (MAP) !  
! TRANSACTION CODE.....:                !  
!                !  
! EXPLICIT KEYWORDS...:                !  
! SESSION NUMBER.....: 0006      LIBRARY.....: AU1      LOCK.....:    !  
!                !  
! O: C1 CH: Odo0030                ACTION:                !  
-----
```


PRESENTATION OF THE EXAMPLE
 THE 'DO0030' SCREEN

PAGE

20

2
 2

```

-----
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!
! SCREEN CALL OF ELEM... DO0030 *** ORDER INPUT SCREEN ***                               !
!                                                                                               !
! A LIN : D.ELEM . PHYSICAL ATTRIBUTES . VALIDATION UPDATE . DISPLAY                               !
!       :      . P LN COL N L C HR VR . P V U UPD TARGET . S SOURCE LV!                               !
!-----
! . 230 : REMIS .          003 V U N .          CD05 .          CD05                               !
! . 300 : LINE . A 10 001 R 1 01 09 .          .          .          !
! . 305 : CODMVT .        003 V Y .          I .          .          !
! . 310 : FOURNI .        003 V .          R T CD00 .          CD00 !
! . 320 : QTMAC .         003 V .          R X CD10 .          CD10 !
! . 325 : . . . . .          + FO10QTMAM .          .          !
! . 330 : QTMAL .         002 F .          .          .          CD10 !
! . 335 : QTMAR .         002 F .          .          .          .          !
! . 340 : INFOR .         001 V .          P X CD10 .          CD10 !
! . 350 : END .          004 Z .          .          .          .          !
! . 400 : . . A 20 002 L .          .          .          .          !
! . 405 : EDIT .         001 V F .          I CD20 .          .          !
! . 415 : DOAC31 . A 20 001 S .          .          .          .          !
! . 500 : DOAP05 . A 22 001 S .          .          .          .          !
!       : . . . . .          .          .          .          !
!       : . . . . .          .          .          .          !
!       : . . . . .          .          .          .          !
! O: C1 CH: -CE23
-----
  
```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

PAGE

21

2
2

```
-----  
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!  
! SCREEN CALL OF ELEM... DO0030 *** ORDER INPUT SCREEN ***                               !  
!                                                                                               !  
! A LIN : D.ELEM . PHYSICAL ATTRIBUTES . LABEL .                               !  
!       : . P LN COL N L HR VR IN PR CO . T LITERALS                               !  
! .....  
!   050 : DOAC30 . A 01 001 S . . . . .                               !  
!   . 080 : DOAP04 . A 01 001 S . . . . .                               !  
!   . 100 : DO0030 . A 01 025 T . . . . .                               !  
!   . 110 : NUCOM . A 03 004 P U . . . . .                               !  
!   . 120 : MATE . . . . . 003 V U . . . . .                               !  
!   . 122 : . . . . . . . . . . . . . . .                               !  
!   . 125 : RELEA . . . . . 012 V U . . . . .                               !  
!   . 130 : NUCLIE . . . . . 01 004 O U . . . . .                               !  
!   . 140 : RAISOC . . . . . 003 P F . . . . .                               !  
!   . 145 : RUE . . . . . 01 009 V F . . . . . P 84, OLD TOWNLINE ROAD . . . . . !  
!   . 150 : COPOS . . . . . 003 V F . . . . .                               !  
!   . 155 : . . . . . . . . . . . . . . .                               !  
!   . 160 : VILLE . . . . . 003 F F . . . . .                               !  
!   . 200 : REFCLI . . . . . 01 004 V U . . . . .                               !  
!   . 210 : DATE . . . . . 003 V U . . . . . I .._... . . . . .                               !  
!   . 220 : CORRES . . . . . 01 005 V U . . . . .                               !  
!                                                                                               !  
! O: C2 CH: -CE                                                                                               !  
-----
```


PRESENTATION OF THE EXAMPLE
 THE 'DO0030' SCREEN

```

-----
!                APPLICATION UNISYS 2200                *PDSG.NDOC.AU1.10!
! ON-LINE SCREEN CALL OF SEGM. DO0030 *** ORDER INPUT SCREEN ***
! ...CA00...CD05...WP30...*CD00...*CD10...*FO10... CD20.....!
! A SEGM      :   USE PREC ACCESS KEY      ACCESS      D EXTERNAL LIB. S   : LIBR!
! C CODE C LN : G R D SEGM SOURCE          KEY      B O T NAME      SEGM N LV:   !
! . CD05  00 :   M A      SPACES           KEYCD     W   CD          CD05   12: 0007!
! . CD05  02 :           'B'              COCARA          : *DCC!
! . CD05  04 :           CA00-NUCOM        NUCOM           : *DCC!
! . CD10 R 00 :   T           'C'          KEYCD     W   CD          CD10   : 0007!
! . CD10 R 02 :           CA00-NUCOM        NUCOM           : *DCC!
! . CD10 R 04 :           0030-FOURNI      FOURNI          : *DCC!
! . CD10 R 06 :   A           SPACES       KEYCD          : *ACC!
! . CD10 R 08 :           'C'              COCARA C         : *DCC!
! . CD10 R 10 :           CA00-NUCOM        NUCOM C         : *DCC!
! . FO10 R 00 :   M N CD10 0030-FOURNI      CLEFO     W   FO          FO10   : 0007!
! . FO10 R 02 :           CA00-LANGU       LANGU           : *DCC!
! . FO10 R 04 :           0030-RELEA      RELEA           : *DCC!
! . FO10 R 06 :           0030-MATE       MATE            : *DCC!
! . CD20 Z 00 :   X N      SPACES           KEYCD     W   CD          CD20   : 0007!
! . CD20 Z 02 :           'E'              COCARA          : *DCC!
! . CD20 Z 04 :           CA00-NUCOM        NUCOM           : *DCC!
! . ME00 Z 00 :   N A      CA00-CLEME      CLEME     W   ME          ME00   : 0007!
!
! O: C1 CH: -CS
-----

```


PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

2
2

```

-----
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!
! WORK AREAS.....ENTITY TYPE O DO0030 *** ORDER INPUT SCREEN ***                               !
!                                                                                               !
! CODE FOR PLACEMENT..:          BB                                                                                               !
! A LIN T LEVEL OR SECTION WORK AREA DESCRIPTION                                         OCCURS!
! . 200 I 01                               WW10-QTMAR                                         !
! . 201                               VALUE ZERO.                                         !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
!                                                                                               !
! O: C1 CH: Odo0030W                                                                                               !
!                                                                                               !
-----

```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

```
-----  
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!  
! WORK AREAS.....ENTITY TYPE O DO0030 *** ORDER INPUT SCREEN ***                               !  
!                               !  
! CODE FOR PLACEMENT..:      WP                               !  
! A LIN T LEVEL OR SECTION WORK AREA DESCRIPTION                               OCCURS !  
! * 000  01      WP00.                               !  
! * 010  02      WP10.                               !  
! * 020  05      FILLER PIC X(25) VALUE                               !  
! * 030      "23400BRISBANE                               " . !  
! * 040  05      FILLER PIC X(25) VALUE                               !  
! * 050      "56400VICTORIA                               " . !  
! * 060  05      FILLER PIC X(25) VALUE                               !  
! * 070      "76500ALICE SPRINGS                               " . !  
! * 080  05      FILLER PIC X(25) VALUE                               !  
! * 090      "55300MELBOURNE                               " . !  
! * 100  05      FILLER PIC X(25) VALUE                               !  
! * 110      "11000CANBERRA                               " . !  
! * 120  05      FILLER PIC X(25) VALUE                               !  
! * 130      "34500PERTH                               " . !  
! * 140  05      FILLER PIC X(25) VALUE                               !  
! * 150      "85270DARWIN                               " . !  
! * 160  05      FILLER PIC X(25) VALUE                               !  
!                               !  
! O: C1 CH: Odo0030Wwp                               !  
-----
```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

PAGE

27

2
2

```
-----  
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!  
! WORK AREAS.....ENTITY TYPE O DO0030 *** ORDER INPUT SCREEN ***                               !  
!                                                                                               !  
! CODE FOR PLACEMENT..:      WP                                                                                               !  
! A LIN T LEVEL OR SECTION WORK AREA DESCRIPTION                               OCCURS !  
! * 170                               "94000HOBART                               " .                               !  
! * 180      05                               FILLER PIC X(25) VALUE                               !  
! * 190                               "89300SYDNEY                               " .                               !  
! * 300      02                               WP20 REDEFINES WP10 OCCURS 9.                               9 !  
! * 320 E      05                               WP20-COPOS .                               !  
! * 340 E      05                               WP20-VILLE .                               !  
! * 400      02                               WP30.                               !  
! * 410 I      05                               WP30-COPOS .                               !  
! * 500      02                               WP40.                               !  
! * 510 E      05                               WP40-VILLE.                               !  
! * 520 E      05                               WP40-VILLEL.                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
! O: C1 CH: Odo0030Wwp17                               !  
-----
```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

2
2

```

-----
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!
! WORK AREAS.....ENTITY TYPE O DO0030 *** ORDER INPUT SCREEN ***                               !
!                                                                                               !
! CODE FOR PLACEMENT..:          BB                                                                                               !
! A LIN T LEVEL OR SECTION WORK AREA DESCRIPTION                                         OCCURS!
! * 100  01                               7-WW00.                                                                                               !
! * 120  05                               FILLER          PICTURE X(6) VALUE 'FNPAC ' .                               !
! * 130  05                               7-WW00-FONCT   PICTURE X(6) .                               !
! * 140  05                               FILLER          PICTURE X(7) VALUE ' ORDRE ' .                               !
! * 150  05                               7-WW00-ORDRE  PICTURE X(8) .                               !
! * 160  05                               FILLER          PICTURE X(7) VALUE ' RBCOD ' .                               !
! * 170  05                               7-WW00-RBCODE PICTURE X(2) .                               !
! * 180  05                               FILLER          PICTURE X(4) VALUE ' FN ' .                               !
! * 190  05                               7-WW00-FUNCT  PICTURE X(2) .                               !
! * 200  05                               FILLER          PICTURE X(7) VALUE ' ERCOD ' .                               !
! * 210  05                               7-WW00-ERCOD  PICTURE X(2) .                               !
! * 220  05                               FILLER          PICTURE X(7) VALUE ' ERNUM ' .                               !
! * 230  05                               7-WW00-NUM    PICTURE X(4) .                               !
! * 500  01                               7-HELP-ERROR  PICTURE X(72) VALUE                               !
! * 510                               '***** HELP FUNCTION UNAVAILABLE *****' .                               !
!                                                                                               !
!                                                                                               !
! O: C1 CH: Odo0030Www
-----

```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

PAGE

29

2
2

```
-----  
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!  
! WORK AREAS.....ENTITY TYPE O DO0030 *** ORDER INPUT SCREEN ***                               !  
!                                                                                               !  
! CODE FOR PLACEMENT..:          BB                                                                                               !  
! A LIN T LEVEL OR SECTION WORK AREA DESCRIPTION                               OCCURS!  
! * 200  01          DBK-REC USAGE IS DATABASE-KEY.                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
!                                                                                               !  
! O: C1 CH: Odo0030Wwx                               !  
-----
```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

PAGE

30

2
2

```
-----  
!                APPLICATION UNISYS 2200                *PDSG.NDOC.AU1.9!  
! ON-LINE SCREEN GENERAL DOC.      DO0030 ***  ORDER INPUT SCREEN  ***  !  
!                !  
! A LIN : T COMMENT                                LIB !  
! . 020 : C      THIS SCREEN ALLOWS THE ENTRY OF AN ORDER FOR          *ACC!  
! . 030 : C      DOCUMENTATION PLACED BY A REFERENCED CLIENT.          *ACC!  
! . 050 : C      FROM THIS SCREEN, YOU MAY ACCESS ANY OTHER SCREEN OF   *ACC!  
! . 055 : C      THE DIALOG BY ENTERING THE CORRESPONDING CHOICE FIELD  *ACC!  
! . 060 : C      VALUE. THE DIFFERENT VALUES ARE DISPLAYED IN THE     *ACC!  
! . 070 : C      BOTTOM PART OF ALL THE DIALOG'S SCREENS.              *ACC!  
! . 120 : S CD05                                          *ACC!  
! . 122 : U F 8  TECHNICAL PROBLEM  CALL E.D.P. DEPT.(CODE 030-CD05 F8) *ACC!  
! . 124 : U F 9  TECHNICAL PROBLEM  CALL E.D.P. DEPT.(CODE 030-CD05 F9) *ACC!  
! . 130 : U G 9  TECHNICAL PROBLEM  CALL E.D.P. DEPT.(CODE 030-CD05 G9) *ACC!  
! . 150 : S CD10 R                                          *ACC!  
! . 152 : U F 8  INCORRECT UPDATE REQUEST.                            *ACC!  
! . 154 : U F 9  INCORRECT REQUEST FOR CREATION.                      *ACC!  
! . 160 : U G 9  END OF DISPLAY FOR THIS ORDER.                        *ACC!  
! . 180 : S ME00 Z                                          *ACC!  
! . 190 : U G 9  TECHNICAL PROBLEM  CALL E.D.P. DEPT.(CODE 030-ME00 G9) *ACC!  
! . 200 : S FO10 R                                          *ACC!  
! . 210 : U F 9  MANUAL DOES NOT BELONG TO DOCUMENTANTION.           *ACC!  
!                !  
! O: C1 CH: Odo0030G                                          !  
-----
```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

PAGE

31

2
2

```
-----  
!                               APPLICATION UNISYS 2200                               *PDSG.NDOC.AU1.9!  
! ON-LINE SCREEN GENERAL DOC.      DO0030 *** ORDER INPUT SCREEN ***                               !  
!                               !  
! A LIN : T COMMENT                                                    LIB !  
! . 350 : F CODMVT                                                    *ACC!  
! . 360 : C      AN ACTION CODE MUST BE ENTERED.                      *ACC!  
! . 400 : F FOURNI                                                    *ACC!  
! . 402 : C      THE FIELD 'ITEM' IS ENTERED WITH THE 3-CHARACTER CODE *ACC!  
! . 403 : C      OF THE MANUAL. IT IS NOT POSSIBLE TO ENTER          *ACC!  
! . 404 : C      REQUESTS CONCERNING THE BINDERS.                    *ACC!  
! . 430 : U      A THIS PROCEDURE DOES NOT PERMIT THE ORDER OF BINDERS. *ACC!  
! . 450 : F MATE                                                    *ACC!  
! . 451 : T      0 DOCUM DD                                           *ACC!  
! . 453 : U      5 THIS TYPE OF HARDWARE IS NOT SUPPORTED.          *ACC!  
! . 500 : F QTMAC                                                    *ACC!  
! . 510 : C      THE 'QUANTITY ORDERED' FIELD MUST BE ENTERED WITH THE *ACC!  
! . 520 : C      NUMBER OF COPIES NEEDED FOR THE SPECIFIED MANUAL.    *ACC!  
! . 530 : C      ACCORDING TO STOCK AVAILABILITY, THE SYSTEM FILLS IN *ACC!  
! . 540 : C      THE 'QUANTITY DELIVERED' AND, IF NEEDED, THE 'QUANTITY *ACC!  
! . 541 : C      OUTSTANDING'.                                       *ACC!  
! . 600 : F INFOR                                                    *ACC!  
! . 610 : C      THE 'REMARKS' COLUMN ALLOWS TO ENTER SPECIFICS     *ACC!  
! . 625 : C      CONCERNING THE LEAD TIMES OF OUTSTANDING ORDERS.    *ACC!  
! O: C1 CH: Odo0030G35                                             !  
-----
```

PRESENTATION OF THE EXAMPLE
 THE 'DO0030' SCREEN

FUSFLIN OPE OPERANDS		LVTY CONDITION
0115270	UPDATE	
0115300	COB GO TO F99RB	99IT ERROR-CODE NOT = ZERO
02CP	N INIT. NUMBER OF LOADED ITEMS	10BL
02CP100	M IWP20M IWP20L	
08BB	N NO UPDATE ==> END OF RECEIVE	10IT OPER NOT = 'M'
08BB100	GFT	
15AA	N INITIALIZATION CATM (HEADING)	10IT CATX = SPACE
15AA100	M 'M' CATM	AN OPER = 'M'
20BB	N ITEM NOT AVAILABLE	10*A FOURNI
20BB100	ERR A FOURNI	99IT I-0030-FOURNI = 'CLA'
20BB110	GF	AN CATM NOT = SPACE
25BB	N ACCESS TO FO10	12*P CD10
25BB100	M '1' CD10-CF	
28BH	N STOCK UPD.: ORDER DELETION/UPD	10IT (CATM = 'A' OR 'M')
28BH100	A CD10-QTMAL FO10-QTMAS	AN CATX = 'R'
28BH120		AN CAT-ER = SPACES
30BD	N QUANTITY PROCESSING	10*P R
30BF	N CALC. DELIV. QUANT. STOCK UPD.	12IT CATM = 'C' OR 'M'
30BF100	M I-0030-QTMAL CD10-QTMAL	99IT FO10-QTMAS NOT <
30BF110		I-0030-QTMAL
30BF120	M FO10-QTMAS CD10-QTMAL	99EL
30BF130	S CD10-QTMAL FO10-QTMAS	99BL
30BF140	M CD10-QTMAL O-0030-QTMAL	
4029	N END OF TRANSACTION	10IT OPER = 'E'
4029100	M '** END OF TRANSACTION **'	
4029110	END-MESSAGE	
64DA	N PREPARATION DISPLAY DATE/HOUR	10IT CATX = ' '
64DA 40	AD6	
64DA 80	AD IM DATOR DAT8C	
64DA120	TIM	99BL
64DA160	TIF TIMCOG TIMDAY	
65BB	N REMAINS TO BE DELIVERED	10*P R
65BB100	C WW10-QTMAR =	99IT CD10-QTMAL NOT = ZERO
65BB110	CD10-QTMAL - CD10-QTMAL	
65BB120	M WW10-QTMAR O-0030-QTMAR	
80BB	N SEARCH RECORD CD05	10*R CD05 R
80BB 10	YR CD05	
80BB 20	YRU CD05	
80BB100	M 'F80BB' 7-WW00-FONCT	
80BB110	M 'FETCH' 7-WW00-ORDRE	
80BB120	M CD05 DC05	
80BB130	COB FETCH DC05 RECORD	
80BB140	M DC05 CD05	
80BB150	P F98ER	99BL
80BB160	COB GO TO F80-OK	99IT IK = ZERO
80BB180	COB GO TO F80-KO	99BL
80BC	N UPDATE RECORD CD05	10*R CD05 RW
80BC 10	YRW CD05	
80BC100	M 'F80CD05' 7-WW00-FONCT	
80BC110	M 'MODIFY' 7-WW00-ORDRE	
80BC120	M CD05 DC05	
80BC130	BFD DC05 RECORD	
80BC140	P F98ER	99BL
80BC160	COB GO TO F80-KO	99IT IK NOT = ZERO
80BC200	BMD DC05 RECORD	
80BC210	P F98ER	
80BC240	COB GO TO F80-OK	99IT IK = ZERO
80BC250	COB GO TO F80-KO	99BL
80ED	N ACCESS RECORD CD20	10*R CD20 R

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

2
2

```

80ED 10 YR CD20
80ED 20 YRU CD20
80ED100 M 'F80CD20' 7-WW00-FONCT
80ED110 M 'FETCH' 7-WW00-ORDRE
80ED120 M CD20 DC20
80ED130 COB FETCH FIRST DC20
80ED140 WITHIN WWS0520 SET
80ED150 M DC20 CD20
80ED160 P F98ER
80ED170 COB GO TO F80-OK 99IT IK = ZERO
80ED180 COB GO TO F80-KO. 99BL

80EF N UPDATE RECORD CD20 10*R CD20 RW
80EF 10 YRW CD20
80EF100 M 'F80CD20' 7-WW00-FONCT
80EF110 M 'MODIFY' 7-WW00-ORDRE
80EF120 M CD20 DC20
80EF130 BFD DC20 RECORD
80EF140 P F98ER 99BL
80EF160 COB GO TO F80-KO 99IT IK NOT = ZERO
80EF200 BMD DC20 RECORD
80EF210 P F98ER
80EF240 COB GO TO F80-OK 99IT IK = ZERO
80EF250 COB GO TO F80-KO 99BL

80EG N CREATION RECORD CD20 10*R CD20 W
80EG 10 YW CD20
80EG100 M 'F80CD20' 7-WW00-FONCT
80EG110 M 'STORE' 7-WW00-ORDRE
80EG120 M CD20 DC20
80EG130 BST DC20
80EG140 P F98ER 99BL
80EG160 COB GO TO F80-OK 99IT IK = ZERO
80EG180 COB GO TO F80-KO 99BL

80FO N SEARCH RECORD FO10 10*R FO10 R
80FO 10 YR FO10
80FO 20 YRU FO10
80FO100 M 'F80FO' 7-WW00-FONCT
80FO110 M 'FETCH' 7-WW00-ORDRE
80FO120 M FO10 OF10
80FO130 COB FETCH OF10 RECORD
80FO140 M OF10 FO10
80FO150 P F98ER 99BL
80FO160 COB GO TO F80-OK 99IT IK = ZERO
80FO180 COB GO TO F80-KO 99BL

80FP N UPDATE RECORD FO10 10*R FO10 RW
80FP 10 YRW FO10
80FP100 M 'F80FO10' 7-WW00-FONCT
80FP110 M 'MODIFY' 7-WW00-ORDRE
80FP120 M FO10 OF10
80FP130 BFD OF10 RECORD
80FP140 P F98ER 99BL
80FP160 COB GO TO F80-KO 99IT IK NOT = ZERO
80FP200 BMD OF10 RECORD
80FP210 P F98ER
80FP240 COB GO TO F80-OK 99IT IK = ZERO
80FP250 COB GO TO F80-KO 99BL

80LI N FETCH CD10 VIA ACCESS 10*R CD10 P
80LI 5 YP CD10
80LI 10 YR CD10
80LI 20 YRU CD10
80LI100 M 'F80CD10' 7-WW00-FONCT
80LI110 M 'FETCHVIA' 7-WW00-ORDRE
80LI120 M CD10 DC10
80LI130 COB FETCH DC10 VIA WW0510
80LI135 USING CD10-FOURNI
80LI140 M DC10 FOURNI
80LI150 P F98ER 99BL
80LI160 COB GO TO F80-OK 99IT IK = ZERO
80LI180 COB GO TO F80-KO 99BL

80LJ N UPDATE RECORD CD10 10*R CD10 RW
80LJ 10 YRW CD10
80LJ100 M 'F80CD10' 7-WW00-FONCT

```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

2
2

```

80LJ110 M 'MODIFY' 7-WW00-ORDRE
80LJ120 M CD10 DC10
80LJ130 BFD DC10 RECORD
80LJ140 P F98ER 99BL
80LJ160 COB GO TO F80-KO 99IT IK NOT = ZERO
80LJ200 BMD DC10 RECORD
80LJ210 P F98ER
80LJ240 COB GO TO F80-OK 99IT IK = ZERO
80LJ250 COB GO TO F80-KO 99BL

80LM N CREATION RECORD CD10 10*R CD10 W
80LM 10 YW CD10
80LM100 M 'F80CD10' 7-WW00-FONCT
80LM110 M 'STORE' 7-WW00-ORDRE
80LM120 M CD10 DC10
80LM130 BST DC10
80LM140 P F98ER 99BL
80LM160 COB GO TO F80-OK 99IT IK = ZERO
80LM180 COB GO TO F80-KO 99BL

80LN N READ NEXT SEGMENT CD10 10*R CD10 RN
80LN 10 YRN CD10
80LN100 M 'F80CD10' 7-WW00-FONCT
80LN110 M 'FETCH' 7-WW00-ORDRE
80LN120 M CD10 DC10
80LN130 COB FETCH NEXT DC10 WITHIN WWS0510 S
80LN140 M DC10 CD10
80LN150 P F98ER 99BL
80LN160 COB GO TO F80-OK 99IT IK = ZERO
80LN180 COB GO TO F80-KO. 99BL

80LQ N DELETE RECORD CD10 10*R CD10 D
80LQ 10 YD CD10
80LQ100 M 'F80LQ' 7-WW00-FONCT
80LQ110 M 'DELETE' 7-WW00-ORDRE
80LQ120 M CD10 DC10
80LQ130 BFD DC10 RECORD
80LQ140 P F98ER 99BL
80LQ160 COB GO TO F80-KO 99IT IK NOT = ZERO
80LQ200 COB DELETE DC10 RECORD 99BL
80LQ210 P F98ER
80LQ220 COB GO TO F80-OK 99IT IK = ZERO
80LQ230 COB GO TO F80-KO 99BL

80ME N SEARCH RECORD ME00 10*R ME00 R
80ME 10 YR ME00
80ME 20 YRU ME00
80ME100 M 'F80ME' 7-WW00-FONCT
80ME110 M 'FETCH' 7-WW00-ORDRE
80ME120 M ME00 MM00
80ME130 COB FETCH MM00 RECORD
80ME140 M MM00 ME00
80ME150 P F98ER 99BL
80ME160 COB GO TO F80-OK 99IT IK = ZERO
80ME180 COB GO TO F80-KO 99BL

8095 N SAVE FOR HELP SCREEN 10*R HELP
8095 10 YR HELP
8095100 M 'F8095' 7-WW00-FONCT
8095120 M '$RELEASE' 7-WW00-ORDRE
8095130 CAL 'D$RELEASE' USING STATUS-WORD
8095150 COB GO TO F81ER 99IT STATUS-FATAL
8095160 COB GO TO F80-OK. 99BL
8095210 YRW HELP
8095300 M 'F8095' 7-WW00-FONCT
8095320 M '$STORE' 7-WW00-ORDRE
8095330 CAL 'D$STORE' USING STATUS-WORD
8095340 SCREEN-DO0030-32
8095350 COB GO TO F81ER. 99IT STATUS-FATAL
8095360 COB GO TO F80-OK. 99BL
8095400 YW HELP
8095410 COB GO TO F80-OK.
8095450 YD HELP
8095460 COB GO TO F80-OK.

8098 N ERROR MESSAGE FILE ACCESS 10*R EM00
8098 10 YR EM00

```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

2
2

```

8098100 M 'F80EM00' 7-WW00-FONCT
8098120 M 'FETCH' 7-WW00-ORDRE
8098130 M EM00-EMKEY ER00-EMKEY
8098150 COB FETCH ER00 RECORD
8098160 P F98ER 99BL
8098200 M ER00 EM00 99IT IK = ZERO
8098220 COB GO TO F80-OK. AN ER00-ERKEY = EM00-EMKE
8098250 COB GO TO F80-KO 99BL

81ER N DPS ERROR 10BL
81ER100 P F81FI
81ER110 M STATUS-FONCTION 7-WW00-FUNCT
81ER120 M STATUS-CODE 7-WW00-ERCOD
81ER140 M 7-WW00 END-MESSAGE

81ES N ERROR DPS MANAGEMENT 15BL
81ES 10 * FUNCTION KEY MSG-WAIT
81ES100 CAL 'D$RESET' USING STATUS-WORD 99IT STATUS-FUNCTION = 05
81ES110 AN (STATUS-CODE = 31 OR 3
81ES200 * HELP FUNCTION NOT AVAILABLE 99IT STATUS-FUNCTION = 06
81ES210 M 7-HELP-ERROR ERROR-MESSAGE AN (STATUS-CODE = 43 OR 4
81ES240 CAL 'D$SENDERR' USING STATUS-WORD
81ES250 ERROR-MESSAGE ERROR-COORDINATES
81ES300 GT 15 99IT STATUS-FATAL
81ES400 COB GO TO F8Z20. 99BL

81EV N DISPLAY DPS ERROR 15BL
81EV100 MES '***** DPS ERROR *****'
81EV110 UPON PRINTER
81EV120 MES 'PROGRAM : ' PROGR
81EV130 UPON PRINTER
81EV140 MES 'FUNCT. PACBASE : '
81EV150 7-WW00-FONCT
81EV160 UPON PRINTER
81EV170 MES 'DPS ORDER : '
81EV180 7-WW00-ORDRE
81EV190 UPON PRINTER
81EV200 MES 'STATUS-FUNCTION : '
81EV210 STATUS-FUNCTION
81EV220 UPON PRINTER
81EV230 MES 'STATUS-CODE : '
81EV240 STATUS-CODE
81EV250 UPON PRINTER
81EV300 COB DEPART WITH ROLLBACK 99IT IMPART-DEPART = '1'
81EV310 CAL 'D$CLCONV' USING STATUS-WORD 99BL
81EV320 CAL 'D$ERRMSG' USING STATUS-WORD
81EV340 M SPACE COMMON-AREA
81EV360 CAL 'D$PUTSCR' USING STATUS-WORD
81EV370 COMMON-AREA
81EV400 CAL 'D$TERM' USING STATUS-WORD

81FI N CLOSE DATABASE 10BL
81FI100 COB CLOSE ALL ON ERROR GO TO F99RB.
81FI200 COB DEPART ON ERROR GO TO F99RB.

93CP N ZIP CODE VALIDATION 10BL
93CP100 SCH WP20-COPOS WP30-COPOS
93CP200 M '5' DEL-ER 99IT IWP20R > IWP20L
93CP220 GT 10

98ER N DMS ERROR 10BL
98ER100 M ZERO IK 99IT ERROR-STATUS = ZERO
98ER110 GT 10
98ER120 M '1' IK 99IT ERROR-CODE = '05'
98ER130 GT 10 AN ERROR-FUNCTION = '12'
98ER140 M '2' IK 99IT ERROR-CODE = '06'
98ER150 GT 10 AN ERROR-FUNCTION = '03'
98ER160 M '3' IK 99IT (ERROR-CODE = '07' OR
98ER170 GT 10 AN ERROR-FUNCTION = '03'
98ER180 M '4' IK 99IT ERROR-CODE = '15'
98ER190 GT 10 AN ERROR-FUNCTION = '02'
98ER200 M '5' IK

99RB N ROLL-BACK ERROR 10BL
99RB100 M RB-ERROR-CODE 7-WW00-RBCODE
99RB110 M ERROR-FUNCTION 7-WW00-FUNCT
99RB120 M ERROR-CODE 7-WW00-ERCOD

```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

2
2

```

99RB130 M   ERROR-NUM          7-WW00-NUM
99RB140 M   7-WW00  ERROR-MESSAGE
99RB200 *   ---->    DISPLAY DMS ERROR  <----
99RB210 MES '*****  DMS      ERROR  *****'
99RB220      UPON PRINTER
99RB300 MES 'PROGRAM          : ' PROGE
99RB310      UPON PRINTER
99RB320 MES 'ERROR-STATUS    : ' ERROR-STATUS
99RB330      UPON PRINTER
99RB340 MES 'ERROR-NUM       : ' ERROR-NUM
99RB350      UPON PRINTER
99RB360 MES 'ERROR-AREA     : ' ERROR-AREA
99RB370      UPON PRINTER
99RB380 MES 'ERROR-RECORD   : ' ERROR-RECORD
99RB390      UPON PRINTER
99RB400 MES 'ERROR-SET      : ' ERROR-SET
99RB410      UPON PRINTER
99RB420 MES 'IK OPER CATX CATM ICATR ICF '
99RB425      'OCF I-PFKEY'
99RB430      UPON PRINTER
99RB440 MES ' ' IK ' ' OPER ' ' CATX
99RB445      ' ' CATM' ' ICATR
99RB447      ICF ' ' OCF ' ' I-PFKEY
99RB450      UPON PRINTER
99RB490 *   ---->          ROLLBACK          <----
99RB500 COB DEPART WITH ROLLBACK          99IT IMPART-DEPART = '1'
99RB510 CAL 'D$CLCONV' USING STATUS-WORD  99BL
99RB520 CAL 'D$USERMSG' USING STATUS-WORD
99RB530      ERROR-MESSAGE
99RB540 M   SPACE  COMMON-AREA
99RB550 CAL 'D$PUTSCR' USING STATUS-WORD
99RB560      COMMON-AREA
99RB580 CAL 'D$TERM'   USING STATUS-WORD

```

PRESENTATION OF THE EXAMPLE
THE 'DO0030' SCREEN

```
-----  
!   XXXXXXX - 0808      *** ORDER INPUT SCREEN ***      ***** 14:45:36 !  
!   !  
!   ORDER NUMBER: 02345  SYSTEM: UNISYS 2              RELEASE: 8.5      !  
!   CUST.      BEST      D.P. MANAGEMENT              !  
!   84, OLD TOWNLINE ROAD      48016 CINCINNATI      !  
!   CUST. REF.: AS-SG  ORDER NUMBER: 05179  ORDER DATE. ..__.. !  
!   COORDINATOR: MR. GUY DANCE      DISCOUNT RATE: 12.25      !  
!   !  
!   A  ITEM      ORDERED  DELIV.  OUTST.  REMARKS      !  
!   C  DLG      3        1        2        REST TO BE DELIVERED : 01/05/96 !  
!   .  ...      ..        ..        ..        ..... !  
!   .  ...      ..        ..        ..        ..... !  
!   .  ...      ..        ..        ..        ..... !  
!   .  ...      ..        ..        ..        ..... !  
!   .  ...      ..        ..        ..        ..... !  
!   .  ...      ..        ..        ..        ..... !  
!   .  ...      ..        ..        ..        ..... !  
!   .  ...      ..        ..        ..        ..... !  
!   !  
!   PRINTING OF FORM : 0 CHOICE: _ UPD : '7', ORDERS (NEXT) : '8', !  
!   MENU : '1', CUSTOMER LIST : '2', CUST. HIST : '3', ORDER LIST : '4', !  
!   SCREEN DOC : '=' , DATA EL. DOC : '$' !  
!   PLEASE CHECK YOUR MAILBOX, THANK YOU. !  
!   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX !  
-----
```

3. GENERATED PROGRAM: DATA DIVISION

3.1. BEGINNING OF PROGRAM

BEGINNING OF PROGRAM

The user cannot modify the IDENTIFICATION DIVISION of the generated program.

The ENVIRONMENT DIVISION is automatically adapted to the variant requested for the program.

The clause 'DECIMAL POINT IS COMMA' is generated if, on the Library Definition screen, the value in the DECIMAL POINT PRESENTATION CHARACTER field is a comma (,).

All other clauses that may be necessary in this part of the program are the user's responsibility.

All modifications to this part of the program must be done on the Beginning Insertions (-B) screen. (See the STRUCTURED CODE Reference Manual).

GENERATED PROGRAM: DATA DIVISION
BEGINNING OF PROGRAM

PAGE

40

3
1

IDENTIFICATION DIVISION.	
PROGRAM-ID. DOP0030.	DO0030
AUTHOR. *** ORDER INPUT SCREEN ***.	DO0030
DATE-COMPILED. 04/08/94.	DO0030
ENVIRONMENT DIVISION.	DO0030
CONFIGURATION SECTION.	DO0030
SOURCE-COMPUTER. UNIVAC-1100-80.	DO0030
OBJECT-COMPUTER. UNIVAC-1100-80.	DO0030
SPECIAL-NAMES.	DO0030
DECIMAL-POINT IS COMMA.	DO0030
INPUT-OUTPUT SECTION.	DO0030
FILE-CONTROL.	DO0030
DATA DIVISION.	D100
SUBSCHEMA SECTION.	D120
INVOKE SUBSCHEMA SSWDO	D140
IN FILE SCH OF SCHEMA WWDO	D150
SAVE DATA INCLUDES RUN-UNIT QUICK-BEFORE-LOOKS	D170
DMCA AND RUN-UNIT-STATISTICS ARE WORKING	D180
ROLLBACK F99RB.	D190
FILE SECTION.	DO0030

3.2. BEGINNING OF WORKING-STORAGE

BEGINNING OF WORKING-STORAGE

The 'LIST OF REFERENCED ENTITIES' at the beginning of the WORKING-STORAGE SECTION is printed when the REFER option on the Dialogue Complement (-O) screen is selected.

The 'WSS-BEGIN' level is generated at the beginning of the WORKING-STORAGE SECTION for all programs.

It contains all the variables and keys necessary for automatic processing.

IK Error indicator for file accesses.

'0' No error.
'1' Error.

OPER Operation code.

'A' Display.
'M' Update.
'S' Screen continuation.
'E' End.
'P' Previous display.
'O' Transfer to another screen.

OPERD Operation code for deferred branching.

Transferred to OPER in F40.

'O' Deferred call of another screen.

OPER and OPERD: If they correspond to a Data Element defined as an Operation Code on the Screen Call of Elements (-CE) screen (value 'O' in the VALIDATION CONDITIONS/SET VARIABLES field), they are processed in the F0520 function. If not, they are processed in the F20 function.

CATX Code of the category being executed.

'0' Beginning of reception or display.
' ' Screen-top.
'R' Repetitive.
'Z' Screen-bottom.

CATM Transaction code.

GENERATED PROGRAM: DATA DIVISION
BEGINNING OF WORKING-STORAGE

PAGE

42

3

2

'C' Creation.
'M' Modification.
'A' Deletion.
'X' Implicit update.

ICATR Indicator for current category being processed.

(Repetitive category only)

SCR-ER Screen error indicator.

'1' no error.
'4' error.

FT End of repetitive category indicator.

'0' Lines to display.
'1' No more lines to display.

ICF Input Configuration.

'1' Screen in input.
'0' No screen in input.

OCF Output Configuration.

'1' Screen in output.
'0' No screen in output.

CAT-ER Ongoing error indicator for current category.

' ' No error.
'E' Error.

INA Number of Data Elements in the screen-top category.

INR INA + Number of Data Elements in the repetitive category.

INZ INR + Number of Data Elements in the screen-bottom category.

IRR Number of repetitions in the repetitive category.

INT Number of input fields.

IER Number of error messages on the screen.

DEL-ER Memorizes Data Element error (work variable).

The 'CONSTANTS' level is also generated for all programs. It contains:

- . The compilation date of the on-line generator (PACE30 and PACE80), as well as the date of the related skeleton (these appear as comment lines),
- . Information on the program and work areas generated according to the procedures executed in the program:

SESSI Session number of the generated program.
LIBRA Code of the library.
DATGN Generated program date.
PROGR System program code.
PROGE COBOL program-id.
TIMGN Generated program time.
USERCO User code.
COBASE Database code.

If a documentation help character has been entered on the Screen Definition screen, the fields below are generated:

PRDOC External name of the "HELP SCREEN" program.

5-0030-PROGE Field containing the name of the called program.
This field is set during screen branching.

The data element code PROGE and the prefix '5-' are fixed, '0030' is the Screen code within the Dialogue.

DATCE This field includes the CENTUR field (containing the value of the current century) and a blank date area (DATOR) in which the user can store the processing date in a year-month-day format (DATOA-DATOM-DATOJ).

DAT6 Fields for date formatting (MMDDYY or DDMMYY) and

DAT7 printing (for example DD/MM/YY).

DAT8 These fields are generated if a date processing operator is used in the '-P' lines of the program or if a variable data element ('V') has a date format.

DATSEP This field contains the separator used for dates. The default value ('/') can be modified by via Procedural Code (-P) lines.

DATSET This field contains the separator used for the Gregorian date.

The default value ('-') can be modified via Procedural Code (-P) lines.

DATCTY Field for century loading.

DAT6C Field for non-formatted date with century.

DAT7C Field for non-formatted date with century.

DAT8C Field for formatted date with century (DD/MM/CCYY).

DAT8G Field for the Gregorian type of date -- with century also -- (CCYY-MM-DD).

TIMCO Field for time loading.

TIMDAY Field for time formatting (HH:MM:SS).

```
WORKING-STORAGE SECTION.                                DO0030
01  WSS-BEGIN.                                          DO0030
    05 FILLER PICTURE X(7) VALUE 'WORKING'.            DO0030
    05 IK      PICTURE X.                                DO0030
    05 BLANC  PICTURE X VALUE SPACE.                    DO0030
    05 OPER  PICTURE X.                                  DO0030
    05 OPERD PICTURE X VALUE SPACE.                    DO0030
    05 CATX  PICTURE X.                                  DO0030
    05 CATM  PICTURE X.                                  DO0030
    05 ICATR PICTURE 99.                                DO0030
    05 SCR-ER PICTURE X.                                DO0030
    05 FT    PICTURE X.                                  DO0030
    05 ICF   PICTURE X.                                  DO0030
    05 OCF   PICTURE X.                                  DO0030
    05 CAT-ER PICTURE X.                                DO0030
    05 I-PFKEY.                                         DO0030
    10 I-FONCT PICTURE 99 VALUE ZERO.                   DO0030
    05 INA   PICTURE 999 VALUE 009.                     DO0030
    05 INR   PICTURE 999 VALUE 013.                     DO0030
    05 INZ   PICTURE 999 VALUE 014.                     DO0030
    05 IRR   PICTURE 99 VALUE 09.                       DO0030
    05 INT   PICTURE 999 VALUE 046.                     DO0030
    05 IER   PICTURE 99 VALUE 01.                       DO0030
    05 DEL-ER PICTURE X.                                DO0030
01  PACBASE-CONSTANTS.                                  DO0030
*  OLSD DATES PACE30 : 28/10/93                         DO0030
*  PACE80 : 04/01/94   PAC7SG : 931207                 DO0030
    05 SESSI PICTURE X(5) VALUE '0385 '.               DO0030
    05 LIBRA PICTURE X(3) VALUE 'AU1'.                 DO0030
    05 DATGN PICTURE X(8) VALUE '04/08/94'.           DO0030
    05 PROGR PICTURE X(6) VALUE 'DO0030'.             DO0030
    05 PROGE PICTURE X(8) VALUE 'DO0030 '.           DO0030
    05 TIMGN PICTURE X(8) VALUE '15:55:36'.           DO0030
    05 USERCO PICTURE X(8) VALUE 'PDSG '.            DO0030
    05 PRDOC PICTURE X(8) VALUE 'WWD050'.            DO0030
    05      5-0030-PROGE PICTURE X(8).                DO0030
01  DATCE.                                              DO0030
    05 CENTUR PICTURE XX VALUE '19'.                   DO0030
    05 DATOR.                                          DO0030
    10 DATOA PICTURE XX.                                DO0030
    10 DATOM PICTURE XX.                                DO0030
    10 DATOJ PICTURE XX.                                DO0030
01  DAT6.                                               DO0030
    10 DAT61.                                          DO0030
    15 DAT619 PICTURE 99.                               DO0030
    10 DAT62.                                          DO0030
    15 DAT629 PICTURE 99.                               DO0030
    10 DAT63 PICTURE XX.                                DO0030
01  DAT7.                                               DO0030
    10 DAT71 PICTURE XX.                                DO0030
    10 DAT72 PICTURE XX.                                DO0030
    10 DAT73 PICTURE XX.                                DO0030
01  DAT8.                                               DO0030
    10 DAT81 PICTURE XX.                                DO0030
    10 DAT8S1 PICTURE X.                                DO0030
    10 DAT82 PICTURE XX.                                DO0030
    10 DAT8S2 PICTURE X.                                DO0030
    10 DAT83 PICTURE XX.                                DO0030
01  DATSEP PICTURE X VALUE '/'.                        DO0030
01  DATSET PICTURE X VALUE '-'.                        DO0030
01  DATCTY.                                            DO0030
    05 DATCTY9 PICTURE 99.                              DO0030
01  DAT6C.                                              DO0030
    10 DAT61C PICTURE XX.                               DO0030
    10 DAT62C PICTURE XX.                               DO0030
    10 DAT63C PICTURE XX.                               DO0030
    10 DAT64C PICTURE XX.                               DO0030
01  DAT7C.                                              DO0030
    10 DAT71C PICTURE XX.                               DO0030
    10 DAT72C PICTURE XX.                               DO0030
    10 DAT73C PICTURE XX.                               DO0030
    10 DAT74C PICTURE XX.                               DO0030
01  DAT8C.                                              DO0030
    10 DAT81C PICTURE XX.                               DO0030
    10 DAT8S1C PICTURE X VALUE '/'.                   DO0030
    10 DAT82C PICTURE XX.                               DO0030
    10 DAT8S2C PICTURE X VALUE '/'.                   DO0030
```

GENERATED PROGRAM: DATA DIVISION
BEGINNING OF WORKING-STORAGE

PAGE

47

3
2

10	DAT83C	PICTURE XX.		DO0030
10	DAT84C	PICTURE XX.		DO0030
01	DAT8G.			DO0030
10	DAT81G	PICTURE XX.		DO0030
10	DAT82G	PICTURE XX.		DO0030
10	DAT8S1G	PICTURE X	VALUE '-'	DO0030
10	DAT83G	PICTURE XX.		DO0030
10	DAT8S2G	PICTURE X	VALUE '-'	DO0030
10	DAT84G	PICTURE XX.		DO0030
01	TIMCO.			DO0030
02	TIMCOG.			DO0030
05	TIMCOH	PICTURE XX.		DO0030
05	TIMCOM	PICTURE XX.		DO0030
05	TIMCOS	PICTURE XX.		DO0030
02	TIMCOC	PICTURE XX.		DO0030
01	TIMDAY.			DO0030
05	TIMHOU	PICTURE XX.		DO0030
05	TIMS1	PICTURE X	VALUE ':'	DO0030
05	TIMMIN	PICTURE XX.		DO0030
05	TIMS2	PICTURE X	VALUE ':'	DO0030
05	TIMSEC	PICTURE XX.		DO0030

3.3. SEGMENT DESCRIPTION

SEGMENT DESCRIPTION

This part of the program is generated when a Segment is used in the screen. It is generated in the WORKING-STORAGE SECTION.

The 'CONFIGURATIONS' level contains a ddss-CF variable (where 'ddss'= the segment code in the generated program) per segment accessed in the program. This permits the access to each segment to be conditioned in the processing.

The Segment DESCRIPTION TYPE is defined by the user on the segment call line:

- Complete segment (common part and specific part in redefinition)
- Specific part only
- Complete segment with a variable length (common part and specific part in redefinition without FILLER).

GENERATED PROGRAM: DATA DIVISION

3

SEGMENT DESCRIPTION

3

01	CONFIGURATIONS.		DO0030
05	CD05-CF	PICTURE X.	DO0030
05	CD20-CF	PICTURE X.	DO0030
05	CD10-CF	PICTURE X.	DO0030
05	FO10-CF	PICTURE X.	DO0030
05	ME00-CF	PICTURE X.	DO0030
01	CD00.		DO0030
10	CD00-KEYCD.		DO0030
15	CD00-COCARA	PICTURE X.	DO0030
15	CD00-NUCOM	PICTURE 9(5).	DO0030
15	CD00-FOURNI	PICTURE X(3).	DO0030
10	CD00-SUITE.		DO0030
15	FILLER	PICTURE X(00157).	DO0030
01	CD05	REDEFINES CD00.	DO0030
10	FILLER	PICTURE X(00009).	DO0030
10	CD05-NUCLIE	PICTURE 9(8).	DO0030
10	CD05-DATE	PICTURE X(6).	DO0030
10	CD05-RELEA	PICTURE X(3).	DO0030
10	CD05-REFCLI	PICTURE X(30).	DO0030
10	CD05-RUE	PICTURE X(40).	DO0030
10	CD05-COPOS	PICTURE X(5).	DO0030
10	CD05-VILLE	PICTURE X(20).	DO0030
10	CD05-CORRES	PICTURE X(25).	DO0030
10	CD05-REMIS	PICTURE S9(4)V99.	DO0030
10	CD05-MATE	PICTURE X(8).	DO0030
10	CD05-LANGU	PICTURE X.	DO0030
10	FILLER	PICTURE X(5).	DO0030
01	CD10	REDEFINES CD00.	DO0030
10	FILLER	PICTURE X(00009).	DO0030
10	CD10-QTMAL	PICTURE 99.	DO0030
10	CD10-QTMAL	PICTURE 99.	DO0030
10	CD10-INFOR	PICTURE X(35).	DO0030
10	CD10-ADFOU	PICTURE X(100).	DO0030
10	FILLER	PICTURE X(00018).	DO0030
01	CD20	REDEFINES CD00.	DO0030
10	FILLER	PICTURE X(00009).	DO0030
10	CD20-EDIT	PICTURE X.	DO0030
10	FILLER	PICTURE X(00156).	DO0030
01	FO00.		DO0030
02	FO00-00.		DO0030
02	FO10.		DO0030
10	FO10-CLEFO.		DO0030
15	FO10-FOURNI	PICTURE X(3).	DO0030
15	FO10-MATE	PICTURE X(8).	DO0030
15	FO10-RELEA	PICTURE X(3).	DO0030
15	FO10-LANGU	PICTURE X.	DO0030
10	FO10-QTMAS	PICTURE S9(4)	DO0030
	COMPUTATIONAL.		DO0030
10	FO10-QTMAM	PICTURE 9(4).	DO0030
10	FO10-LIBFO	PICTURE X(20).	DO0030
10	FO10-DATE	PICTURE X(6).	DO0030
10	FO10-HEURE	PICTURE X(8).	DO0030
10	FILLER	PICTURE XX.	DO0030
01	ME00.		DO0030
10	ME00-CLEME.		DO0030
15	ME00-COPERS	PICTURE X(5).	DO0030
15	ME00-NUMORD	PICTURE XX.	DO0030
10	ME00-MESSA	PICTURE X(75).	DO0030

3.4. FORM DESCRIPTION

SCREEN FORM DESCRIPTION

See Chapter "INTRODUCTION", Subchapter "SCREEN GENERATION - OPERATING MODE", for important information regarding the implementation of screen generation.

The structure of the procedure generated by DPS contains two 01 levels:

- 01 SCREEN-DO0030-130-FCA (Attributes)
- 01 SCREEN-DO0030-130-DATA (Input-Output Message)

130 = Form (map) external name in this example

The screen fields are coded according to the rules illustrated in the example:

```
.I-0030      Screen in reception.  
.O-0030      Screen in display.  
  
.I-0030-REMIS Reception field.  
.E-0030-REMIS Alphanumeric definition of an I-0030-REMIS  
              field which is numeric in reception.  
  
.O-0030-QTMAC Display field.  
.F-0030-QTMAC Alphanumeric definition of a O-0030-QTMAC  
              field which is numeric in display.
```

A field's attributes are described by the following sequence:

```
.FILLER PICTURE XX.  
.S-QTMAL-YCO PICTURE 9(2) COMP.  
.S-QTMAL-XCO PICTURE 9(2) COMP.  
.FILLER PICTURE X(5)  
.S-QTMAL-DYN PICTURE X  
.S-QTMAL-BACK PICTURE X  
.S-QTMAL-FORE PICTURE X  
.S-QTMAL-INT PICTURE X  
.S-QTMAL-HIGH PICTURE X.  
.FILLER PICTURE X.  
.S-QTMAL-EMPH PICTURE X.
```

The Data Element defining the repetitive category is coded in the screen map description as follows:

.J-0030-LINE OCCURS 10 in reception,
.P-0030-LINE OCCURS 10 in display,

and contains a FILLER.

The description of the fields of the data element which defines the repetitive category is generated outside of the screen description.

This description is made up of a 'FILLER' field which is filled in with each occurrence of the category. It is used to execute the procedures for each of the elementary data elements.

This description is generated according to the same rules as above, for example:

.I-0030-LINE Used for procedures in reception,
containing:
.I-0030-FOURNI
.E-0030-QTMAL
etc.

.O-0030-LINE Used for procedures in display,
containing:
.O-0030-FOURNI
.O-0030-QTMAC

An ordinary repetitive data element (which does not define a repetitive category) is described directly in the screen description as follows:

.05 FILLER OCCURS 2.
.10 I-0030-LREF1 in reception

.05 FILLER OCCURS 2.
.10 O-0030-LREF1 in display

In this case, the procedures for each occurrence of the data element are not generated and are to be inserted by the user via Structured Code (validations, transfers, etc.).

The formats used in the generated programs correspond to the following rules:

DATA ELEMENT WITH NATURE = 'F' or 'P'

Whether reception or display screen, the format is the internal format of the data element.

DATA ELEMENT WITH NATURE = 'V'

Reception screen:

- . The format is the internal format of the data element.

Display screen:

- . For alphanumeric data elements, it is the extended internal format of the data element,
- . For numeric data elements, it is a print format based on the internal format, with replacement of non-significant leading zeros with spaces.

DATA ELEMENT WITH A CONVERSATIONAL FORMAT

(See the SPECIFICATIONS DICTIONARY Reference Manual, Chapter "DATA ELEMENTS", Subchapter "DESCRIPTION SCREEN").

Reception screen:

- . The internal format is based on the conversational format entered on the Data Element (-D) Description screen.

EXAMPLE :	Conversational format	ZZZ99.99
	Internal format	9(5)v9(2)

Display screen:

- . The format is the conversational format of the data element entered on the Data Element (-D) Description screen.

GENERATED PROGRAM: DATA DIVISION
FORM DESCRIPTION

PAGE

53

3
4

	COPY SCREEN-DO0030-130.	*AA040
01	FIELD-CONTROL-TABLE REDEFINES	*AA040
	SCREEN-DO0030-130-FCA.	*AA040
	FILLER PICTURE XX.	*AA040
05	S-PROGE-YCO PICTURE 9(2) COMP.	*AA040
05	S-PROGE-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-PROGE-DYN PICTURE X.	*AA040
05	S-PROGE-BACK PICTURE X.	*AA040
05	S-PROGE-FORE PICTURE X.	*AA040
05	S-PROGE-INT PICTURE X.	*AA040
05	S-PROGE-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-PROGE-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-SESSI-YCO PICTURE 9(2) COMP.	*AA040
05	S-SESSI-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-SESSI-DYN PICTURE X.	*AA040
05	S-SESSI-BACK PICTURE X.	*AA040
05	S-SESSI-FORE PICTURE X.	*AA040
05	S-SESSI-INT PICTURE X.	*AA040
05	S-SESSI-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-SESSI-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-DATEM-YCO PICTURE 9(2) COMP.	*AA040
05	S-DATEM-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-DATEM-DYN PICTURE X.	*AA040
05	S-DATEM-BACK PICTURE X.	*AA040
05	S-DATEM-FORE PICTURE X.	*AA040
05	S-DATEM-INT PICTURE X.	*AA040
05	S-DATEM-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-DATEM-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-HEURE-YCO PICTURE 9(2) COMP.	*AA040
05	S-HEURE-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-HEURE-DYN PICTURE X.	*AA040
05	S-HEURE-BACK PICTURE X.	*AA040
05	S-HEURE-FORE PICTURE X.	*AA040
05	S-HEURE-INT PICTURE X.	*AA040
05	S-HEURE-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-HEURE-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-NUCOM-YCO PICTURE 9(2) COMP.	*AA040
05	S-NUCOM-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-NUCOM-DYN PICTURE X.	*AA040
05	S-NUCOM-BACK PICTURE X.	*AA040
05	S-NUCOM-FORE PICTURE X.	*AA040
05	S-NUCOM-INT PICTURE X.	*AA040
05	S-NUCOM-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-NUCOM-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-MATE-YCO PICTURE 9(2) COMP.	*AA040
05	S-MATE-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-MATE-DYN PICTURE X.	*AA040
05	S-MATE-BACK PICTURE X.	*AA040
05	S-MATE-FORE PICTURE X.	*AA040
05	S-MATE-INT PICTURE X.	*AA040
05	S-MATE-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-MATE-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-RELEA-YCO PICTURE 9(2) COMP.	*AA040
05	S-RELEA-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-RELEA-DYN PICTURE X.	*AA040
05	S-RELEA-BACK PICTURE X.	*AA040
05	S-RELEA-FORE PICTURE X.	*AA040
05	S-RELEA-INT PICTURE X.	*AA040

GENERATED PROGRAM: DATA DIVISION
FORM DESCRIPTION

PAGE

54

3
4

05	S-RELEA-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-RELEA-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-RAISOC-YCO PICTURE 9(2) COMP.	*AA040
05	S-RAISOC-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-RAISOC-DYN PICTURE X.	*AA040
05	S-RAISOC-BACK PICTURE X.	*AA040
05	S-RAISOC-FORE PICTURE X.	*AA040
05	S-RAISOC-INT PICTURE X.	*AA040
05	S-RAISOC-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-RAISOC-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-RUE-YCO PICTURE 9(2) COMP.	*AA040
05	S-RUE-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-RUE-DYN PICTURE X.	*AA040
05	S-RUE-BACK PICTURE X.	*AA040
05	S-RUE-FORE PICTURE X.	*AA040
05	S-RUE-INT PICTURE X.	*AA040
05	S-RUE-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-RUE-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-VILLE-YCO PICTURE 9(2) COMP.	*AA040
05	S-VILLE-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-VILLE-DYN PICTURE X.	*AA040
05	S-VILLE-BACK PICTURE X.	*AA040
05	S-VILLE-FORE PICTURE X.	*AA040
05	S-VILLE-INT PICTURE X.	*AA040
05	S-VILLE-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-VILLE-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-COPOS-YCO PICTURE 9(2) COMP.	*AA040
05	S-COPOS-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-COPOS-DYN PICTURE X.	*AA040
05	S-COPOS-BACK PICTURE X.	*AA040
05	S-COPOS-FORE PICTURE X.	*AA040
05	S-COPOS-INT PICTURE X.	*AA040
05	S-COPOS-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-COPOS-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-REFCLI-YCO PICTURE 9(2) COMP.	*AA040
05	S-REFCLI-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-REFCLI-DYN PICTURE X.	*AA040
05	S-REFCLI-BACK PICTURE X.	*AA040
05	S-REFCLI-FORE PICTURE X.	*AA040
05	S-REFCLI-INT PICTURE X.	*AA040
05	S-REFCLI-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-REFCLI-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-DATE-YCO PICTURE 9(2) COMP.	*AA040
05	S-DATE-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-DATE-DYN PICTURE X.	*AA040
05	S-DATE-BACK PICTURE X.	*AA040
05	S-DATE-FORE PICTURE X.	*AA040
05	S-DATE-INT PICTURE X.	*AA040
05	S-DATE-HIGH PICTURE X.	*AA040
05	FILLER PICTURE X.	*AA040
05	S-DATE-EMPH PICTURE X.	*AA040
05	FILLER PICTURE XX.	*AA040
05	S-CORRES-YCO PICTURE 9(2) COMP.	*AA040
05	S-CORRES-XCO PICTURE 9(2) COMP.	*AA040
05	FILLER PICTURE X(5).	*AA040
05	S-CORRES-DYN PICTURE X.	*AA040
05	S-CORRES-BACK PICTURE X.	*AA040
05	S-CORRES-FORE PICTURE X.	*AA040
05	S-CORRES-INT PICTURE X.	*AA040

GENERATED PROGRAM: DATA DIVISION
FORM DESCRIPTION

PAGE

55

3
4

05	S-CORRES-HIGH	PICTURE X.	*AA040
05	FILLER	PICTURE X.	*AA040
05	S-CORRES-EMPH	PICTURE X.	*AA040
05	FILLER	PICTURE XX.	*AA040
05	S-REMIS-YCO	PICTURE 9(2) COMP.	*AA040
05	S-REMIS-XCO	PICTURE 9(2) COMP.	*AA040
05	FILLER	PICTURE X(5).	*AA040
05	S-REMIS-DYN	PICTURE X.	*AA040
05	S-REMIS-BACK	PICTURE X.	*AA040
05	S-REMIS-FORE	PICTURE X.	*AA040
05	S-REMIS-INT	PICTURE X.	*AA040
05	S-REMIS-HIGH	PICTURE X.	*AA040
05	FILLER	PICTURE X.	*AA040
05	S-REMIS-EMPH	PICTURE X.	*AA040
05	J-LINE	OCCURS 9.	*AA040
10	FILLER	PICTURE X(96).	*AA040
05	FILLER	PICTURE XX.	*AA040
05	S-EDIT-YCO	PICTURE 9(2) COMP.	*AA040
05	S-EDIT-XCO	PICTURE 9(2) COMP.	*AA040
05	FILLER	PICTURE X(5).	*AA040
05	S-EDIT-DYN	PICTURE X.	*AA040
05	S-EDIT-BACK	PICTURE X.	*AA040
05	S-EDIT-FORE	PICTURE X.	*AA040
05	S-EDIT-INT	PICTURE X.	*AA040
05	S-EDIT-HIGH	PICTURE X.	*AA040
05	FILLER	PICTURE X.	*AA040
05	S-EDIT-EMPH	PICTURE X.	*AA040
05	FILLER	PICTURE XX.	*AA040
05	S-CHOIX-YCO	PICTURE 9(2) COMP.	*AA040
05	S-CHOIX-XCO	PICTURE 9(2) COMP.	*AA040
05	FILLER	PICTURE X(5).	*AA040
05	S-CHOIX-DYN	PICTURE X.	*AA040
05	S-CHOIX-BACK	PICTURE X.	*AA040
05	S-CHOIX-FORE	PICTURE X.	*AA040
05	S-CHOIX-INT	PICTURE X.	*AA040
05	S-CHOIX-HIGH	PICTURE X.	*AA040
05	FILLER	PICTURE X.	*AA040
05	S-CHOIX-EMPH	PICTURE X.	*AA040
05	FILLER	PICTURE XX.	*AA040
05	S-MESSA-YCO	PICTURE 9(2) COMP.	*AA040
05	S-MESSA-XCO	PICTURE 9(2) COMP.	*AA040
05	FILLER	PICTURE X(5).	*AA040
05	S-MESSA-DYN	PICTURE X.	*AA040
05	S-MESSA-BACK	PICTURE X.	*AA040
05	S-MESSA-FORE	PICTURE X.	*AA040
05	S-MESSA-INT	PICTURE X.	*AA040
05	S-MESSA-HIGH	PICTURE X.	*AA040
05	FILLER	PICTURE X.	*AA040
05	S-MESSA-EMPH	PICTURE X.	*AA040
05	FILLER	OCCURS 1.	*AA040
15	FILLER	PICTURE XX.	*AA040
15	S-ERMSG-YCO	PICTURE 9(2) COMP.	*AA040
15	S-ERMSG-XCO	PICTURE 9(2) COMP.	*AA040
15	FILLER	PICTURE X(5).	*AA040
15	S-ERMSG-DYN	PICTURE X.	*AA040
15	S-ERMSG-BACK	PICTURE X.	*AA040
15	S-ERMSG-FORE	PICTURE X.	*AA040
15	S-ERMSG-INT	PICTURE X.	*AA040
15	S-ERMSG-HIGH	PICTURE X.	*AA040
15	FILLER	PICTURE X.	*AA040
15	S-ERMSG-EMPH	PICTURE X.	*AA040
COPY	SCREEN-DO0030-130-DATA.		*AA050
01	INPUT-SCREEN-FIELDS REDEFINES		*AA050
	SCREEN-DO0030-130-DATA.		*AA050
02	I-0030.		*AA050
05	I-0030-PROGE	PICTURE X(8).	*AA050
05	I-0030-SESSI	PICTURE X(5).	*AA050
05	FILLER	PICTURE X(03).	*AA050
05	I-0030-DATEM	PICTURE X(10).	*AA050
05	FILLER	PICTURE X(02).	*AA050
05	I-0030-HEURE	PICTURE X(8).	*AA050
05	I-0030-NUCOM	PICTURE 9(5).	*AA050
05	FILLER	PICTURE X(03).	*AA050
05	I-0030-MATE	PICTURE X(8).	*AA050
05	I-0030-RELEA	PICTURE X(3).	*AA050
05	FILLER	PICTURE X(01).	*AA050
05	I-0030-RAISOC	PICTURE X(50).	*AA050

GENERATED PROGRAM: DATA DIVISION
FORM DESCRIPTION

3

4

	05	FILLER PICTURE X(02).	*AA050
	05	I-0030-RUE PICTURE X(40).	*AA050
	05	I-0030-VILLE PICTURE X(20).	*AA050
	05	I-0030-COPOS PICTURE X(5).	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	I-0030-REFCLI PICTURE X(30).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	I-0030-DATE PICTURE X(6).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	I-0030-CORRES PICTURE X(25).	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	E-0030-REMIS.	*AA050
	10	I-0030-REMIS PICTURE S9(4)V99.	*AA050
	10	FILLER PICTURE X(2).	*AA050
	05	J-0030-LINE OCCURS 9.	*AA050
	10	FILLER PICTURE X(56).	*AA050
	05	I-0030-EDIT PICTURE X.	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	I-0030-CHOIX PICTURE X.	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	I-0030-MESSA PICTURE X(75).	*AA050
	05	FILLER PICTURE X(01).	*AA050
	05	I-0030-ERMS.	*AA050
	10	FILLER OCCURS 1.	*AA050
	15	I-0030-ERMSG PICTURE X(72).	*AA050
01		OUTPUT-SCREEN-FIELDS.	*AA050
	02	O-0030.	*AA050
	05	O-0030-PROGE PICTURE X(8).	*AA050
	05	O-0030-SESSI PICTURE X(5).	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	O-0030-DATEM PICTURE X(10).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	O-0030-HEURE PICTURE X(8).	*AA050
	05	O-0030-NUCOM PICTURE 9(5).	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	O-0030-MATE PICTURE X(8).	*AA050
	05	O-0030-RELEA PICTURE X(3).	*AA050
	05	FILLER PICTURE X(01).	*AA050
	05	O-0030-RAISOC PICTURE X(50).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	O-0030-RUE PICTURE X(40).	*AA050
	05	O-0030-VILLE PICTURE X(20).	*AA050
	05	O-0030-COPOS PICTURE X(5).	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	O-0030-REFCLI PICTURE X(30).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	O-0030-DATE PICTURE X(6).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	O-0030-CORRES PICTURE X(25).	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	F-0030-REMIS.	*AA050
	10	O-0030-REMIS PICTURE -(04)9,9(02).	*AA050
	05	P-0030-LINE OCCURS 9.	*AA050
	10	FILLER PICTURE X(56).	*AA050
	05	O-0030-EDIT PICTURE X.	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	O-0030-CHOIX PICTURE X.	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	O-0030-MESSA PICTURE X(75).	*AA050
	05	FILLER PICTURE X(01).	*AA050
	05	O-0030-ERMS.	*AA050
	10	FILLER OCCURS 1.	*AA050
	15	O-0030-ERMSG PICTURE X(72).	*AA050
01		REPEAT-LINE.	*AA050
	02	I-0030-LINE.	*AA050
	05	I-0030-CODMVT PICTURE X.	*AA050
	05	FILLER PICTURE X(03).	*AA050
	05	I-0030-FOURNI PICTURE X(3).	*AA050
	05	FILLER PICTURE X(01).	*AA050
	05	E-0030-QTMAC.	*AA050
	10	I-0030-QTMAC PICTURE 99.	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	I-0030-QTMAL PICTURE 99.	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	I-0030-QTMAR PICTURE 99.	*AA050
	05	FILLER PICTURE X(02).	*AA050
	05	I-0030-INFOR PICTURE X(35).	*AA050

GENERATED PROGRAM: DATA DIVISION
FORM DESCRIPTION

PAGE

57

3
4

05	FILLER PICTURE X(01).	*AA050
02	O-0030-LINE.	*AA050
05	O-0030-CODMVT PICTURE X.	*AA050
05	FILLER PICTURE X(03).	*AA050
05	O-0030-FOURNI PICTURE X(3).	*AA050
05	FILLER PICTURE X(01).	*AA050
05	F-0030-QTMAC.	*AA050
10	O-0030-QTMAC PICTURE Z(01)9.	*AA050
05	FILLER PICTURE X(02).	*AA050
05	O-0030-QTMAL PICTURE 99.	*AA050
05	FILLER PICTURE X(02).	*AA050
05	O-0030-QTMAR PICTURE 99.	*AA050
05	FILLER PICTURE X(02).	*AA050
05	O-0030-INFOR PICTURE X(35).	*AA050
05	FILLER PICTURE X(01).	*AA050
02	I-LINE.	*AA050
05	FILLER PICTURE XX.	*AA050
05	S-CODMVT-YCO PICTURE 9(2) COMP.	*AA050
05	S-CODMVT-XCO PICTURE 9(2) COMP.	*AA050
05	FILLER PICTURE X(5).	*AA050
05	S-CODMVT-DYN PICTURE X.	*AA050
05	S-CODMVT-BACK PICTURE X.	*AA050
05	S-CODMVT-FORE PICTURE X.	*AA050
05	S-CODMVT-INT PICTURE X.	*AA050
05	S-CODMVT-HIGH PICTURE X.	*AA050
05	FILLER PICTURE X.	*AA050
05	S-CODMVT-EMPH PICTURE X.	*AA050
05	FILLER PICTURE XX.	*AA050
05	S-FOURNI-YCO PICTURE 9(2) COMP.	*AA050
05	S-FOURNI-XCO PICTURE 9(2) COMP.	*AA050
05	FILLER PICTURE X(5).	*AA050
05	S-FOURNI-DYN PICTURE X.	*AA050
05	S-FOURNI-BACK PICTURE X.	*AA050
05	S-FOURNI-FORE PICTURE X.	*AA050
05	S-FOURNI-INT PICTURE X.	*AA050
05	S-FOURNI-HIGH PICTURE X.	*AA050
05	FILLER PICTURE X.	*AA050
05	S-FOURNI-EMPH PICTURE X.	*AA050
05	FILLER PICTURE XX.	*AA050
05	S-QTMAC-YCO PICTURE 9(2) COMP.	*AA050
05	S-QTMAC-XCO PICTURE 9(2) COMP.	*AA050
05	FILLER PICTURE X(5).	*AA050
05	S-QTMAC-DYN PICTURE X.	*AA050
05	S-QTMAC-BACK PICTURE X.	*AA050
05	S-QTMAC-FORE PICTURE X.	*AA050
05	S-QTMAC-INT PICTURE X.	*AA050
05	S-QTMAC-HIGH PICTURE X.	*AA050
05	FILLER PICTURE X.	*AA050
05	S-QTMAC-EMPH PICTURE X.	*AA050
05	FILLER PICTURE XX.	*AA050
05	S-QTMAL-YCO PICTURE 9(2) COMP.	*AA050
05	S-QTMAL-XCO PICTURE 9(2) COMP.	*AA050
05	FILLER PICTURE X(5).	*AA050
05	S-QTMAL-DYN PICTURE X.	*AA050
05	S-QTMAL-BACK PICTURE X.	*AA050
05	S-QTMAL-FORE PICTURE X.	*AA050
05	S-QTMAL-INT PICTURE X.	*AA050
05	S-QTMAL-HIGH PICTURE X.	*AA050
05	FILLER PICTURE X.	*AA050
05	S-QTMAL-EMPH PICTURE X.	*AA050
05	FILLER PICTURE XX.	*AA050
05	S-QTMAR-YCO PICTURE 9(2) COMP.	*AA050
05	S-QTMAR-XCO PICTURE 9(2) COMP.	*AA050
05	FILLER PICTURE X(5).	*AA050
05	S-QTMAR-DYN PICTURE X.	*AA050
05	S-QTMAR-BACK PICTURE X.	*AA050
05	S-QTMAR-FORE PICTURE X.	*AA050
05	S-QTMAR-INT PICTURE X.	*AA050
05	S-QTMAR-HIGH PICTURE X.	*AA050
05	FILLER PICTURE X.	*AA050
05	S-QTMAR-EMPH PICTURE X.	*AA050
05	FILLER PICTURE XX.	*AA050
05	S-INFOR-YCO PICTURE 9(2) COMP.	*AA050
05	S-INFOR-XCO PICTURE 9(2) COMP.	*AA050
05	FILLER PICTURE X(5).	*AA050
05	S-INFOR-DYN PICTURE X.	*AA050
05	S-INFOR-BACK PICTURE X.	*AA050

GENERATED PROGRAM: DATA DIVISION
FORM DESCRIPTION

PAGE

58

3
4

05	S-INFOR-FORE	PICTURE X.	*AA050
05	S-INFOR-INT	PICTURE X.	*AA050
05	S-INFOR-HIGH	PICTURE X.	*AA050
05	FILLER	PICTURE X.	*AA050
05	S-INFOR-EMPH	PICTURE X.	*AA050

3.5. DESCRIPTION OF VALIDATION AREAS

DESCRIPTION OF VALIDATION AREAS

The validation processing part of the program is always generated in the WORKING-STORAGE SECTION. It includes all the work areas necessary for the generated validation processing.

NUMERIC FIELDS OF THE SCREEN

The 'NUMERIC-FIELDS' level is generated when the screen includes at least one variable Data Element.

Field '9-scrn-delco' (scrn = last 4 characters of the screen code) is generated for each numeric Data Element. It contains the breakdown of the Data Element's VALUE in 'seedd' where:

s = ' ' non-signed Data Element.

'+' signed Data Element.

ee = number of digits in the integer part of the Data Element.

dd = number of digits in the decimal part of the Data Element.

DESCRIPTION OF ERROR MESSAGE FILE

The EM00 level corresponding to the description of the error message file, is systematically generated. This description can be replaced by the user, as long as the new description uses the same field names as in the PROCEDURE DIVISION: EM00-LIBRA, EM00-PROGR, etc.

VALIDATION VARIABLES

The 'VALIDATION-TABLE-FIELDS' level is generated if there is at least one variable data element (NATURE = 'V') used on the screen.

DE-ERR : memorizes the presence and/or status of each Data Element of the screen.

A position in this table (coded ER-scrn-delco) is associated with each Data Element of the screen. This is generated at the '05' level ('scrn' = last four characters of the screen code).

Depending on the stages of validation, this position can be set to the following values:

- .0 Data Element absent.
- .1 Data Element present.
- .2 Invalid absence of data element.
- .4 Erroneous class.
- .5 Invalid content.

This table of error positions is structured according to the categories defined on the screen and the group data element in the following manner:

A group level for the Data Elements from the beginning of the screen is systematically generated in the form of:

ER-nn-BEGIN.

For a repetitive Data Element defining a repetitive area of the screen (data element on the screen with NATURE = 'R'), the generation of the error positions is as follows:

- .03 ES-scrn-LINE OCCURS 9.
- .05 FILLER PICTURE X(0004).

In this example:

LINE is the code of the Data Element with NATURE = 'R' (see above),
9 is the number of repetitions,
0004 is the number of Data Elements in the repetitive category.

After the table of errors, there is an area which will contain the error positions of the Data Elements from the repetitive category. This area is used to position the errors for each of these data elements, with each occurrence.

.02 ER-nn-LINE.

.05 ER-nn-CODMVT PICTURE X.

.05 ER-nn-FOURNI PICTURE X.

etc.

For a repetitive Data Element whose NATURE is other than 'R', the generation in the table of error positions does not provide the description of the sample item, but does provide the following:

.05 FILLER OCCURS 2.

.10 ER-nn-LREF1 PICTURE X.

A group level for the Data Elements from the screen-bottom category is generated using a Data Element whose NATURE = 'Z', which contains the error positions of Data Elements belonging to that category:

.03 ER-nn-END.

.05 ER-nn-EDIT PICTURE X.

etc.

TT-DAT

The 'TT-DAT' level is generated if a variable Data Element (NATURE = 'V') contains a 'date' format. It is used in sub-function F8120-M for date formatting purposes.

LEAP-YEAR

The 'LEAP-YEAR' level is generated if a variable Data Element (NATURE = 'V') contains a 'date' format (always generated with CICS). It is used in F81-ER to determine whether or not the year is a leap year.

USERS-ERROR

The 'USERS-ERROR' level is always generated, and it contains:

XEMKY: Table position used to build the key, including:

'XPROGR' Name of the program or dialogue,
'XERCD' Error number and type of error,

T-XEMKY: Table of errors, corresponding to the number of error messages on the screen (default value = 1).

INDEXES

The 'INDEXES' level is always generated. It includes:

K01, K02, K03, K04

Indexes for automatic numeric class.

K50R, K50L, K50M

Indexes associated with the table of user errors (the value assigned to K50M directly relates to the number of vertical repetitions of Data Element 'ERMSG' in the screen description).

5-dd00-LTH

Length of longest Segment of the Data Structure (common part + specific part; 'dd' = code of the Data Structure).

5-ddss-LTH

Length of the Segment without the common part (not generated for the common part, 'dd00'; 'ddss' = code of the Segment).

5-ddss-LTHV

Length of the Data Structure Segment including the common part (not generated for the common part, 'dd00'; 'ddss' = code of the Segment).

LTH Calculation area used during access to files with a Table or VSAM ORGANIZATION.

KEYLTH

Calculation area of the key used during access to files with a VSAM ORGANIZATION.

5-scrn-LENGTH

Area containing the length of the communication area (scrn = last four char. of screen code).

GENERATED PROGRAM: DATA DIVISION
DESCRIPTION OF VALIDATION AREAS

PAGE

64

3
5

NUMERIC-VALIDATION-FIELDS

The 'NUMERIC-VALIDATION-FIELDS' level is generated if there is at least one variable numeric field on the screen. It contains the work areas necessary for analyzing and formatting numeric Data Elements on the screen (refer to subchapter "F81 : CALLED VALIDATION FUNCTIONS").

GENERATED PROGRAM: DATA DIVISION
DESCRIPTION OF VALIDATION AREAS

3
5

01		NUMERIC-FIELDS.		*AA050
	05	9-0030-REMIS	PICTURE X(5) VALUE '+0402'.	*AA050
	05	9-0030-QTMAC	PICTURE X(5) VALUE ' 0200'.	*AA050
01		EM00.		*AA100
	05	EM00-EMKEY.		*AA100
	10	EM00-LIBRA	PICTURE X(3).	*AA100
	10	EM00-ENTYP	PICTURE X.	*AA100
	10	EM00-XEMKY.		*AA100
	15	EM00-PROGR	PICTURE X(6).	*AA100
	15	EM00-ERCOD.		*AA100
	20	EM00-ERCOD9	PICTURE 9(3).	*AA100
	15	EM00-ERTYP	PICTURE X.	*AA100
	10	EM00-LINUM	PICTURE 9(3).	*AA100
	05	EM00-ERLVL	PICTURE X.	*AA100
	05	EM00-ERMSG	PICTURE X(66).	*AA100
	05	FILLER	PICTURE X(6).	*AA100
01		VALIDATION-TABLE-FIELDS.		*AA150
	02	DE-ERR.		*AA150
	05	DE-ER	PICTURE X	*AA150
			OCCURS 046.	*AA150
	02	DE-E	REDEFINES DE-ERR.	*AA150
	03	ER-0030-BEGIN.		*AA150
	05	ER-0030-CHOIX	PICTURE X.	*AA150
	05	ER-0030-MATE	PICTURE X.	*AA150
	05	ER-0030-RELEA	PICTURE X.	*AA150
	05	ER-0030-RUE	PICTURE X.	*AA150
	05	ER-0030-COPOS	PICTURE X.	*AA150
	05	ER-0030-REFCLI	PICTURE X.	*AA150
	05	ER-0030-DATE	PICTURE X.	*AA150
	05	ER-0030-CORRES	PICTURE X.	*AA150
	05	ER-0030-REMIS	PICTURE X.	*AA150
	03	PS-30-LINE	OCCURS 9.	*AA150
	05	FILLER	PICTURE X(0004).	*AA150
	03	ER-0030-END.		*AA150
	05	ER-0030-EDIT	PICTURE X.	*AA150
	02	ER-0030-LINE.		*AA150
	05	ER-0030-CODMVT	PICTURE X.	*AA150
	05	ER-0030-FOURNI	PICTURE X.	*AA150
	05	ER-0030-QTMAC	PICTURE X.	*AA150
	05	ER-0030-INFOR	PICTURE X.	*AA150
01		TT-DAT.		*AA200
	05	T-DAT	PICTURE X OCCURS 5.	*AA200
01		LEAP-YEAR.		*AA200
	05	LEAP-FLAG	PICTURE X.	*AA200
	05	LEAP-REM	PICTURE 99.	*AA200
01		USERS-ERROR.		*AA200
	05	XEMKY.		*AA200
	10	XPROGR	PICTURE X(6).	*AA200
	10	XERCD	PICTURE X(4).	*AA200
	05	T-XEMKY	OCCURS 01.	*AA200
	10	T-XPROGR	PICTURE X(6).	*AA200
	10	T-XERCD	PICTURE X(4).	*AA200
01		PACBASE-INDEXES	COMPUTATIONAL.	*AA200
	05	TALLI	PICTURE S9(4) VALUE ZERO.	*AA200
	05	K01	PICTURE S9(4).	*AA200
	05	K02	PICTURE S9(4).	*AA200
	05	K03	PICTURE S9(4).	*AA200
	05	K04	PICTURE S9(4).	*AA200
	05	K50R	PICTURE S9(4) VALUE ZERO.	*AA200
	05	K50L	PICTURE S9(4) VALUE ZERO.	*AA200
	05	K50M	PICTURE S9(4)	*AA200
			VALUE +01.	*AA200
	05	IWP20L	PICTURE S9(4) VALUE ZERO.	*AA200
	05	IWP20R	PICTURE S9(4) VALUE ZERO.	*AA200
	05	IWP20M	PICTURE S9(4) VALUE +0009.	*AA200
	05	5-CD00-LTH	PICTURE S9(4) VALUE +0166.	*AA200
	05	5-CD05-LTH	PICTURE S9(4) VALUE +0157.	*AA200
	05	5-CD10-LTH	PICTURE S9(4) VALUE +0139.	*AA200
	05	5-CD20-LTH	PICTURE S9(4) VALUE +0001.	*AA200
	05	5-FO00-LTH	PICTURE S9(4) VALUE +0057.	*AA200
	05	5-FO10-LTH	PICTURE S9(4) VALUE +0057.	*AA200
	05	5-ME00-LTH	PICTURE S9(4) VALUE +0082.	*AA200
	05	5-CA00-LTH	PICTURE S9(4) VALUE +0147.	*AA200
	05	5-EM00-LTH	PICTURE S9(4) VALUE +0090.	*AA200
	05	5-CD05-LTHV	PICTURE S9(4) VALUE +0166.	*AA200
	05	5-CD10-LTHV	PICTURE S9(4) VALUE +0148.	*AA200
	05	5-CD20-LTHV	PICTURE S9(4) VALUE +0010.	*AA200

GENERATED PROGRAM: DATA DIVISION
DESCRIPTION OF VALIDATION AREAS

PAGE

66

3
5

05	5-F010-LTHV	PICTURE S9(4) VALUE +0057.	*AA200
05	LTH	PICTURE S9(4) VALUE ZERO.	*AA200
05	5-0030-LENGTH	PICTURE S9(4)	*AA200
		VALUE +0880.	*AA200
01	NUMERIC-VALIDATION-FIELDS.		*AA200
05	ZONUM1.		*AA200
	10 C1	PICTURE X OCCURS 27.	*AA200
05	ZONUM2.		*AA200
	10 C2	OCCURS 18.	*AA200
	15 C29	PICTURE S9.	*AA200
05	ZONUM9	REDEFINES ZONUM2 PICTURE 9(18).	*AA200
05	NUMPIC.		*AA200
	10 SIGNE	PICTURE X.	*AA200
	10 NBCHA	PICTURE 99.	*AA200
	10 NBCHP	PICTURE 99.	*AA200
05	C9	PICTURE S9.	*AA200
05	C91	PICTURE X.	*AA200
05	TPOINT	PICTURE X.	*AA200
05	ZONUM3.		*AA200
	10 C3	PICTURE X OCCURS 18.	*AA200
05	ZONUM4	REDEFINES ZONUM3 PICTURE 9(18).	*AA200
05	ZONUM5	PICTURE S99 VALUE -10.	*AA200
05	ZONUM6	REDEFINES ZONUM5.	*AA200
	10 FILLER	PICTURE X.	*AA200
	10 C4	PICTURE X.	*AA200

3.6. TABLE-OF-ATTRIBUTES AND SEGMENT VARIABLES

TABLE-OF-ATTRIBUTES AND SEGMENT VARIABLES

The 'TABLE-OF-ATTRIBUTES' level is generated if the screen includes at least one variable Data Element (NATURE = 'V').

The DE-ATT table is the image of DE-ERR repeated four times. It is used to store the attributes of the Data Elements on the screen.

It is used to set the error attributes (which have been defined at the screen level) for a Data Element in error (for the management of this table refer to Subchapter "ERROR PROCESSING (F70)", Chapter "GENERATED PROGRAM: PROCEDURE DIVISION").

The coding for each Data Element is formatted as follows:

```
.A-scrn-MATE (A) for non-repetitive Data Elements  
.B-scrn-LINE (B) for the Data Elements defining a  
repetitive category (Nature 'R').
```

NOTE: 'scrn' = the last four characters of the screen code.

The table positions correspond to the attributes:

```
A = 1 Intensity attribute.  
A = 2 Presentation attribute.  
A = 3 Color attribute.  
A = 4 Cursor positioned on the Data Element.
```

After the Table-of-Attributes, there is an area detailing the attributes of the Data Elements of the repetitive category. This area is used to position the attributes of each occurrence of these Data Elements.

```
.02 A-0030-LINE OCCURS 4.  
.05 A-0030-CODMVT PICTURE X.  
.05 A-0030-FOURNI PICTURE X.  
etc.
```

The 'STOP-FIELDS' level is generated if a display control break has been defined for at least one Data Element of the repetitive category (display control break 'C' for a Data Element of a Segment used on the screen):

```
.02 C-0030  
.05 C-0030-COCARA PICTURE X.  
.05 C-0030-NUCOM PICTURE 9(5).
```

These areas are used to store the value of a Data Element which must remain constant in the display.

The 'FIRST-ON-SEGMENT' level is generated when at least one Segment that is not preceded by an access to another Segment, is used on display in the repetitive category.

In this case, a variable is generated for each Segment, indicating the first access to the Segment (key to be loaded in order to read the Segment on display).

Example:

```
05 CD10-FST PICTURE X.  
  
. '1' First on the Segment,  
. '0' Next read of the Segment.
```

GENERATED PROGRAM: DATA DIVISION

3

TABLE-OF-ATTRIBUTES AND SEGMENT VARIABLES

6

01	TABLE-OF-ATTRIBUTES.	*AA250
02	DE-ATT.	*AA250
03	DE-ATT1 OCCURS 4.	*AA250
05	DE-AT PICTURE X	*AA250
	OCCURS 046.	*AA250
02	DE-A REDEFINES DE-ATT.	*AA250
03	DE-ATT2 OCCURS 4.	*AA250
04	A-0030-BEGIN.	*AA250
05	A-0030-CHOIX PICTURE X.	*AA250
05	A-0030-MATE PICTURE X.	*AA250
05	A-0030-RELEA PICTURE X.	*AA250
05	A-0030-RUE PICTURE X.	*AA250
05	A-0030-COPOS PICTURE X.	*AA250
05	A-0030-REFCLI PICTURE X.	*AA250
05	A-0030-DATE PICTURE X.	*AA250
05	A-0030-CORRES PICTURE X.	*AA250
05	A-0030-REMIS PICTURE X.	*AA250
04	B-0030-LINE OCCURS 9.	*AA250
05	FILLER PICTURE X(0004).	*AA250
04	A-0030-END.	*AA250
05	A-0030-EDIT PICTURE X.	*AA250
02	A-0030-LINE OCCURS 4.	*AA250
05	A-0030-CODMVT PICTURE X.	*AA250
05	A-0030-FOURNI PICTURE X.	*AA250
05	A-0030-QTMAC PICTURE X.	*AA250
05	A-0030-INFOR PICTURE X.	*AA250
01	STOP-FIELDS.	*AA300
02	C-0030.	*AA300
05	C-0030-COCARA PICTURE X.	*AA300
05	C-0030-NUCOM PICTURE 9(5).	*AA300
01	FIRST-ON-SEGMENT.	*AA301
05	CD10-FST PICTURE X.	*AA301
01	5-0030-TRX PICTURE X(6).	*AA400
01	END-MESSAGE PICTURE X(80) VALUE SPACE.	*AA400
	COPY INFO-BUFFER.	*AA400
	COPY SENDERROR.	*AA400
	COPY STATUS-WORD.	*AA400
01	WW10-QTMAR	*BB200
	PICTURE 99	*BB200
	VALUE ZERO.	*BB201
01	WP00.	*WP000
02	WP10.	*WP010
05	FILLER PIC X(25) VALUE	*WP020
	'23400BRISBANE	*WP030
05	FILLER PIC X(25) VALUE	*WP040
	'56400VICTORIA	*WP050
05	FILLER PIC X(25) VALUE	*WP060
	'76500ALICE SPRINGS	*WP070
05	FILLER PIC X(25) VALUE	*WP080
	'55300MELBOURNE	*WP090
05	FILLER PIC X(25) VALUE	*WP100
	'11000CANBERRA	*WP110
05	FILLER PIC X(25) VALUE	*WP120
	'34500PERTH	*WP130
05	FILLER PIC X(25) VALUE	*WP140
	'85270DARWIN	*WP150
05	FILLER PIC X(25) VALUE	*WP160
	'94000HOBART	*WP170
05	FILLER PIC X(25) VALUE	*WP180
	'89300SYDNEY	*WP190
02	WP20 REDEFINES WP10 OCCURS 9.	*WP300
05	WP20-COPOS	*WP320
	PICTURE X(5).	*WP320
05	WP20-VILLE	*WP340
	PICTURE X(20).	*WP340
02	WP30.	*WP400
05	WP30-COPOS	*WP410
	PICTURE X(5).	*WP410
02	WP40.	*WP500
05	WP40-VILLE	*WP510
	PICTURE X(20).	*WP510
05	WP40-VILLEL	*WP520
	PICTURE X(20).	*WP520
01	7-WW00.	*WW100
05	FILLER PICTURE X(6) VALUE 'FNPAC '	*WW120
05	7-WW00-FONCT PICTURE X(6).	*WW130
05	FILLER PICTURE X(7) VALUE ' ORDRE '	*WW140

GENERATED PROGRAM: DATA DIVISION

3

TABLE-OF-ATTRIBUTES AND SEGMENT VARIABLES

6

05	7-WW00-ORDRE	PICTURE X(8).	*WW150
05	FILLER	PICTURE X(7) VALUE ' RBCOD '.	*WW160
05	7-WW00-RBCODE	PICTURE X(2).	*WW170
05	FILLER	PICTURE X(4) VALUE ' FN '.	*WW180
05	7-WW00-FUNCT	PICTURE X(2).	*WW190
05	FILLER	PICTURE X(7) VALUE ' ERCOD '.	*WW200
05	7-WW00-ERCOD	PICTURE X(2).	*WW210
05	FILLER	PICTURE X(7) VALUE ' ERNUM '.	*WW220
05	7-WW00-NUM	PICTURE X(4).	*WW230
01	7-HELP-ERROR	PICTURE X(72) VALUE	*WW500
	'***** HELP FUNCTION UNAVAILABLE *****'.		*WW510
01	DBK-REC USAGE	IS DATABASE-KEY.	*WX200

3.7. COMMUNICATION AREA DESCRIPTION

COMMON AREA

The description of the COMMON AREA is found in the WORKING-STORAGE SECTION.

It is generated according to the values entered on the Dialogue Complement (-O) screen, and the segment access keys used in display.

It is the common area for all of the screens in the dialogue, and contains the following fields:

K-S0030-PROGR

systematically generated and is used to store the complete screen code.

CA00

Data Structure describing the user Common Area (if the data structure contains several segments, they are described in 'REDEFINE' clauses.

If a documentation Help character has been entered on the Screen Definition screen, the following fields are generated:

.K-S0030-DOC is the Help function flag:
 '0' No Help function request
 '1' Return to Help function
 '2' Request for screen-level documentation
 '3' Request for field-level documentation

.K-S0030-PROGE Used to store the external name of the calling program.

.K-S0030-LIBRA Used to store the library code.

.K-S0030-PROHE Four fields specific to the On-Line
.K-S0030-ERCOD Help Function.
.K-S0030-ERTYP
.K-S0030-LINUM

K-0030

Complementary field for memorization of the Dialogue (Refer to Subchapter "DIALOGUE COMPLEMENT (-O)", Chapter "DESCRIPTION OF A TRANSACTION" in the General OLSD Reference Manual).

The following fields are used to store the access keys of segments used in display (without a preceding segment):

K-A0030-DEBUT

Automatically generated for the screen-top category.

K-ACD05-KEYCD

Key for the screen-top category.

K-R0030-LINE OCCURS 2

Generated according to the data element defining the repetitive category (the 1st occurrence stores the key for the beginning of display; the 2nd stores the next screen's display key).

K-RCD10-KEYCD

Key for the repetitive category.

K-Z0030-END

Key of the screen-bottom category. Generated according to the data element defining the screen-bottom category.

K-ZME00-CLEME

Key for the screen-bottom category.

A 'FILLER' aligns the K-0030 fields on 100 characters, except if the user has entered a greater length on the Dialogue Complement (-O) screen.

GENERATED PROGRAM: DATA DIVISION
COMMUNICATION AREA DESCRIPTION

PAGE

73

3
7

01	COMMON-AREA.	*00000
02	K-S0030-PROGR PICTURE X(6).	*00000
02	CA00.	*00001
10	CA00-CLECD.	*00001
15	CA00-NUCOM PICTURE 9(5).	*00001
10	CA00-CLECL1.	*00001
15	CA00-NUCLIE PICTURE 9(8).	*00001
10	CA00-ME00.	*00001
15	CA00-CLEME.	*00001
20	CA00-COPERS PICTURE X(5).	*00001
20	CA00-NUMORD PICTURE XX.	*00001
15	CA00-MESSA PICTURE X(75).	*00001
10	CA00-PREM PICTURE X.	*00001
10	CA00-LANGU PICTURE X.	*00001
10	CA00-RAISOC PICTURE X(50).	*00001
02	K-S0030-DOC PICTURE X.	*00002
02	K-S0030-PROGE PICTURE X(8).	*00002
02	K-S0030-LIBRA PICTURE XXX.	*00002
02	K-S0030-PROHE PICTURE X(8).	*00002
02	K-S0030-ERCOD.	*00002
05	K-S0030-ERCOD9 PICTURE 999.	*00002
02	K-S0030-ERTYP PICTURE X.	*00002
02	K-S0030-LINUM PICTURE 999.	*00002
02	K-0030.	*00002
03	K-A0030-DEBUT.	*00002
05	K-ACD05-KEYCD PICTURE X(00009).	*00002
03	K-R0030-LINE OCCURS 2.	*00002
05	K-RCD10-KEYCD PICTURE X(00009).	*00002
03	K-Z0030-END.	*00002
05	K-ZME00-CLEME PICTURE X(7).	*00002
02	FILLER PICTURE X(0666).	*00002

4. GENERATED PROGRAM: PROCEDURE DIVISION

4.1. STRUCTURE OF THE PROCEDURE DIVISION

STANDARD STRUCTURE OF THE PROCEDURE DIVISION

```
F0110  Initializations
-----

F05      RECEPTION      (ICF = '1')

F0510    Reception of the screen
F0510-A  PFkey positioning
F0512    Documentation call procedure
F0520    Validation of Operation Code (OPER)
F1010    Category processing      <-----
F15      Validation of the Transaction Code (CATM)!
F20      Data element validation      !
F25      Segment access for validation      !
F30      Data element transfer        !
F35      Segment access for update      !
F3999-ITER-FN. GO TO F10.  -----
F3999-ITER-FT.  EXIT.

F40      END-OF-RECEPTION PROCESSING

F4010    Set-up keys for new display
F4020    Set-up keys for screen paging
F4030    End of transaction
F4040    Transfer to another screen

END-OF-RECEPTION. (F45-FN)
-----

F50      DISPLAY PREPARATION      (OCF = '1')

F5010    Initialization
F5510    Category processing      <-----
F60      Segment access for display      !
F65      Data element transfer        !
F6999-ITER-FN. GO TO F55.  -----
F6999-ITER-FT.  EXIT.

F7010    Error processing
F7020    Positioning of attributes

END-OF-DISPLAY. (F78-FN)
```

F8Z DISPLAY AND END OF PROGRAM

F8Z05 Memorization of the screen
F8Z10 Display
F8Z20 End of program

----- Performed Functions -----

F80 PHYSICAL SEGMENT ACCESS ROUTINES
F8098 Error Message File Access
F81ER Abnormal End Procedure
F81UT Memorization of User's Errors
F8110 Numeric Validation
F8115 Initialization of the Variable Fields
F8120 Date Format Validation
F8125 Transfer to Display
F8130 Help Sub-function
F8135 Transfer to Reception
F8140 Cursor Position Calculation

4.2. INITIALIZATIONS (F01)

F01: INITIALIZATIONS

The INITIALIZATION FUNCTION (F01) is always generated.

It contains the initializations of work areas.

This function triggers the procedure to be executed in case of error.

It ensures the branching to the physical display function after consultation of HELP documentation (if a documentation Help character has been entered on the Screen Definition screen).

It ensures the positioning of the cursor for the first display.

GENERATED PROGRAM: PROCEDURE DIVISION
INITIALIZATIONS (F01)

PAGE

78

4
2

```

PROCEDURE DIVISION.                                *99999
*          *****                                DO0030
*          *                                     DO0030
*          *   INITIALIZATIONS                   *   DO0030
*          *                                     *   DO0030
*          *****                                DO0030
F01.          EXIT.                                DO0030
F0105.        DO0030
              CALL 'D$INIT' USING STATUS-WORD, INFO-BUFFER. DO0030
              IF STATUS-FATAL      GO TO F81ER.           DO0030
              IF INFO-PREVIOUS-PROGRAM-ID NOT = SPACE   DO0030
                AND INFO-PREVIOUS-PROGRAM-ID NOT = 'LOGON' DO0030
              CALL 'D$GETSCR' USING STATUS-WORD, COMMON-AREA. DO0030
              IF STATUS-FATAL      GO TO F81ER.           DO0030
F0105-FN.     EXIT.                                DO0030
F0110.        DO0030
              ACCEPT TIMCO FROM TIME.                    DO0030
              ACCEPT DATOR FROM DATE.                    DO0030
              MOVE ZERO TO CATX FT K50L.                  DO0030
              MOVE '1' TO ICF OCF SCR-ER.                DO0030
              MOVE ZERO TO VALIDATION-TABLE-FIELDS.     DO0030
              MOVE SPACE TO CATM OPER OPERD CAT-ER.     DO0030
              MOVE SPACE TO TABLE-OF-ATTRIBUTES.       DO0030
              MOVE ZERO TO CONFIGURATIONS.               DO0030
              IF INFO-PREVIOUS-PROGRAM-ID = SPACE       DO0030
                OR INFO-PREVIOUS-PROGRAM-ID = 'LOGON'   DO0030
              MOVE '0' TO K-S0030-DOC.                   DO0030
              IF INFO-CONVERSATION NOT = 'Y'            DO0030
                MOVE ZERO TO ICF                         DO0030
              CALL 'D$OPEN'   USING STATUS-WORD           DO0030
              SCREEN-DO0030-130.                          DO0030
              IF STATUS-FATAL      GO TO F81ER.           DO0030
              MOVE SPACE TO I-0030   O-0030 ERROR-MESSAGE. DO0030
              IF ICF = ZERO PERFORM F8115 THRU F8115-FN. DO0030
                IF K-S0030-DOC = '2' OR K-S0030-DOC = '3' DO0030
              MOVE '1' TO K-S0030-DOC   GO TO F8Z05.     DO0030
              MOVE 'X' TO DE-AT (4, 010).                DO0030
              MOVE SPACE          TO          O-0030-ERMSG (01). DO0030
F0110-FN.     EXIT.                                DO0030
*          +-----+                               P000
* LEVEL 10    I OPEN DATABASE                       I   P000
*          +-----+                               P000
F0115.        P000
              MOVE          'F0115' TO 7-WW00-FONCT     P100
              MOVE          'IMPART' TO 7-WW00-ORDRE    P120
              IMPART ON ERROR GO TO F99RB.              P140
              MOVE          'OPEN' TO 7-WW00-ORDRE      P200
              OPEN WWA21E  USAGE-MODE IS                P220
                RETRIEVAL                               P230
                WWA81E  USAGE-MODE IS                  P240
                RETRIEVAL                               P250
                WWA20E  USAGE-MODE IS                  P260
                UPDATE.                                 P270
              IF ERROR-CODE NOT = ZERO                  P300
              GO TO F99RB.                               P300
F0115-FN.     EXIT.                                P000
F0160.        DO0030
              IF ICF = ZERO MOVE 'A' TO OPER            DO0030
              GO TO F3999-ITER-FT.                      DO0030
F0160-FN.     EXIT.                                DO0030
F01-FN.       EXIT.                                DO0030
*          +-----+                               P000
* LEVEL 10    I INIT. NUMBER OF LOADED ITEMS        I   P000
*          +-----+                               P000
F02CP.        P000
              MOVE          IWP20M TO IWP20L.          P100
F02CP-FN.     EXIT.                                P000

```

4.3. RECEPTION AND OPERATION CODE (F05)

F05: RECEPTION AND OPERATION CODE

The F05 paragraph contains the conditions for all the procedures which have to do with the 'RECEPTION' part of the program: from F05 to END-OF-RECEPTION (F45-FN).

In general, all the automatic functions in this part of the program are generated if at least one variable data element (NATURE = 'V') is defined on the screen.

Sub-function F0510 contains the 'SCREEN RECEPTION' procedure upon entry into the program.

If an initialization character is entered on the Screen Definition screen, it is set to blanks (except when a branch to a documentation screen is executed).

Sub-function F0512 is generated if a documentation call is entered in the fields that are necessary for branching to the documentation screen.

Sub-function F0520 is generated if a variable data element of the screen is defined as an Operation Code on the Screen Call of Elements (-CE) screen.

The internal Operation Code 'OPER' is set according to:

- the value of the screen data element defined as an Operation Code (values specified with TYPE OF LINE = 'O' on the Data Element Description (-D) screen;

- The value of the 'PFKEY' data element: this value is entered on the Screen Call of Elements (-CE).

If an error is found in the value of the Operation Code, following 'RECEPTION' procedures are not executed.

GENERATED PROGRAM: PROCEDURE DIVISION
 RECEPTION AND OPERATION CODE (F05)

PAGE

80

4
3

```

*          *****
*          *
*          * RECEPTION
*          *
*          *****
F05.  IF ICF = ZERO GO TO END-OF-RECEPTION.
F0510.
      MOVE ZERO TO K-S0030-ERCOD.
      MOVE INFO-FUNCTION-KEY TO I-FONCT.
      IF I-PFKEY NOT = ZERO GO TO F0510-FN.
      CALL 'D$READ' USING STATUS-WORD
          SCREEN-DO0030-130.
      IF STATUS-FATAL GO TO F81ER.
      MOVE SCREEN-DO0030-130-DATA TO
          O-0030
      MOVE 'A' TO OPER MOVE SPACE TO OPERD.
      PERFORM F8150 THRU F8150-FN.
      IF K-S0030-ERCOD = ZERO
      INSPECT I-0030 REPLACING ALL '_' BY SPACE.
F0510-FN.  EXIT.
F0512.  IF K-S0030-ERCOD NOT = ZERO
      NEXT SENTENCE ELSE GO TO F0512-FN.
      MOVE '2' TO K-S0030-DOC
      MOVE PROGE TO K-S0030-PROGE
      MOVE LIBRA TO K-S0030-LIBRA.
      IF K-S0030-ERCOD NOT = SPACE
      MOVE '3' TO K-S0030-DOC.
      PERFORM F80-HELP-R THRU F80-FN
      PERFORM F8130 THRU F8130-FN
      PERFORM F80-HELP-RW THRU F80-FN
      MOVE PRDOC TO 5-0030-PROGE K-S0030-PROHE
      MOVE 'O' TO OPER GO TO F4040.
F0512-FN.  EXIT.
*          *****
*          *
*          * VALIDATION OF OPERATION CODE
*          *
*          *****
F0520.
      IF I-0030-CHOIX = '1'
      MOVE 'DO0000' TO 5-0030-PROGE
      MOVE 'O' TO OPER GO TO F40-A.
      IF I-0030-CHOIX = '2'
      MOVE 'DO0010' TO 5-0030-PROGE
      MOVE 'O' TO OPER GO TO F40-A.
      IF I-0030-CHOIX = '3'
      MOVE 'DO0020' TO 5-0030-PROGE
      MOVE 'O' TO OPER GO TO F40-A.
      IF I-0030-CHOIX = '4'
      MOVE 'DO0040' TO 5-0030-PROGE
      MOVE 'O' TO OPER GO TO F40-A.
      IF I-0030-CHOIX = '5'
      MOVE 'DO0050' TO 5-0030-PROGE
      MOVE 'O' TO OPER GO TO F40-A.
      IF I-0030-CHOIX = '0'
      MOVE 'DO0070' TO 5-0030-PROGE
      MOVE 'O' TO OPER GO TO F40-A.
      IF I-0030-CHOIX = '7'
      MOVE 'M' TO OPER GO TO F0520-900.
      IF I-0030-CHOIX = '8'
      MOVE 'S' TO OPER GO TO F0520-900.
      MOVE '5' TO ER-0030-CHOIX MOVE '4' TO SCR-ER
      GO TO F3999-ITER-FT.
F0520-900.
      IF OPER NOT = 'A' AND OPER NOT = 'M' AND OPER NOT = 'O'
      GO TO F3999-ITER-FT.
F0520-FN.  EXIT.
F05-FN.  EXIT.
*          +-----+
* LEVEL 10  I NO UPDATE ==> END OF RECEIVE I
*          +-----+
F08BB.  IF OPER NOT = 'M'
      NEXT SENTENCE ELSE GO TO F08BB-FN.
      GO TO F3999-ITER-FT.
F08BB-FN.  EXIT.

```


4.4. CATEGORY PROCESSING LOOP (F10)

F10 : CATEGORY POSITIONING

The CATEGORY POSITIONING function positions the category to be processed in 'RECEPTION' using the CATX indicator which may be set to one of the following values:

'0' Beginning of RECEPTION
' ' Screen-top category
'R' Repetitive category
'Z' Screen-bottom category

Procedures are generated according to the categories defined on the Screen Call of Elements ('-CE') screen.

If no category has been defined, the screen is considered to be a screen-top category.

For the repetitive category, this function includes the interaction between the line of the category to be processed and the input screen description field used to access each of the data elements on the line.

This function also includes the initialization and incrementation of the ICATR index, which manages the repetitive category.

If an error is detected (CAT-ER = 'E') once the processing of a category is complete (F15 to F3999-ITER-FI), SCR-ER is set and validation processing on the subsequent categories is not executed.

GENERATED PROGRAM: PROCEDURE DIVISION
 CATEGORY PROCESSING LOOP (F10)

PAGE

82

4
4

```

*          *****
*          *
*          *   CATEGORY PROCESSING LOOP   *
*          *
*          *****
F10.      EXIT.
F1010.    MOVE SPACE TO CATM.
          IF CATX = 'R'
MOVE      O-0030-LINE          TO
          P-0030-LINE          (ICATR)
MOVE      A-0030-LINE          (1) TO
          B-0030-LINE          (1, ICATR)
MOVE      A-0030-LINE          (2) TO
          B-0030-LINE          (2, ICATR)
MOVE      A-0030-LINE          (3) TO
          B-0030-LINE          (3, ICATR)
MOVE      A-0030-LINE          (4) TO
          B-0030-LINE          (4, ICATR)
MOVE      I-0030-LINE          TO
          J-0030-LINE          (ICATR)
MOVE      ER-0030-LINE          TO
          PS-30-LINE          (ICATR).
IF CAT-ER = 'E' MOVE '4' TO SCR-ER GO TO F3999-ITER-FT.
MOVE SPACE TO CAT-ER.
IF CATX = '0' MOVE ' ' TO CATX GO TO F1010-FN.
IF CATX = ' ' MOVE 'R' TO CATX MOVE ZERO TO ICATR.
IF CATX = 'R' AND ICATR < IRR ADD 1 TO ICATR
MOVE      PS-30-LINE          (ICATR) TO
          ER-0030-LINE
MOVE      B-0030-LINE          (4, ICATR) TO
          A-0030-LINE          (4)
MOVE      P-0030-LINE          (ICATR) TO
          O-0030-LINE
MOVE      J-0030-LINE          (ICATR) TO
          I-0030-LINE          GO TO F1010-FN.
IF CATX = 'R' MOVE 'Z' TO CATX GO TO F1010-FN.
F1010-A.  GO TO F3999-ITER-FT.
F1010-FN. EXIT.
F10-FN.   EXIT.

```

4.5. VALIDATION OF TRANSACTION CODE (F15)

F15 : TRANSACTION CODE POSITIONING

The VALIDATION OF TRANSACTION CODE (F15) function is generated if at least one Data Element is defined as a Transaction Code in a category on the Screen Call of Elements ('-CE') screen.

The internal transaction code (CATM) is set according to the Data Element's value that is defined as a Transaction Code for the category. The value can be given to the Data Element on:

- . the Data Element Description (-D) screen with TYPE OF LINE = 'I',
- . the Screen Call of Elements (-CE) screen in the Transaction Code Data Element call line.

Depending on the categories defined on the screen (and for which a transaction code is indicated) the F15 function includes the following:

- .F15A for the screen-top category,
- .F15R for the repetitive category,
- .F15Z for the screen-bottom category.

If the transaction code is wrong, the subsequent 'RECEPTION' procedures are not executed.

4.6. DATA ELEMENT VALIDATION (F20)

F20 : DATA ELEMENT VALIDATION

The DATA ELEMENT VALIDATION (F20) function is generated when one variable Data Element has been specified on the screen.

Depending on which category or categories defined on the screen contain at least one Data Element to be validated, the F20 function includes the following:

- . F20A for the screen-top category.
- . F20R for the repetitive category.
- . F20Z for the screen-bottom category.

The procedure for each category contains one sub-function per Data Element to be validated. The validation procedures are the following:

- . Presence validation.
- . Numeric class validation.
- . Value validation according to the values or value ranges defined on the Data Element Description ('-D') screen, or on the Screen Call of Elements ('-CE') screen.
- . Validation of date (via PERFORM) for Data Elements defined with a 'DATE' format.
- . Validation of a sub-function (via PERFORM) defined by the user.

The conditioning of each sub-function is generated based on the procedure option of the Data Element.

The validation result for each Data Element is stored in a field coded ER-scrn-delcod (scrn: last four characters of the screen code; delcod: Data Element code), which takes the following values:

```
'0' : Data Element absent  
'1' : Data Element present  
'2' : invalid absence  
'4' : invalid class  
'5' : invalid value
```

'CAT-ER' is set when any Data Element (or user) error is detected.

GENERATED PROGRAM: PROCEDURE DIVISION
DATA ELEMENT VALIDATION (F20)

PAGE

86

4
6

NOTE: Sub-functions are numbered based on the number of Data Elements, their position on the screen, etc.

As a result, direct references should never be made to a label generated in specific procedures.

Use the Relative Positioning types *A, *P, and *R (see chapter "USE OF STRUCTURED CODE" in the ON-LINE SYSTEMS DEVELOPMENT Reference Manual).

GENERATED PROGRAM: PROCEDURE DIVISION
 DATA ELEMENT VALIDATION (F20)

PAGE

87

4
6

```

*          *****
*          *
*          *   DATA ELEMENT VALIDATION   *
*          *
*          *****
F20.          EXIT.
F20A.  IF CATX NOT = ' ' GO TO F20A-FN.
F20A2.          IF I-0030-CHOIX NOT = SPACE
              MOVE '1' TO ER-0030-CHOIX.
F20A2-FN.      EXIT.
F20B1.          IF I-0030-MATE NOT = SPACE
              MOVE '1' TO ER-0030-MATE.
              IF ER-0030-MATE NOT = 1
                  GO TO F20B1-FN.
              IF I-0030-MATE = 'I1'
              OR I-0030-MATE = 'I2'
              OR I-0030-MATE = 'I3'
              OR I-0030-MATE = 'I4'
              OR I-0030-MATE = 'I5'
              OR I-0030-MATE = 'B7'
              OR I-0030-MATE = 'B8'
              OR I-0030-MATE = 'UN'
              OR I-0030-MATE = 'IC'
              OR I-0030-MATE = 'IBM.V.OS'
              OR I-0030-MATE = 'IBM.V.DO'
              OR I-0030-MATE = 'IBM.D.OS'
              OR I-0030-MATE = 'IBM.D.DO'
              OR I-0030-MATE = 'IBM.IMS '
              OR I-0030-MATE = 'DPS7 '
              OR I-0030-MATE = 'DPS8 '
              OR I-0030-MATE = 'UNISYS '
              OR I-0030-MATE = 'ICL '
              OR I-0030-MATE = 'SPECIAL'
              NEXT SENTENCE ELSE
              MOVE '5' TO ER-0030-MATE.
              IF ER-0030-MATE > '1'
                  MOVE 'E' TO CAT-ER          GO TO F20B1-FN.
F20B1-FN.      EXIT.
F20B2.          IF I-0030-RELEA NOT = SPACE
              MOVE '1' TO ER-0030-RELEA
              ELSE
              MOVE '2' TO ER-0030-RELEA
              MOVE 'E' TO CAT-ER          GO TO F20B2-FN.
              IF I-0030-RELEA = '7.2'
              OR I-0030-RELEA = '7.3'
              OR I-0030-RELEA = '8.0'
              NEXT SENTENCE ELSE
              MOVE '5' TO ER-0030-RELEA.
              IF ER-0030-RELEA > '1'
                  MOVE 'E' TO CAT-ER          GO TO F20B2-FN.
F20B2-FN.      EXIT.
F20B5.          IF I-0030-RUE NOT = SPACE
              MOVE '1' TO ER-0030-RUE.
F20B5-FN.      EXIT.
F20B7.          IF I-0030-COPOS NOT = SPACE
              MOVE '1' TO ER-0030-COPOS
              ELSE
              MOVE '2' TO ER-0030-COPOS
              MOVE 'E' TO CAT-ER          GO TO F20B7-FN.
              MOVE I-0030-COPOS TO WP30-COPOS
              MOVE ER-0030-COPOS TO DEL-ER
              PERFORM F93CP THRU F93CP-FN
              MOVE WP30-COPOS TO
                  I-0030-COPOS
              MOVE DEL-ER TO ER-0030-COPOS.
              IF ER-0030-COPOS > '1'
                  MOVE 'E' TO CAT-ER          GO TO F20B7-FN.
F20B7-FN.      EXIT.
F20B8.          IF I-0030-REFCLI NOT = SPACE
              MOVE '1' TO ER-0030-REFCLI.
F20B8-FN.      EXIT.

```

GENERATED PROGRAM: PROCEDURE DIVISION
 DATA ELEMENT VALIDATION (F20)

PAGE

88

4
6

```

F20B9.                                DO0030
      IF I-0030-DATE NOT = SPACE      DO0030
      MOVE '1' TO ER-0030-DATE        DO0030
      ELSE                             DO0030
      MOVE '2' TO ER-0030-DATE        DO0030
      MOVE 'E' TO CAT-ER              DO0030
      MOVE I-0030-DATE TO DAT7        DO0030
      PERFORM F8120-D THRU F8120-FN  DO0030
      MOVE DEL-ER TO ER-0030-DATE    DO0030
      IF DEL-ER > '1' MOVE 'E' TO CAT-ER GO TO F20B9-FN. DO0030
F20B9-FN. EXIT.                      DO0030
F20C0.                                DO0030
      IF I-0030-CORRES NOT = SPACE    DO0030
      MOVE '1' TO ER-0030-CORRES.     DO0030
      IF ER-0030-CORRES NOT = 1      DO0030
      GO TO F20C0-FN.                DO0030
F20C0-FN. EXIT.                      DO0030
F20C1.                                DO0030
      IF E-0030-REMIS NOT = SPACE     DO0030
      MOVE '1' TO ER-0030-REMIS.     DO0030
      MOVE E-0030-REMIS TO ZONUM1    DO0030
      MOVE 9-0030-REMIS TO NUMPIC    DO0030
      MOVE ER-0030-REMIS TO DEL-ER   DO0030
      PERFORM F8110 THRU F8110-FN    DO0030
      MOVE DEL-ER TO ER-0030-REMIS  DO0030
      IF DEL-ER > 1 MOVE 'E' TO CAT-ER GO TO F20C1-FN. DO0030
      MOVE ZONUM2 TO E-0030-REMIS.   DO0030
      IF DEL-ER = '1'                DO0030
      MOVE I-0030-REMIS TO O-0030-REMIS. DO0030
F20C1-FN. EXIT.                      DO0030
F20A-FN. EXIT.                      DO0030
F20R. IF CATX NOT = 'R' GO TO F20R-FN. DO0030
F20C3.                                DO0030
      IF I-0030-CODMVT NOT = SPACE    DO0030
      MOVE '1' TO ER-0030-CODMVT.    DO0030
F20C3-FN. EXIT.                      DO0030
*          +-----+
* LEVEL 10 I ITEM NOT AVAILABLE I P000
*          +-----+
F20BB.                                P000
      IF I-0030-FOURNI = 'CLA'       P100
      AND CATM NOT = SPACE           P110
      MOVE 'A' TO ER-0030-FOURNI     P100
      MOVE 'E' TO CAT-ER             P100
      GO TO F20C4-FN.                P110
F20BB-FN. EXIT.                     P000
F20C4.                                DO0030
      IF CATM = SPACE                 DO0030
      IF I-0030-FOURNI NOT = SPACE    DO0030
      MOVE '1' TO ER-0030-FOURNI     DO0030
      ELSE                             DO0030
      MOVE '2' TO ER-0030-FOURNI     DO0030
      MOVE 'E' TO CAT-ER              DO0030
      GO TO F20C4-FN.                DO0030
      IF I-0030-FOURNI = 'DIC'       DO0030
      OR I-0030-FOURNI = 'MER'       DO0030
      OR I-0030-FOURNI = 'TAB'       DO0030
      OR I-0030-FOURNI = 'DBD'       DO0030
      OR I-0030-FOURNI = 'DSO'       DO0030
      OR I-0030-FOURNI = 'LGS'       DO0030
      OR I-0030-FOURNI = 'LGB'       DO0030
      OR I-0030-FOURNI = 'DLG'       DO0030
      NEXT SENTENCE ELSE             DO0030
      MOVE '5' TO ER-0030-FOURNI.    DO0030
      IF ER-0030-FOURNI > '1'        DO0030
      MOVE 'E' TO CAT-ER              DO0030
      GO TO F20C4-FN.                DO0030
F20C4-FN. EXIT.                      DO0030
F20C5.                                DO0030
      IF CATM = 'A' OR CATM = SPACE   DO0030
      IF E-0030-QTMAC NOT = SPACE    DO0030
      MOVE '1' TO ER-0030-QTMAC      DO0030
      ELSE                             DO0030
      MOVE '2' TO ER-0030-QTMAC      DO0030
      MOVE 'E' TO CAT-ER              DO0030
      GO TO F20C5-FN.                DO0030
      MOVE E-0030-QTMAC TO ZONUM1    DO0030
      MOVE 9-0030-QTMAC TO NUMPIC    DO0030
      MOVE ER-0030-QTMAC TO DEL-ER   DO0030
      PERFORM F8110 THRU F8110-FN    DO0030
  
```


GENERATED PROGRAM: PROCEDURE DIVISION
DATA ELEMENT VALIDATION (F20)

PAGE

89

4
6

```
MOVE DEL-ER TO ER-0030-QTMAC DO0030
IF DEL-ER > 1 MOVE 'E' TO CAT-ER GO TO F20C5-FN. DO0030
MOVE ZONUM2 TO E-0030-QTMAC. DO0030
IF DEL-ER = '1' DO0030
MOVE I-0030-QTMAC TO O-0030-QTMAC. DO0030
IF I-0030-QTMAC NOT < 01 DO0030
AND I-0030-QTMAC NOT > 50 DO0030
NEXT SENTENCE ELSE DO0030
MOVE '5' TO ER-0030-QTMAC. DO0030
IF ER-0030-QTMAC > '1' DO0030
MOVE 'E' TO CAT-ER GO TO F20C5-FN. DO0030
F20C5-FN. EXIT. DO0030
F20C8. DO0030
IF CATM = 'A' OR CATM = SPACE GO TO F20C8-FN. DO0030
IF I-0030-INFOR NOT = SPACE DO0030
MOVE '1' TO ER-0030-INFOR. DO0030
IF ER-0030-INFOR NOT = 1 DO0030
GO TO F20C8-FN. DO0030
F20C8-FN. EXIT. DO0030
F20R-FN. EXIT. DO0030
F20Z. IF CATX NOT = 'Z' GO TO F20Z-FN. DO0030
F20D0. DO0030
IF I-0030-EDIT NOT = SPACE DO0030
MOVE '1' TO ER-0030-EDIT. DO0030
F20D0-FN. EXIT. DO0030
F20Z-FN. EXIT. DO0030
F20-FN. EXIT. DO0030
```

4.7. SEGMENT ACCESS FOR VALIDATION (F25)

F25 : SEGMENT ACCESS FOR VALIDATION

The SEGMENT ACCESS FOR VALIDATION (F25) function is generated when there is at least one segment to be accessed in RECEPTION.

Depending on which categories defined on the screen contain a segment to be accessed in RECEPTION, the F25 function includes the following:

- . F25A for the screen-top category.
- . F25R for the repetitive category.
- . F25Z for the screen-bottom category.

In the processing for each category there is one sub-function per segment to be accessed, including:

- . The initialization of the key (if indicated on the -CS)
- . Read or Read with Segment Update depending on its use in the screen (by a PERFORM of F80-ddss-R or RU)
- . Positioning of the segment ddss-CF variable (1 if OK)
- . Error processing, if any.

Within a category, accesses are generated in the alphabetical order of the segment codes, except for segments which contain a 'preceding' segment.

If a segment is to be updated, its access depends on the CATM value. It is not performed if CATM = SPACE.

If a segment has a preceding segment, its access is performed if the ddss-CF variable of the preceding segment is equal to '1'.

Other types of reads are not conditioned.

Sub-function F2599 is generated if at least one of the Read segments can be updated.

It contains the PERFORM of functions F80-ddss-UN, according to the segments used, as well as cursor positioning on the first variable data element of the category, in the case of segment error.

GENERATED PROGRAM: PROCEDURE DIVISION
SEGMENT ACCESS FOR VALIDATION (F25)

PAGE

91

4
7

NOTE: Sub-functions are numbered based on the number of segments, their positions on the '-CS' screen, etc. As a result, a direct reference should never be made to a generated label in the specific procedures.

Use the Relative Positioning types '*A', '*P' and '*R' (see chapter "USE OF STRUCTURED CODE" in the ON-LINE SYSTEMS DEVELOPMENT Reference Manual).

GENERATED PROGRAM: PROCEDURE DIVISION
 SEGMENT ACCESS FOR VALIDATION (F25)

PAGE

92

4
7

```

*          *****
*          *
*          *   SEGMENT ACCESS FOR VALIDATION   *
*          *
*          *****
F25.      IF CAT-ER NOT = SPACE GO TO F25-FN.
F25A.    IF CATX NOT = ' ' GO TO F25A-FN.
F2501.
  MOVE '0' TO CD05-CF.
  IF CATM = SPACE          GO TO F2501-FN.
  MOVE SPACES              TO CD00-KEYCD
  MOVE 'B'                 TO CD00-COCARA
  MOVE CA00-NUCOM          TO CD00-NUCOM
  PERFORM F80-CD05-RU THRU F80-FN.
  IF IK = '0'
  MOVE '1' TO CD05-CF.
  IF CATM NOT = 'C' AND IK = '1'
    MOVE 'F019' TO XERCD
    PERFORM F81UT          GO TO F2501-FN.
F2501-FN.  EXIT.
F25A-FN.  EXIT.
F25R.    IF CATX NOT = 'R' GO TO F25R-FN.
F2502.
  MOVE '0' TO CD10-CF.
  IF CATM = SPACE          GO TO F2502-FN.
  MOVE 'C'                 TO CD00-KEYCD
  MOVE CA00-NUCOM          TO CD00-NUCOM
  MOVE I-0030-FOURNI      TO CD00-FOURNI
  PERFORM F80-CD10-RU THRU F80-FN.
  IF IK = '0'
  MOVE '1' TO CD10-CF.
  IF CATM = 'X' AND IK = '1' MOVE 'C' TO CATM.
  IF CATM = 'X' AND IK = '0' MOVE 'M' TO CATM.
  IF CATM = 'C' AND IK = '0'
    MOVE 'F028' TO XERCD
    PERFORM F81UT          GO TO F2502-FN.
  IF CATM NOT = 'C' AND IK = '1'
    MOVE 'F029' TO XERCD
    PERFORM F81UT          GO TO F2502-FN.
*          +-----+
* LEVEL 12  I ACCESS TO FO10          I
*          +-----+
F25BB.
  MOVE '1' TO CD10-CF.
F25BB-FN.  EXIT.
F2502-FN.  EXIT.
F2503.
  MOVE '0' TO FO10-CF.
  IF CD10-CF NOT = '1' GO TO F2503-FN.
  IF CATM = SPACE          GO TO F2503-FN.
  MOVE I-0030-FOURNI      TO FO00-CLEFO
  MOVE CA00-LANGU         TO FO00-LANGU
  MOVE I-0030-RELEA       TO FO00-RELEA
  MOVE I-0030-MATE        TO FO00-MATE
  PERFORM F80-FO10-RU THRU F80-FN.
  IF IK = '0'
  MOVE '1' TO FO10-CF.
  IF IK = '1' MOVE 'F039' TO XERCD
    PERFORM F81UT          GO TO F2503-FN.
F2503-FN.  EXIT.
F25R-FN.  EXIT.
F25Z.    IF CATX NOT = 'Z' GO TO F25Z-FN.
F2505.
  MOVE '0' TO CD20-CF.
  IF CATM = SPACE          GO TO F2505-FN.
  MOVE SPACES              TO CD00-KEYCD
  MOVE 'E'                 TO CD00-COCARA
  MOVE CA00-NUCOM          TO CD00-NUCOM
  PERFORM F80-CD20-RU THRU F80-FN.
  IF IK = '0'
  MOVE '1' TO CD20-CF.
  IF CATM = 'X' AND IK = '1' MOVE 'C' TO CATM.
  IF CATM = 'X' AND IK = '0' MOVE 'M' TO CATM.
  IF CATM = 'C' AND IK = '0'
    MOVE 'F058' TO XERCD
    PERFORM F81UT          GO TO F2505-FN.
  IF CATM NOT = 'C' AND IK = '1'

```

GENERATED PROGRAM: PROCEDURE DIVISION
 SEGMENT ACCESS FOR VALIDATION (F25)

PAGE

93

4
7

	MOVE 'F059' TO XERCD	DO0030
	PERFORM F81UT GO TO F2505-FN.	DO0030
F2505-FN.	EXIT.	DO0030
F25Z-FN.	EXIT.	DO0030
F2599.	IF CAT-ER = SPACE GO TO F2599-FN.	DO0030
	IF CD05-CF = '1'	DO0030
	PERFORM F80-CD05-UN THRU F80-FN.	DO0030
	IF CD10-CF = '1'	DO0030
	PERFORM F80-CD10-UN THRU F80-FN.	DO0030
	IF FO10-CF = '1'	DO0030
	PERFORM F80-FO10-UN THRU F80-FN.	DO0030
	IF CD20-CF = '1'	DO0030
	PERFORM F80-CD20-UN THRU F80-FN.	DO0030
	IF CATX = ' ' AND DE-AT (4, 010) = 'X'	DO0030
	MOVE ' ' TO DE-AT (4, 010).	DO0030
	IF CATX = ' '	DO0030
	MOVE 'X' TO A-0030-CHOIX (4).	DO0030
	IF CATX = 'R' AND DE-AT (4, 010) = 'X'	DO0030
	MOVE ' ' TO DE-AT (4, 010).	DO0030
	IF CATX = 'R'	DO0030
	MOVE 'X' TO A-0030-CODMVT (4).	DO0030
	IF CATX = 'Z' AND DE-AT (4, 010) = 'X'	DO0030
	MOVE ' ' TO DE-AT (4, 010).	DO0030
	IF CATX = 'Z'	DO0030
	MOVE 'X' TO A-0030-EDIT (4).	DO0030
F2599-FN.	EXIT.	DO0030
F25-FN.	EXIT.	DO0030
*	+-----+	P000
* LEVEL 10	I STOCK UPD.: ORDER DELETION/UPD I	P000
*	+-----+	P000
F28BH.	IF (CATM = 'A' OR 'M')	P000
	AND CATX = 'R'	P100
	AND CAT-ER = SPACES	P120
	NEXT SENTENCE ELSE GO TO F28BH-FN.	P120
ADD	CD10-QTMAL TO FO10-QTMAS.	P100
F28BH-FN.	EXIT.	P000

4.8. DATA ELEMENT TRANSFER (F30)

F30: DATA ELEMENT TRANSFER

The DATA ELEMENT TRANSFER (F30) function ensures the transfer of Data Elements on the screen to the corresponding Data Elements in the Segments.

Depending on which categories defined on the screen contain at least one Data Element transfer on reception, the F30 function includes the following:

- . F30A for the screen-top category.
- . F30R for the repetitive category.
- . F30Z for the screen-bottom category.

The condition of the transfer is generated based on the use of the Segment on reception, or the value of the PRESENCE VALIDATION OF DATA ELEMENT field on the Screen Call of Elements ('-CE') screen.

GENERATED PROGRAM: PROCEDURE DIVISION
 DATA ELEMENT TRANSFER (F30)

PAGE

95

4
8

```

*          *****
*          *
*          * DATA ELEMENT TRANSFER *
*          *
*          *****
F30.      IF CAT-ER NOT = SPACE GO TO F30-FN.
F30A.    IF CATX NOT = ' ' GO TO F30A-FN.
          MOVE I-0030-RELEA TO CD05-RELEA.
          MOVE I-0030-COPOS TO CD05-COPOS.
          MOVE I-0030-REFCLI TO CD05-REFCLI.
          MOVE I-0030-DATE TO CD05-DATE.
          MOVE I-0030-REMIS TO CD05-REMIS.
          IF ER-0030-MATE = '1'
          MOVE I-0030-MATE TO CD05-MATE.
          IF ER-0030-CORRES = '1'
          MOVE I-0030-CORRES TO CD05-CORRES.
F30A-FN. EXIT.
F30R.    IF CATX NOT = 'R' GO TO F30R-FN.
          IF ER-0030-INFOR = '1'
          MOVE I-0030-INFOR TO CD10-INFOR.
          IF CATM NOT = SPACE
          MOVE I-0030-FOURNI TO CD00-FOURNI.
          IF CATM NOT = SPACE AND CATM NOT = 'A'
          MOVE I-0030-QTMAC TO CD10-QTMAC
          ADD I-0030-QTMAC TO FO10-QTMAM.
*          +-----+
* LEVEL 10 I QUANTITY PROCESSING I
*          +-----+
F30BD.
*          +-----+
* LEVEL 12 I CALC. DELIV. QUANT. STOCK UPD. I
*          +-----+
F30BF.   IF CATM = 'C' OR 'M'
          NEXT SENTENCE ELSE GO TO F30BF-FN.
          IF FO10-QTMAS NOT <
          I-0030-QTMAC
          MOVE I-0030-QTMAC TO CD10-QTMAL
          ELSE
          MOVE FO10-QTMAS TO CD10-QTMAL.
          SUBTRACT CD10-QTMAL FROM FO10-QTMAS
          MOVE CD10-QTMAL TO O-0030-QTMAL.
F30BF-FN. EXIT.
F30BD-FN. EXIT.
F30R-FN. EXIT.
F30Z.   IF CATX NOT = 'Z' GO TO F30Z-FN.
          MOVE I-0030-EDIT TO CD20-EDIT.
F30Z-FN. EXIT.
F30-FN. EXIT.

```

4.9. SEGMENT ACCESS FOR UPDATE (F35)

F35: SEGMENT ACCESS FOR UPDATE

This function ensures Segment updates. If an error has been detected by the error checks (CAT-ER), this function is not executed.

Depending on which categories contain a Segment to be updated, the SEGMENT ACCESS FOR UPDATE (F35) function includes the following:

- . F35A for the screen-top category.
- . F35R for the repetitive category.
- . F35Z for the screen-bottom category.

In the processing for each category there is one sub-function per Segment to be updated, possibly including several types of access.

The function is accessed by executing a PERFORM of the appropriate subfunction in F80.

For a Segment that does not follow an access to another Segment (i.e. the PRECEDING SEGMENT field in the Screen Call of Segments ('-CS') screen is left blank), access is conditioned by the value of the internal Transaction Code (CATM) found in the category, which corresponds to one of the following operations:

- . Creation: writing (F80-ddss-R).
- . Deletion: suppression (F80-ddss-D).
- . Other cases: rewriting (F80-ddss-RW)

The user must manage the access to other transactions if the rewrite option does not correspond to user needs.

For a Segment that follows an access to another Segment (i.e. a Segment is listed in the PRECEDING SEGMENT field on the Screen Call of Segments ('-CS') screen), access is conditioned by the Segment configuration, which is either:

- . ddss-CF = 0, writing, or
- . ddss-CF = 1, rewriting.

If a Data Element was defined as a Transaction Code on the Screen Call of Elements ('-CE') screen (in the VALIDATION CONDITIONS/SET VARIABLES field), it is set to blanks.

Paragraph F3999-ITER-FI returns to the beginning of the 'RECEPTION' iteration.

NOTE: Sub-functions are numbered based on the number of segments, their positions on the '-CS' screen, etc. As a result, a direct reference should never be made to a generated label in the specific procedures.

Use the Relative Positioning types '*A', '*P' and '*R' (see chapter "USE OF STRUCTURED CODE" in the ON-LINE SYSTEMS DEVELOPMENT Reference Manual.)

GENERATED PROGRAM: PROCEDURE DIVISION
SEGMENT ACCESS FOR UPDATE (F35)

PAGE

98

4
9

```
*          *****  
*          *                               *          DO0030  
*          * SEGMENT ACCESS FOR UPDATE   *          DO0030  
*          *                               *          DO0030  
*          *****  
*          *****  
F35.      IF CAT-ER NOT = SPACE OR CATM = SPACE GO TO F35-FN. DO0030  
F35A.     IF CATX NOT = ' ' GO TO F35A-FN. DO0030  
F3501.    DO0030  
          IF CATM NOT = 'C' AND CATM NOT = 'A' DO0030  
          PERFORM F80-CD05-RW THRU F80-FN. DO0030  
F3501-FN. EXIT. DO0030  
F35A-FN.  EXIT. DO0030  
F35R.     IF CATX NOT = 'R' GO TO F35R-FN. DO0030  
F3502.    DO0030  
          IF CATM = 'C' DO0030  
          PERFORM F80-CD10-W THRU F80-FN. DO0030  
          IF CATM = 'A' DO0030  
          PERFORM F80-CD10-D THRU F80-FN. DO0030  
          IF CATM NOT = 'C' AND CATM NOT = 'A' DO0030  
          PERFORM F80-CD10-RW THRU F80-FN. DO0030  
F3502-FN. EXIT. DO0030  
F3503.    DO0030  
          IF          FO10-CF = '1' DO0030  
          PERFORM F80-FO10-RW THRU F80-FN. DO0030  
F3503-FN. EXIT. DO0030  
F35R-C3.  MOVE SPACE TO O-0030-CODMVT. DO0030  
F35R-FN.  EXIT. DO0030  
F35Z.     IF CATX NOT = 'Z' GO TO F35Z-FN. DO0030  
F3505.    DO0030  
          IF CATM = 'C' DO0030  
          PERFORM F80-CD20-W THRU F80-FN. DO0030  
          IF CATM NOT = 'C' AND CATM NOT = 'A' DO0030  
          PERFORM F80-CD20-RW THRU F80-FN. DO0030  
F3505-FN. EXIT. DO0030  
F35Z-D0.  MOVE SPACE TO O-0030-EDIT. DO0030  
F35Z-FN.  EXIT. DO0030  
F35-FN.   EXIT. DO0030  
F3999-ITER-FI. GO TO F10. DO0030  
F3999-ITER-FT. EXIT. DO0030  
F3999-FN. EXIT. DO0030
```

4.10. END OF RECEPTION (F40)

F40: END-OF-RECEPTION PROCESSING

This function contains the procedures for the end-of-reception processing of the program. It is executed as long as no errors have been found.

Within this function, there are four sub-functions which correspond to four automatically generated procedures that are conditioned by the value of the Operation Code (OPER).

SET-UP KEYS FOR NEW DISPLAY (F4010)

This function is executed for a 'display' or an 'update' operation. The keys to the segments with no preceding segment, or those used in display, are given a value here.

Depending on the categories defined on the screen, the memorization of the access key for the display segment is found in:

- . F40A for the screen-top category.
- . F40R for the repetitive category.
- . F40Z for the screen-bottom category.

SET-UP KEYS FOR SCREEN PAGING (F4020)

This function is executed for a 'screen continuation' operation. It contains the memorization of the first key for the display of the screen continuation, if the segment is used in the repetitive category.

END OF TRANSACTION (F4030)

This function is executed for an end-of-transaction operation.

TRANSFER TO ANOTHER SCREEN (F4040)

This function is executed for a transfer to another screen operation.

GENERATED PROGRAM: PROCEDURE DIVISION
 END OF RECEPTION (F40)

PAGE

100

4
 10

```

F40.          IF SCR-ER > '1' MOVE 'A' TO OPER GO TO F40-FN.          DO0030
F40-A.        IF OPERD NOT = SPACE MOVE OPERD TO OPER.              DO0030
*             *****                                              DO0030
*             *                                                     * DO0030
*             *   SET-UP KEYS FOR NEW DISPLAY                       * DO0030
*             *                                                     * DO0030
*             *****                                              DO0030
F4010.        IF OPER NOT = 'A' AND NOT = 'M' GO TO F4010-FN.        DO0030
F40A.
MOVE          SPACES          TO          CD00-KEYCD                DO0030
MOVE          'B'             TO          CD00-COCARA              DO0030
MOVE          CA00-NUCOM      TO          CD00-NUCOM              DO0030
MOVE          CD00-KEYCD     TO          K-ACD05-KEYCD.           DO0030
F40A-FN.      EXIT.                                                DO0030
F40R.
MOVE          J-0030-LINE    (1) TO          DO0030
                I-0030-LINE.                                DO0030
MOVE          SPACES          TO          CD00-KEYCD              DO0030
MOVE          'C'             TO          CD00-COCARA              DO0030
MOVE          CA00-NUCOM      TO          CD00-NUCOM              DO0030
MOVE          CD00-KEYCD     TO          K-RCD10-KEYCD (1).       DO0030
F40R-FN.      EXIT.                                                DO0030
F40Z.
MOVE          CA00-CLEME      TO          ME00-CLEME              DO0030
MOVE          ME00-CLEME      TO          K-ZME00-CLEME.          DO0030
F40Z-FN.      EXIT.                                                DO0030
F4010-FN.     EXIT.                                                DO0030
*             *****                                              DO0030
*             *                                                     * DO0030
*             *   SET-UP KEYS FOR SCREEN PAGING                     * DO0030
*             *                                                     * DO0030
*             *****                                              DO0030
F4020.        IF OPER NOT = 'S' GO TO F4020-FN.                      DO0030
MOVE          K-RCD10-KEYCD  (2) TO          DO0030
                K-RCD10-KEYCD (1).                            DO0030
F4020-FN.     EXIT.                                                DO0030
*             +-----+                                             P000
*             | I END OF TRANSACTION                                | I P000
*             +-----+                                             P000
F4029.        IF OPER = 'E'                                         P000
                NEXT SENTENCE ELSE GO TO          F4029-FN.       P000
MOVE          '** END OF TRANSACTION **' TO          P100
                END-MESSAGE.                                     P110
F4029-FN.     EXIT.                                                P000
*             *****                                              DO0030
*             *                                                     * DO0030
*             *   END OF TRANSACTION                                * DO0030
*             *                                                     * DO0030
*             *****                                              DO0030
F4030.        IF OPER NOT = 'E' GO TO F4030-FN.                      DO0030
PERFORM F80-HELP-D THRU F80-FN.                                    DO0030
PERFORM F81FI THRU F81FI-FN.                                      DO0030
CALL 'D$CLCONV' USING STATUS-WORD.                               DO0030
IF STATUS-FATAL GO TO F81ER.                                     DO0030
CALL 'D$ENDMSG' USING STATUS-WORD, END-MESSAGE.                 DO0030
IF STATUS-FATAL GO TO F81ER.                                     DO0030
CALL 'D$CLOSE' USING STATUS-WORD.                               DO0030
IF STATUS-FATAL GO TO F81ER.                                     DO0030
STOP RUN.                                                       DO0030
F4030-FN.     EXIT.                                                DO0030
*             *****                                              DO0030
*             *                                                     * DO0030
*             *   TRANSFER TO ANOTHER SCREEN                       * DO0030
*             *                                                     * DO0030
*             *****                                              DO0030
F4040.        IF OPER NOT = 'O' GO TO F4040-FN.                      DO0030
PERFORM F81FI THRU F81FI-FN.                                    DO0030
CALL 'D$PUTSCR' USING STATUS-WORD, COMMON-AREA.                 DO0030
IF STATUS-FATAL GO TO F81ER.                                     DO0030
CALL 'D$CLCONV' USING STATUS-WORD.                               DO0030
IF STATUS-FATAL GO TO F81ER.                                     DO0030
MOVE 5-0030-PROGE TO 5-0030-TRX                                  DO0030
CALL 'D$PASSOFF' USING STATUS-WORD, 5-0030-TRX.                 DO0030
IF STATUS-FATAL GO TO F81ER.                                     DO0030
CALL 'D$CLOSE' USING STATUS-WORD.                               DO0030
IF STATUS-FATAL GO TO F81ER.                                     DO0030
STOP RUN.                                                       DO0030

```

GENERATED PROGRAM: PROCEDURE DIVISION
END OF RECEPTION (F40)

PAGE

101

4
10

F4040-FN. EXIT.
F40-FN. EXIT.
END-OF-RECEPTION. EXIT.

DO0030
DO0030
DO0030

4.11. DISPLAY PREPARATION (F50)

F50: DISPLAY PREPARATION

The DISPLAY PREPARATION (F50) function contains the conditions for the set of procedures used in the 'DISPLAY' part of the program, F50 to F78-FN (END-OF-DISPLAY).

Sub-function F5010 is always generated. It ensures the initialization of work areas, and of the display screen description.

GENERATED PROGRAM: PROCEDURE DIVISION
DISPLAY PREPARATION (F50)

PAGE

103

4

11

```
*          *****  
*          *                               *          DO0030  
*          * DISPLAY PREPARATION         *          DO0030  
*          *                               *          DO0030  
*          *****  
F50.      IF OCF = '0' GO TO END-OF-DISPLAY. DO0030  
F5010.    DO0030  
          MOVE ZERO TO CATX. DO0030  
          MOVE ZERO TO CONFIGURATIONS. DO0030  
          MOVE ALL '1' TO FIRST-ON-SEGMENT. DO0030  
          IF SCR-ER > '1' GO TO F6999-ITER-FT. DO0030  
          MOVE SPACE TO O-0030. DO0030  
          PERFORM F8115 THRU F8115-FN. DO0030  
          MOVE K-R0030-LINE (1) TO DO0030  
          K-R0030-LINE (2). DO0030  
F5010-FN. EXIT. DO0030  
F50-FN.   EXIT. DO0030
```

4.12. CATEGORY PROCESSING LOOP (F55)

F55: CATEGORY PROCESSING LOOP

The CATEGORY PROCESSING LOOP (F55) function positions the category to be processed in 'DISPLAY' based on the CATX indicator, which can have the following values:

- . '0' Beginning of display.
- . ' ' Screen-top category.
- . 'R' Repetitive category.
- . 'Z' Screen-bottom category.

The procedures are generated based on the categories defined on the Call of Elements ('-CE') screen.

If no category is defined, the screen is considered a screen-top category.

For the repetitive category this function includes:

- . The interaction between the line of the category to be processed, and the output screen description field used to access each of the data elements of the line,
- . The initialization and incrementation of the ICATR indicator which manages the repetitive category.

4.13. SEGMENT ACCESS FOR DISPLAY (F60)

F60: SEGMENT ACCESS FOR DISPLAY

The SEGMENT ACCESS FOR DISPLAY (F60) function is generated when there is a segment to be accessed for display.

Depending on which categories defined on the screen contain a segment to be accessed for display, the F60 function includes the following:

- . F60A for the screen-top category,
- . F60R for the repetitive category,
- . F60Z for the screen-bottom category.

To process each category, there is one sub-function per access to a segment, including:

- . Loading of the key from the 'K-cddss-KEY' field stored in function F40. For the first display (OCF = '1'), the user must ensure that the 'K-' field is loaded.
- . Access by a PERFORM to the appropriate F80 sub-function depending on the category:
 - Direct read (F80-ddss-R),
 - Sequential Read after positioning (repetitive) (F80-ddss-P and F80-ddss-RN) based on the use of the segment (indicated on the '-CS').
- . The positioning of the Segment 'ddss-CF' variable.
- . Error processing, if necessary.

If a segment has a preceding segment, its Read will always be a Direct Read, even in the Repetitive category.

NOTE: Sub-functions are numbered based on the number of segments, their positions on the '-CS' screen, etc. As a result, a direct reference should never be made to a generated label in the specific procedures.

Use the Relative Positioning types '*A', '*P' and '*R' (see chapter "USE OF STRUCTURED CODE" in the ON-LINE SYSTEMS DEVELOPMENT Reference Manual.)

4.14. DATA ELEMENT TRANSFER (F65)

F65: DATA ELEMENT TRANSFER

The DATA ELEMENT TRANSFER (F65) function ensures the transfer of the segment data elements to the corresponding data elements on the screen.

Depending on which categories defined on the screen contain at least one transfer of a data element for display, the F65 function includes:

- . F65A for the screen-top category,
- . F65R for the repetitive category,
- . F65Z for the screen-bottom category.

If the data element is filled from a segment, the transfer is conditioned by the segment configuration variable (ddss-CF=1).

Paragraph 'F6999-ITER-FI' contains the return to the beginning of the display iteration.

GENERATED PROGRAM: PROCEDURE DIVISION
 DATA ELEMENT TRANSFER (F65)

PAGE

109

4

14

```

*          *****
*          *
*          * DATA ELEMENT TRANSFER
*          *
*          *****
F65.      EXIT.
F65A.    IF CATX NOT = ' ' GO TO F65A-FN.
        MOVE   PROGE          TO
            O-0030-PROGE.
        MOVE   SESSI          TO
            O-0030-SESSI.
        MOVE   DAT8C          TO
            O-0030-DATEM.
        MOVE   TIMDAY         TO
            O-0030-HEURE.
F65A-A7. MOVE   CA00-NUCOM     TO
            O-0030-NUCOM.
F65A-A7-FN. EXIT.
F65A-A8. MOVE   CA00-RAISOC    TO
            O-0030-RAISOC.
F65A-A8-FN. EXIT.
F65A-CD05.
        IF     CD05-CF NOT = '1' GO TO F65A-CD05-FN.
        MOVE   CD05-MATE       TO
            O-0030-MATE.
F65A-B0. MOVE   CD05-RELEA     TO
            O-0030-RELEA.
F65A-B0-FN. EXIT.
F65A-B1. MOVE   CD05-VILLE     TO
            O-0030-VILLE.
F65A-B1-FN. EXIT.
F65A-B2. MOVE   CD05-COPOS     TO
            O-0030-COPOS.
F65A-B2-FN. EXIT.
F65A-B3. MOVE   CD05-REFCLI    TO
            O-0030-REFCLI.
F65A-B3-FN. EXIT.
F65A-B4. MOVE   CD05-DATE      TO
            O-0030-DATE.
F65A-B4-FN. EXIT.
F65A-B5. MOVE   CD05-CORRES    TO
            O-0030-CORRES.
F65A-B5-FN. EXIT.
F65A-B6. MOVE   CD05-REMIS     TO
            O-0030-REMIS.
F65A-B6-FN. EXIT.
F65A-CD05-FN. EXIT.
F65A-FN. EXIT.
F65R.   IF CATX NOT = 'R' OR FT = '1' GO TO F65R-FN.
        IF ICATR > IRR GO TO F65R-FN.
F65R-A4. MOVE   CD00-FOURNI     TO
            O-0030-FOURNI.
F65R-A4-FN. EXIT.
F65R-CD10.
        IF     CD10-CF NOT = '1' GO TO F65R-CD10-FN.
        MOVE   CD10-QTMAC      TO
            O-0030-QTMAC.
F65R-A6. MOVE   CD10-QTMAL     TO
            O-0030-QTMAL.
F65R-A6-FN. EXIT.
F65R-A7. MOVE   CD10-INFOR     TO
            O-0030-INFOR.
F65R-A7-FN. EXIT.
F65R-CD10-FN. EXIT.
*          +-----+

```

GENERATED PROGRAM: PROCEDURE DIVISION
DATA ELEMENT TRANSFER (F65)

PAGE

110

4
14

```
* LEVEL 10      I REMAINS TO BE DELIVERED      I      P000
*              +-----+
F65BB.          IF      CD10-QTMAL NOT = ZERO      P000
                COMPUTE WW10-QTMAR =              P100
                    CD10-QTMAR - CD10-QTMAL      P110
                MOVE   WW10-QTMAR TO O-0030-QTMAR. P120
F65BB-FN.      EXIT.                               P000
F65R-FN.      EXIT.                               DO0030
F65Z.  IF CATX NOT = 'Z' GO TO F65Z-FN.           DO0030
F65Z-ME00.    IF      ME00-CF NOT = '1' GO TO F65Z-ME00-FN. DO0030
                MOVE   ME00-MESSA      TO
                    O-0030-MESSA.                DO0030
F65Z-ME00-FN. EXIT.                               DO0030
F65Z-FN.      EXIT.                               DO0030
F65-FN.      EXIT.                               DO0030
F6999-ITER-FI. GO TO F55.                        DO0030
F6999-ITER-FT. EXIT.                             DO0030
F6999-FN.      EXIT.                             DO0030
```

4.15. ERROR PROCESSING (F70)

F70: ERROR PROCESSING

The ERROR PROCESSING (F70) function is always generated.

Sub-function F7010 contains:

- . in F7010-A, testing of DE-ERR, positioning of the error attributes, access to the error message file, and coding of the error message on the screen.
- . in F7010-B, testing of T-XEMKY, access to the error message file, and coding of the error message on the screen.

Sub-function F7020 is generated if at least one variable field exists on the Screen Call of Elements (-CE).

This sub-function positions the attributes of the fields on the screen in display.

An 'invisible' field ('DARK' attribute) retains this attribute, even if it is erroneous (for ex., with passwords).

GENERATED PROGRAM: PROCEDURE DIVISION
 ERROR PROCESSING (F70)

4
 15

```

F70.          EXIT.                                DO0030
*             *****                                DO0030
*             *                                     *                                DO0030
*             *   ERROR PROCESSING                 *                                DO0030
*             *                                     *                                DO0030
*             *****                                DO0030
F7010.        MOVE ZERO TO K01 K02 K04 MOVE 1 TO K03. DO0030
              MOVE LIBRA TO EM00-LIBRA MOVE PROGR TO EM00-PROGR DO0030
              MOVE ZERO TO EM00-LINUM MOVE 'H' TO EM00-ENTYP. DO0030
F7010-A.      IF K02 = INR AND K03 < IRR MOVE INA TO K02 DO0030
              ADD 1 TO K03. ADD 1 TO K01 K02. DO0030
              IF DE-ER (K01) > '1' OR < '0' MOVE 'Y' TO DE-AT (4, K01) DO0030
              MOVE 'N' TO DE-AT (1, K01) DO0030
              MOVE 'N' TO DE-AT (2, K01) DO0030
              MOVE 'W' TO DE-AT (3, K01) DO0030
              IF K04 < IER MOVE DE-ER (K01) TO EM00-ERTYP DO0030
              MOVE K02 TO EM00-ERCOD9 MOVE EM00-XEMKY TO EM00-ERMSG DO0030
              PERFORM F80-EM00-R THRU F80-FN ADD 1 TO K04 DO0030
              MOVE EM00-ERMSG TO O-0030-ERMSG (K04). DO0030
              IF K01 < INT GO TO F7010-A. DO0030
              MOVE ZERO TO K50R. DO0030
F7010-B.      ADD 1 TO K50R IF K50R > K50L OR K04 NOT < IER GO TO DO0030
              F7010-FN. MOVE T-XEMKY (K50R) TO EM00-XEMKY EM00-ERMSG DO0030
              PERFORM F80-EM00-R THRU F80-FN. ADD 1 TO K04 DO0030
              MOVE EM00-ERMSG TO O-0030-ERMSG (K04) DO0030
              GO TO F7010-B. DO0030
F7010-FN.     EXIT.                                DO0030
*             *****                                DO0030
*             *                                     *                                DO0030
*             *   POSITIONING OF ATTRIBUTES         *                                DO0030
*             *                                     *                                DO0030
*             *****                                DO0030
F7020.        INSPECT DE-ATT1 (1) REPLACING ALL 'B' BY 'H' DO0030
              INSPECT DE-ATT1 (1) REPLACING ALL 'D' BY 'S'. DO0030
              INSPECT DE-ATT1 (2) REPLACING ALL SPACE BY LOW-VALUE. DO0030
              INSPECT DE-ATT1 (3) REPLACING ALL SPACE BY LOW-VALUE. DO0030
              INSPECT DE-ATT1 (3) REPLACING ALL 'B' BY 'E'. DO0030
              INSPECT DE-ATT1 (3) REPLACING ALL 'P' BY 'M'. DO0030
              INSPECT DE-ATT1 (3) REPLACING ALL 'T' BY 'C'. DO0030
              MOVE ZERO TO TALLI INSPECT DE-ATT1 (4) DO0030
              TALLYING TALLI FOR CHARACTERS BEFORE 'Y'. DO0030
              IF TALLI NOT < 0046 DO0030
              MOVE ZERO TO TALLI INSPECT DE-ATT1 (4) DO0030
              TALLYING TALLI FOR CHARACTERS BEFORE 'Z'. DO0030
              IF TALLI NOT < 0046 DO0030
              MOVE ZERO TO TALLI INSPECT DE-ATT1 (4) DO0030
              TALLYING TALLI FOR CHARACTERS BEFORE 'X'. DO0030
              IF TALLI NOT < 0046 DO0030
              MOVE ZERO TO TALLI. DO0030
              MOVE SPACE TO DE-ATT1 (4) ADD 1 TO TALLI DO0030
              MOVE 'Y' TO DE-AT (4, TALLI). DO0030
F7020-A.      IF A-0030-CHOIX (1 ) NOT = SPACE DO0030
              MOVE A-0030-CHOIX (1 ) TO DO0030
                  S-CHOIX-INT. DO0030
              IF A-0030-CHOIX (2 ) = 'U' DO0030
              MOVE A-0030-CHOIX (2 ) TO DO0030
                  S-CHOIX-EMPH ELSE DO0030
              MOVE A-0030-CHOIX (2 ) TO DO0030
                  S-CHOIX-HIGH. DO0030
              MOVE A-0030-CHOIX (3 ) TO DO0030
                  S-CHOIX-FORE. DO0030
              IF A-0030-CHOIX (4 ) = 'Y' DO0030
              MOVE S-CHOIX-XCO TO ERROR-X DO0030
              MOVE S-CHOIX-YCO TO ERROR-Y. DO0030
              IF A-0030-MATE (1 ) NOT = SPACE DO0030
              MOVE A-0030-MATE (1 ) TO DO0030
                  S-MATE-INT. DO0030
              IF A-0030-MATE (2 ) = 'U' DO0030
              MOVE A-0030-MATE (2 ) TO DO0030
                  S-MATE-EMPH ELSE DO0030
              MOVE A-0030-MATE (2 ) TO DO0030
                  S-MATE-HIGH. DO0030
              MOVE A-0030-MATE (3 ) TO DO0030
                  S-MATE-FORE. DO0030
  
```


GENERATED PROGRAM: PROCEDURE DIVISION
ERROR PROCESSING (F70)

PAGE

113

4

15

```
IF A-0030-MATE (4) = 'Y' DO0030
MOVE S-MATE-XCO TO ERROR-X DO0030
MOVE S-MATE-YCO TO ERROR-Y. DO0030
IF A-0030-RELEA (1) NOT = SPACE DO0030
MOVE A-0030-RELEA (1) TO DO0030
S-RELEA-INT. DO0030
IF A-0030-RELEA (2) = 'U' DO0030
MOVE A-0030-RELEA (2) TO DO0030
S-RELEA-EMPH ELSE DO0030
MOVE A-0030-RELEA (2) TO DO0030
S-RELEA-HIGH. DO0030
MOVE A-0030-RELEA (3) TO DO0030
S-RELEA-FORE. DO0030
IF A-0030-RELEA (4) = 'Y' DO0030
MOVE S-RELEA-XCO TO ERROR-X DO0030
MOVE S-RELEA-YCO TO ERROR-Y. DO0030
IF A-0030-RUE (1) NOT = SPACE DO0030
MOVE A-0030-RUE (1) TO DO0030
S-RUE-INT. DO0030
IF A-0030-RUE (2) = 'U' DO0030
MOVE A-0030-RUE (2) TO DO0030
S-RUE-EMPH ELSE DO0030
MOVE A-0030-RUE (2) TO DO0030
S-RUE-HIGH. DO0030
MOVE A-0030-RUE (3) TO DO0030
S-RUE-FORE. DO0030
IF A-0030-RUE (4) = 'Y' DO0030
MOVE S-RUE-XCO TO ERROR-X DO0030
MOVE S-RUE-YCO TO ERROR-Y. DO0030
IF A-0030-COPOS (1) NOT = SPACE DO0030
MOVE A-0030-COPOS (1) TO DO0030
S-COPOS-INT. DO0030
IF A-0030-COPOS (2) = 'U' DO0030
MOVE A-0030-COPOS (2) TO DO0030
S-COPOS-EMPH ELSE DO0030
MOVE A-0030-COPOS (2) TO DO0030
S-COPOS-HIGH. DO0030
MOVE A-0030-COPOS (3) TO DO0030
S-COPOS-FORE. DO0030
IF A-0030-COPOS (4) = 'Y' DO0030
MOVE S-COPOS-XCO TO ERROR-X DO0030
MOVE S-COPOS-YCO TO ERROR-Y. DO0030
IF A-0030-REFCLI (1) NOT = SPACE DO0030
MOVE A-0030-REFCLI (1) TO DO0030
S-REFCLI-INT. DO0030
IF A-0030-REFCLI (2) = 'U' DO0030
MOVE A-0030-REFCLI (2) TO DO0030
S-REFCLI-EMPH ELSE DO0030
MOVE A-0030-REFCLI (2) TO DO0030
S-REFCLI-HIGH. DO0030
MOVE A-0030-REFCLI (3) TO DO0030
S-REFCLI-FORE. DO0030
IF A-0030-REFCLI (4) = 'Y' DO0030
MOVE S-REFCLI-XCO TO ERROR-X DO0030
MOVE S-REFCLI-YCO TO ERROR-Y. DO0030
IF A-0030-DATE (1) NOT = SPACE DO0030
MOVE A-0030-DATE (1) TO DO0030
S-DATE-INT. DO0030
IF A-0030-DATE (2) = 'U' DO0030
MOVE A-0030-DATE (2) TO DO0030
S-DATE-EMPH ELSE DO0030
MOVE A-0030-DATE (2) TO DO0030
S-DATE-HIGH. DO0030
MOVE A-0030-DATE (3) TO DO0030
S-DATE-FORE. DO0030
IF A-0030-DATE (4) = 'Y' DO0030
MOVE S-DATE-XCO TO ERROR-X DO0030
MOVE S-DATE-YCO TO ERROR-Y. DO0030
IF A-0030-CORRES (1) NOT = SPACE DO0030
MOVE A-0030-CORRES (1) TO DO0030
S-CORRES-INT. DO0030
IF A-0030-CORRES (2) = 'U' DO0030
MOVE A-0030-CORRES (2) TO DO0030
S-CORRES-EMPH ELSE DO0030
MOVE A-0030-CORRES (2) TO DO0030
S-CORRES-HIGH. DO0030
MOVE A-0030-CORRES (3) TO DO0030
```

GENERATED PROGRAM: PROCEDURE DIVISION
ERROR PROCESSING (F70)

PAGE

114

4

15

```
                S-CORRES-FORE.                                DO0030
IF      A-0030-CORRES ( 4 ) = 'Y'                            DO0030
MOVE S-CORRES-XCO TO ERROR-X                                DO0030
MOVE S-CORRES-YCO TO ERROR-Y.                              DO0030
IF      A-0030-REMIS ( 1 ) NOT = SPACE                      DO0030
MOVE A-0030-REMIS ( 1 ) TO                                  DO0030
        S-REMIS-INT.                                        DO0030
IF      A-0030-REMIS ( 2 ) = 'U'                            DO0030
MOVE A-0030-REMIS ( 2 ) TO                                  DO0030
        S-REMIS-EMPH ELSE                                  DO0030
MOVE A-0030-REMIS ( 2 ) TO                                  DO0030
        S-REMIS-HIGH.                                      DO0030
MOVE A-0030-REMIS ( 3 ) TO                                  DO0030
        S-REMIS-FORE.                                      DO0030
IF      A-0030-REMIS ( 4 ) = 'Y'                            DO0030
MOVE S-REMIS-XCO TO ERROR-X                                DO0030
MOVE S-REMIS-YCO TO ERROR-Y.                              DO0030
MOVE ZERO TO ICATR.                                        DO0030
F7020-R. ADD 1 TO ICATR.                                     DO0030
MOVE P-0030-LINE (ICATR) TO                                  DO0030
        O-0030-LINE                                        DO0030
MOVE J-LINE (ICATR) TO                                     DO0030
        I-LINE                                            DO0030
MOVE B-0030-LINE (1, ICATR) TO                              DO0030
        A-0030-LINE (1)                                    DO0030
MOVE B-0030-LINE (4, ICATR) TO                              DO0030
        A-0030-LINE (4)                                    DO0030
MOVE B-0030-LINE (2, ICATR) TO                              DO0030
        A-0030-LINE (2)                                    DO0030
MOVE B-0030-LINE (3, ICATR) TO                              DO0030
        A-0030-LINE (3)                                    DO0030
IF      A-0030-CODMVT ( 1 ) NOT = SPACE                      DO0030
MOVE A-0030-CODMVT ( 1 ) TO                                  DO0030
        S-CODMVT-INT.                                      DO0030
IF      A-0030-CODMVT ( 2 ) = 'U'                            DO0030
MOVE A-0030-CODMVT ( 2 ) TO                                  DO0030
        S-CODMVT-EMPH ELSE                                  DO0030
MOVE A-0030-CODMVT ( 2 ) TO                                  DO0030
        S-CODMVT-HIGH.                                      DO0030
MOVE A-0030-CODMVT ( 3 ) TO                                  DO0030
        S-CODMVT-FORE.                                      DO0030
IF      A-0030-CODMVT ( 4 ) = 'Y'                            DO0030
MOVE S-CODMVT-XCO TO ERROR-X                                DO0030
MOVE S-CODMVT-YCO TO ERROR-Y.                              DO0030
IF      A-0030-FOURNI ( 1 ) NOT = SPACE                      DO0030
MOVE A-0030-FOURNI ( 1 ) TO                                  DO0030
        S-FOURNI-INT.                                      DO0030
IF      A-0030-FOURNI ( 2 ) = 'U'                            DO0030
MOVE A-0030-FOURNI ( 2 ) TO                                  DO0030
        S-FOURNI-EMPH ELSE                                  DO0030
MOVE A-0030-FOURNI ( 2 ) TO                                  DO0030
        S-FOURNI-HIGH.                                      DO0030
MOVE A-0030-FOURNI ( 3 ) TO                                  DO0030
        S-FOURNI-FORE.                                      DO0030
IF      A-0030-FOURNI ( 4 ) = 'Y'                            DO0030
MOVE S-FOURNI-XCO TO ERROR-X                                DO0030
MOVE S-FOURNI-YCO TO ERROR-Y.                              DO0030
IF      A-0030-QTMAC ( 1 ) NOT = SPACE                      DO0030
MOVE A-0030-QTMAC ( 1 ) TO                                  DO0030
        S-QTMAC-INT.                                      DO0030
IF      A-0030-QTMAC ( 2 ) = 'U'                            DO0030
MOVE A-0030-QTMAC ( 2 ) TO                                  DO0030
        S-QTMAC-EMPH ELSE                                  DO0030
MOVE A-0030-QTMAC ( 2 ) TO                                  DO0030
        S-QTMAC-HIGH.                                      DO0030
MOVE A-0030-QTMAC ( 3 ) TO                                  DO0030
        S-QTMAC-FORE.                                      DO0030
IF      A-0030-QTMAC ( 4 ) = 'Y'                            DO0030
MOVE S-QTMAC-XCO TO ERROR-X                                DO0030
MOVE S-QTMAC-YCO TO ERROR-Y.                              DO0030
IF      A-0030-INFOR ( 1 ) NOT = SPACE                      DO0030
MOVE A-0030-INFOR ( 1 ) TO                                  DO0030
        S-INFOR-INT.                                      DO0030
IF      A-0030-INFOR ( 2 ) = 'U'                            DO0030
MOVE A-0030-INFOR ( 2 ) TO                                  DO0030
        S-INFOR-EMPH ELSE                                  DO0030
MOVE A-0030-INFOR ( 2 ) TO                                  DO0030
```

GENERATED PROGRAM: PROCEDURE DIVISION
ERROR PROCESSING (F70)

PAGE

115

4

15

S-INFOR-HIGH.	DO0030
MOVE A-0030-INFOR (3) TO	DO0030
S-INFOR-FORE.	DO0030
IF A-0030-INFOR (4) = 'Y'	DO0030
MOVE S-INFOR-XCO TO ERROR-X	DO0030
MOVE S-INFOR-YCO TO ERROR-Y.	DO0030
MOVE O-0030-LINE TO	DO0030
P-0030-LINE (ICATR)	DO0030
MOVE I-LINE TO	DO0030
J-LINE (ICATR)	DO0030
IF ICATR < IRR GO TO F7020-R.	DO0030
F7020-Z.	DO0030
IF A-0030-EDIT (1) NOT = SPACE	DO0030
MOVE A-0030-EDIT (1) TO	DO0030
S-EDIT-INT.	DO0030
IF A-0030-EDIT (2) = 'U'	DO0030
MOVE A-0030-EDIT (2) TO	DO0030
S-EDIT-EMPH ELSE	DO0030
MOVE A-0030-EDIT (2) TO	DO0030
S-EDIT-HIGH.	DO0030
MOVE A-0030-EDIT (3) TO	DO0030
S-EDIT-FORE.	DO0030
IF A-0030-EDIT (4) = 'Y'	DO0030
MOVE S-EDIT-XCO TO ERROR-X	DO0030
MOVE S-EDIT-YCO TO ERROR-Y.	DO0030
F7020-FN. EXIT.	DO0030
F70-FN. EXIT.	DO0030
END-OF-DISPLAY. EXIT.	DO0030

4.16. DISPLAY AND END OF PROGRAM (F8Z)

F8Z: DISPLAY AND END OF PROGRAM

The DISPLAY AND END-OF-PROGRAM (F8Z) function is always generated.

Sub-function F8Z05 is generated if a call for help documentation is entered on the Screen Definition screen. It ensures the memorization of screen fields in the 'HE' file.

Sub-function F8Z10 contains the operation which sends the screen in three steps:

.Fixed content and display fields,

.Variable fields,

.Screen end.

If it is an initial display, it executes a PERFORM of F7020 (Positioning of attributes) in order to take the cursor position into account (in relation to F0110).

Sub-function F8Z20 contains the end-of-program operations.

GENERATED PROGRAM: PROCEDURE DIVISION
DISPLAY AND END OF PROGRAM (F8Z)

PAGE

117

4

16

```
F8Z.          EXIT.          DO0030
F8Z05.  IF SCR-ER = '1'      DO0030
      NEXT SENTENCE ELSE GO TO F8Z05-FN.      DO0030
      IF K-S0030-DOC NOT = '1'      GO TO F8Z05-A.      DO0030
      MOVE K-S0030-ERCOD9 TO K01 K02.      DO0030
      IF K02 > INR      DO0030
      COMPUTE K02 = K01 + (INR - INA) * (IRR - 1).      DO0030
      IF K02 < 1 OR K02 > INT MOVE 1 TO K02.      DO0030
      MOVE 'X' TO DE-AT (4, K02)      DO0030
      PERFORM F7020 THRU F7020-FN.      DO0030
F8Z05-A.
      IF K-S0030-DOC = '1'      DO0030
      PERFORM F80-HELP-R THRU F80-FN      DO0030
      MOVE '0' TO K-S0030-DOC      GO TO F8Z05-FN.      DO0030
      IF K-S0030-DOC NOT = ZERO      GO TO F8Z05-FN.      DO0030
      PERFORM F80-HELP-R THRU F80-FN.      DO0030
      IF IK = '1'      DO0030
      PERFORM F80-HELP-W THRU F80-FN ELSE      DO0030
      PERFORM F80-HELP-RW THRU F80-FN.      DO0030
F8Z05-FN.  EXIT.          DO0030
*          *****      DO0030
*          *          *      DO0030
*          * DISPLAY          *      DO0030
*          *          *      DO0030
*          *****      DO0030
F8Z10.
      IF SCR-ER NOT > '1'      DO0030
      AND DE-AT (4, 010) = 'X'      DO0030
      PERFORM F7020 THRU F7020-FN.      DO0030
      MOVE PROGR TO K-S0030-PROGR      DO0030
      CALL 'D$SETCV' USING STATUS-WORD.      DO0030
      IF STATUS-FATAL      GO TO F81ER.      DO0030
      MOVE PROGR TO K-S0030-PROGR      DO0030
      CALL 'D$PUTSCR' USING STATUS-WORD, COMMON-AREA.      DO0030
      IF STATUS-FATAL      GO TO F81ER.      DO0030
      MOVE ZERO TO S130-OUT-FID      DO0030
      MOVE ERROR-X TO S130-OUT-XCO      DO0030
      MOVE ERROR-Y TO S130-OUT-YCO.      DO0030
      MOVE O-0030 TO SCREEN-DO0030-130-DATA      DO0030
      CALL 'D$SEND' USING STATUS-WORD      DO0030
      SCREEN-DO0030-130.      DO0030
      IF STATUS-FATAL      GO TO F81ER.      DO0030
F8Z10-FN.  EXIT.          DO0030
*          *****      DO0030
*          *          *      DO0030
*          * END OF PROGRAM      *      DO0030
*          *          *      DO0030
*          *****      DO0030
F8Z20.
      PERFORM F81FI THRU F81FI-FN.      DO0030
      CALL 'D$CLOSE' USING STATUS-WORD.      DO0030
      IF STATUS-FATAL      GO TO F81ER.      DO0030
      STOP RUN.      DO0030
F8Z20-FN.  EXIT.          DO0030
F8Z-FN.    EXIT.          DO0030
```

4.17. PHYSICAL SEGMENT ACCESS ROUTINES (F80)

F80 : PHYSICAL SEGMENT ACCESS ROUTINES

This function must contain the physical accesses to the segments.

In this function, the user has to code the physical accesses to the segments or databases. In order to conform to the logical accesses (F25, F35, F60), the coding of the access sub-functions must be done as illustrated in the following example. The segment code in the program in this example is CD10.

- . F80-CD10-R Direct read.
- . F80-CD10-RU Direct read with update.
- . F80-CD10-P Positioning of a sequential read.
- . F80-CD10-RN Sequential read.
- . F80-CD10-W Write.
- . F80-CD10-RW Rewrite.
- . F80-CD10-D Deletion.
- . F80-CD10-UN Unlock of record.

If a call for HELP documentation has been entered on the Screen Definition screen, the labels of the following sub-functions are generated; however the user has to code these sub-functions manually.

- . F80-HELP-W Write.
- . F80-HELP-RW Rewrite.
- . F80-HELP-R Direct read.
- . F80-HELP-D Deletion.

If the access methods are programmed by the user, refer to Chapter "USE OF STRUCTURED CODE" in the ON-LINE SYSTEMS DEVELOPMENT Reference Manual.

GENERATED PROGRAM: PROCEDURE DIVISION
 PHYSICAL SEGMENT ACCESS ROUTINES (F80)

PAGE

119

4
17

```

*          *****
*          *
*          *   PHYSICAL SEGMENT ACCESS ROUTINES *
*          *
*          *****
F80.
*          +-----+
* LEVEL 10  I SEARCH RECORD CD05          I
*          +-----+
F80BB.     EXIT.
F80-CD05-R. EXIT.
F80-CD05-RU.
  MOVE      'F80BB' TO 7-WW00-FONCT
  MOVE      'FETCH' TO 7-WW00-ORDRE
  MOVE      CD05 TO DC05
  FETCH DC05 RECORD
  MOVE      DC05 TO CD05.
  PERFORM   F98ER THRU F98ER-FN.
  IF       IK = ZERO
  GO TO F80-OK.
  GO TO F80-KO.
F80BB-FN.  EXIT.
*          +-----+
* LEVEL 10  I UPDATE RECORD CD05          I
*          +-----+
F80BC.     EXIT.
F80-CD05-RW.
  MOVE      'F80CD05' TO 7-WW00-FONCT
  MOVE      'MODIFY' TO 7-WW00-ORDRE
  MOVE      CD05 TO DC05
  FIND      DC05 RECORD.
  PERFORM   F98ER THRU F98ER-FN.
  IF       IK NOT = ZERO
  GO TO F80-KO
  MODIFY    DC05 RECORD
  PERFORM   F98ER THRU F98ER-FN.
  IF       IK = ZERO
  GO TO F80-OK.
  GO TO F80-KO.
F80BC-FN.  EXIT.
*          +-----+
* LEVEL 10  I FETCH CD10 VIA ACCESS       I
*          +-----+
F80LI.     EXIT.
F80-CD10-P. EXIT.
F80-CD10-R. EXIT.
F80-CD10-RU.
  MOVE      'F80CD10' TO 7-WW00-FONCT
  MOVE      'FETCHVIA' TO 7-WW00-ORDRE
  MOVE      CD10 TO DC10
  FETCH DC10 VIA WW0510
  USING CD10-FOURNI
  MOVE      DC10 TO FOURNI.
  PERFORM   F98ER THRU F98ER-FN.
  IF       IK = ZERO
  GO TO F80-OK.
  GO TO F80-KO.
F80LI-FN.  EXIT.
*          +-----+
* LEVEL 10  I READ NEXT SEGMENT CD10      I
*          +-----+
F80LN.     EXIT.
F80-CD10-RN.
  MOVE      'F80CD10' TO 7-WW00-FONCT
  MOVE      'FETCH' TO 7-WW00-ORDRE
  MOVE      CD10 TO DC10
  FETCH NEXT DC10 WITHIN WWS0510 SET
  MOVE      DC10 TO CD10.
  PERFORM   F98ER THRU F98ER-FN.
  IF       IK = ZERO
  GO TO F80-OK.
  GO TO F80-KO.
F80LN-FN.  EXIT.
*          +-----+
* LEVEL 10  I CREATION RECORD CD10        I
*          +-----+
F80LM.     EXIT.

```


4.18. PERFORMED VALIDATION FUNCTIONS (F81)

F81: PERFORMED VALIDATION FUNCTIONS

The PERFORMED VALIDATIONS FUNCTIONS (F81) function is always generated.

Sub-function F81ER contains the abnormal end routine.

Sub-function F81ES contains routines specific to UNISYS.

Sub-function F81FI contains UNISYS routines to close the files. It is called at the end of a transaction and at the end of the program.

Sub-function F81UT contains the memorization of errors in the user's error 'stack'.

Sub-function F8110 is generated if the screen contains at least one numeric field.

It contains the procedures which format the field to be validated in the work area; the numeric class validation; and the possible positioning of error messages.

Sub-function F8115 insures the initialization of the output variable zones according to the initialization character entered on the Dialogue/Screen Definition screen and/or the initialization values given to Data Elements.

Sub-function F8120 is generated if at least one variable data element ('V') has a date format, or if a date processing operator is specified on Procedural Code (-P) lines of a program. (In this case, the F8120-ER and F8120-KO functions are not generated.) It contains the formatting and validation of a date.

Sub-function F8130 is generated if a HELP documentation call is entered on the Screen Definition line. It prepares the field to be backed-up.

Sub-function F8150 searches the first character of each authorized field in order to detect the two documentation request characters (documentation on the screen, or documentation on a data element.)

GENERATED PROGRAM: PROCEDURE DIVISION
 PERFORMED VALIDATION FUNCTIONS (F81)

PAGE

124

4

18

```

F81. DO0030
* -----+
* LEVEL 10 I DPS ERROR I P000
* -----+ P000
F81ER. P000
  PERFORM F81FI THRU F81FI-FN P100
  MOVE STATUS-FONCTION TO 7-WW00-FUNCT P110
  MOVE STATUS-CODE TO 7-WW00-ERCOD P120
  MOVE 7-WW00 TO END-MESSAGE. P140
* -----+ P000
* LEVEL 15 I ERROR DPS MANAGEMENT I P000
* -----+ P000
F81ES. P000
*-----> FUNCTION KEY MSG-WAIT P010
  IF STATUS-FUNCTION = 05 P100
  AND (STATUS-CODE = 31 OR 34) P110
  CALL 'D$RESET' USING STATUS-WORD. P100
  IF STATUS-FUNCTION = 06 P200
  AND (STATUS-CODE = 43 OR 44) P210
*-----> HELP FUNCTION NOT AVAILABLE P200
  MOVE 7-HELP-ERROR TO ERROR-MESSAGE P210
  CALL 'D$SENDERR' USING STATUS-WORD P240
  ERROR-MESSAGE ERROR-COORDINATES. P250
  IF STATUS-FATAL P300
  GO TO F81ES-FN. P300
  GO TO F8Z20. P400
F81ES-FN. EXIT. P000
* -----+ P000
* LEVEL 15 I DISPLAY DPS ERROR I P000
* -----+ P000
F81EV. P000
  DISPLAY '***** DPS ERROR *****' P100
  UPON PRINTER P110
  DISPLAY 'PROGRAM : ' PROGR P120
  UPON PRINTER P130
  DISPLAY 'FUNCT. PACBASE : ' P140
  7-WW00-FONCT P150
  UPON PRINTER P160
  DISPLAY 'DPS ORDER : ' P170
  7-WW00-ORDRE P180
  UPON PRINTER P190
  DISPLAY 'STATUS-FUNCTION : ' P200
  STATUS-FUNCTION P210
  UPON PRINTER P220
  DISPLAY 'STATUS-CODE : ' P230
  STATUS-CODE P240
  UPON PRINTER. P250
  IF IMPART-DEPART = '1' P300
  DEPART WITH ROLLBACK. P300
  CALL 'D$CLCONV' USING STATUS-WORD P310
  CALL 'D$ERRMSG' USING STATUS-WORD P320
  MOVE SPACE TO COMMON-AREA P340
  CALL 'D$PUTSCR' USING STATUS-WORD P360
  COMMON-AREA P370
  CALL 'D$TERM' USING STATUS-WORD. P400
F81EV-FN. EXIT. P000
F81ER-FN. EXIT. P000
* -----+ P000
* LEVEL 10 I CLOSE DATABASE I P000
* -----+ P000
F81FI. P000
  CLOSE ALL ON ERROR GO TO F99RB. P100
  DEPART ON ERROR GO TO F99RB. P200
F81FI-FN. EXIT. P000
* ***** DO0030
* ***** DO0030
* * MEMORIZATION OF USER'S ERRORS * DO0030
* * * * * DO0030
* ***** DO0030
F81UT. IF K50L < K50M ADD 1 TO K50L DO0030
  MOVE XEMKY TO T-XEMKY (K50L). MOVE 'E' TO CAT-ER. DO0030
F81UT-FN. EXIT. DO0030
* ***** DO0030
* * * * * DO0030
* * NUMERIC VALIDATION * DO0030
* * * * * DO0030
* ***** DO0030

```

GENERATED PROGRAM: PROCEDURE DIVISION
 PERFORMED VALIDATION FUNCTIONS (F81)

4

18

```

F8110.    MOVE ZERO TO TPOINT K01 K02 K03 ZONUM3 ZONUM2          DO0030
          C9 C91.                                               DO0030
F8110-1.  IF K01 > 26 OR K02 > 17 GO TO F8110-5.                DO0030
          ADD 1 TO K01.                                          DO0030
          IF C1 (K01) = SPACE OR C1 (K01) = '.' GO TO F8110-1.  DO0030
          IF C1 (K01) NOT = '-' AND C1 (K01) NOT = '+' GO TO F8110-2. DO0030
          IF C9 NOT = ZERO                                       DO0030
          MOVE '5' TO DEL-ER GO TO F8110-FN.                    DO0030
          IF K02 = ZERO MOVE '1' TO C91.                         DO0030
          IF C1 (K01) = '+' MOVE 1 TO C9 GO TO F8110-1.         DO0030
          IF SIGNE = ' ' MOVE '5' TO DEL-ER GO TO F8110-FN.    DO0030
          MOVE -1 TO C9 GO TO F8110-1.                           DO0030
F8110-2.  IF C1 (K01) NOT = ',' GO TO F8110-4.                  DO0030
          IF TPOINT = '1' OR NBCHP = 0                           DO0030
          MOVE '5' TO DEL-ER GO TO F8110-FN.                    DO0030
F8110-3.  IF K02 > NBCHA MOVE '5' TO DEL-ER GO TO F8110-FN.    DO0030
          COMPUTE K04 = 18 - NBCHA + K02 MOVE 1 TO C3 (K04)     DO0030
          DIVIDE ZONUM4 INTO ZONUM9 MOVE NBCHA TO K02           DO0030
          MOVE '1' TO TPOINT GO TO F8110-1.                     DO0030
F8110-4.  IF C1 (K01) NOT NUMERIC MOVE '4' TO DEL-ER           DO0030
          GO TO F8110-FN.                                         DO0030
          IF C9 NOT = ZERO AND C91 = ZERO                        DO0030
          MOVE '5' TO DEL-ER GO TO F8110-FN.                    DO0030
          IF C1 (K01) = '0' AND K02 = ZERO AND TPOINT = '0'    DO0030
          GO TO F8110-1. ADD 1 TO K02 MOVE C1 (K01) TO C2 (K02). DO0030
          IF TPOINT = '1' ADD 1 TO K03. IF K03 > NBCHP MOVE '5' DO0030
          TO DEL-ER GO TO F8110-FN. GO TO F8110-1.             DO0030
F8110-5.  IF TPOINT = '0' AND K02 > ZERO GO TO F8110-3.        DO0030
          IF SIGNE NOT = '+' GO TO F8110-FN.                    DO0030
          IF C9 = ZERO MOVE 1 TO C9.                              DO0030
          ADD NBCHA NBCHP GIVING K01 MULTIPLY C9 BY C29 (K01).  DO0030
          IF C29 (K01) = ZERO AND C9 = -1 MOVE C4 TO C2 (K01).  DO0030
F8110-FN.  EXIT.                                                DO0030
F8115.
          MOVE ALL '_'                                          DO0030
          TO O-0030-CHOIX.                                       DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-MATE.                                         DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-RELEA.                                       DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-RUE.                                          DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-COPOS.                                       DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-REFCLI.                                       DO0030
          MOVE '..___.'                                          DO0030
          TO O-0030-DATE.                                         DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-CORRES.                                       DO0030
          MOVE ALL '_'                                          DO0030
          TO F-0030-REMIS.                                       DO0030
          MOVE ZERO TO ICATR.                                     DO0030
F8115-GRP. ADD 1 TO ICATR                                        DO0030
          MOVE P-0030-LINE (ICATR) TO O-0030-LINE                DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-CODMVT.                                       DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-FOURNI.                                       DO0030
          MOVE ALL '_'                                          DO0030
          TO F-0030-QTMAC.                                       DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-INFOR.                                       DO0030
          MOVE O-0030-LINE TO P-0030-LINE (ICATR).              DO0030
          IF ICATR < IRR GO TO F8115-GRP.                        DO0030
          MOVE ALL '_'                                          DO0030
          TO O-0030-EDIT.                                         DO0030
F8115-FN.  EXIT.                                                DO0030
*          *****                                              DO0030
*          *                                                    DO0030
*          *  VALIDATION AND SETTING OF DATE  *                  DO0030
*          *                                                    DO0030
*          *****                                              DO0030
F8120.    EXIT.                                                 DO0030
F8120-C.  MOVE DAT73C TO DATCTY.                                  DO0030
          MOVE DAT71C TO DAT71.                                  DO0030
          MOVE DAT72C TO DAT72.                                  DO0030

```

GENERATED PROGRAM: PROCEDURE DIVISION
PERFORMED VALIDATION FUNCTIONS (F81)

PAGE

126

4

18

```
MOVE DAT74C TO DAT73. DO0030
MOVE '00111' TO TT-DAT GO TO F8120-T. DO0030
F8120-D. MOVE CENTUR TO DATCTY DAT73C. DO0030
MOVE DAT71 TO DAT71C. DO0030
MOVE DAT72 TO DAT72C. DO0030
MOVE DAT73 TO DAT74C. DO0030
MOVE '00111' TO TT-DAT GO TO F8120-T. DO0030
F8120-E. MOVE CENTUR TO DATCTY DAT83C. DO0030
MOVE DAT81 TO DAT81C. DO0030
MOVE DAT82 TO DAT82C. DO0030
MOVE DAT83 TO DAT84C MOVE DATSEP TO DAT8S1C DAT8S2C. DO0030
MOVE '01011' TO TT-DAT GO TO F8120-T. DO0030
F8120-G. MOVE DAT81G TO DATCTY. DO0030
MOVE DAT82G TO DAT61. DO0030
MOVE DAT83G TO DAT62. DO0030
MOVE DAT84G TO DAT63. DO0030
MOVE '10110' TO TT-DAT GO TO F8120-T. DO0030
F8120-I. MOVE CENTUR TO DATCTY DAT61C. DO0030
MOVE DAT61 TO DAT62C. DO0030
MOVE DAT62 TO DAT63C. DO0030
MOVE DAT63 TO DAT64C. DO0030
MOVE '10101' TO TT-DAT GO TO F8120-T. DO0030
F8120-M. MOVE DAT83C TO DATCTY. DO0030
MOVE DAT81C TO DAT81. DO0030
MOVE DAT82C TO DAT82. DO0030
MOVE DAT84C TO DAT83 MOVE DATSEP TO DAT8S1 DAT8S2. DO0030
MOVE '01011' TO TT-DAT GO TO F8120-T. DO0030
F8120-S. MOVE DAT61C TO DATCTY. DO0030
MOVE DAT62C TO DAT61. DO0030
MOVE DAT63C TO DAT62. DO0030
MOVE DAT64C TO DAT63. DO0030
MOVE '10101' TO TT-DAT. DO0030
F8120-T. IF T-DAT (1) = '1' DO0030
MOVE DAT61 TO DAT73 DAT74C DO0030
MOVE DAT62 TO DAT72 DAT72C DO0030
MOVE DAT63 TO DAT71 DAT71C DO0030
MOVE DATCTY TO DAT73C. DO0030
IF T-DAT (2) = '1' DO0030
MOVE DAT81 TO DAT71 DAT71C DO0030
MOVE DAT82 TO DAT72 DAT72C DO0030
MOVE DAT83 TO DAT73 DAT74C DO0030
MOVE DATCTY TO DAT73C. DO0030
IF T-DAT (3) = '1' DO0030
MOVE DAT71 TO DAT81 DAT81C DO0030
MOVE DAT72 TO DAT82 DAT82C DO0030
MOVE DAT73 TO DAT83 DAT84C DO0030
MOVE DATSEP TO DAT8S1 DAT8S2 DAT8S1C DAT8S2C DO0030
MOVE DATCTY TO DAT83C. DO0030
IF T-DAT (4) = '1' DO0030
MOVE DAT71 TO DAT63 DAT64C DO0030
MOVE DAT72 TO DAT62 DAT63C DO0030
MOVE DAT73 TO DAT61 DAT62C DO0030
MOVE DATCTY TO DAT61C. DO0030
IF T-DAT (5) = '1' DO0030
MOVE DAT61 TO DAT82G DO0030
MOVE DAT62 TO DAT83G DO0030
MOVE DAT63 TO DAT84G DO0030
MOVE DATSET TO DAT8S1G DAT8S2G DO0030
MOVE DATCTY TO DAT81G. DO0030
F8120-Z. EXIT. DO0030
F8120-ER. MOVE '1' TO DEL-ER. DO0030
IF DAT6 NOT NUMERIC GO TO F8120-KO. DO0030
IF DATCTY NOT NUMERIC GO TO F8120-KO. DO0030
IF DAT62 > '12' OR DAT62 = '00' OR DO0030
DAT63 > '31' OR DAT63 = '00' GO TO F8120-KO. DO0030
IF DAT63 > '30' AND DO0030
(DAT62 = '04' OR DAT62 = '06' OR DO0030
DAT62 = '09' OR DAT62 = '11') GO TO F8120-KO. DO0030
IF DAT62 NOT = '02' GO TO F8120-FN. DO0030
IF DAT63 > '29' GO TO F8120-KO. DO0030
IF DAT619 = ZERO DO0030
DIVIDE DATCTY9 BY 4 GIVING LEAP-REM DO0030
COMPUTE LEAP-REM = DATCTY9 - 4 * LEAP-REM DO0030
ELSE DIVIDE DAT619 BY 4 GIVING LEAP-REM DO0030
COMPUTE LEAP-REM = DAT619 - 4 * LEAP-REM. DO0030
IF DAT63 < '29' OR LEAP-REM = ZERO GO TO F8120-FN. DO0030
F8120-KO. MOVE '5' TO DEL-ER. DO0030
```

```

F8120-FN.      EXIT.                                DO0030
*              *****                                DO0030
*              *                                      *                                DO0030
*              *   HELP SUB-FUNCTION                 *                                DO0030
*              *                                      *                                DO0030
*              *****                                DO0030
F8130.
IF I-0030-CHOIX NOT = HIGH-VALUE                    DO0030
MOVE I-0030-CHOIX TO O-0030-CHOIX.                  DO0030
IF I-0030-MATE NOT = HIGH-VALUE                      DO0030
MOVE I-0030-MATE TO O-0030-MATE.                    DO0030
IF I-0030-RELEA NOT = HIGH-VALUE                    DO0030
MOVE I-0030-RELEA TO O-0030-RELEA.                 DO0030
IF I-0030-RUE NOT = HIGH-VALUE                      DO0030
MOVE I-0030-RUE TO O-0030-RUE.                     DO0030
IF I-0030-COPOS NOT = HIGH-VALUE                    DO0030
MOVE I-0030-COPOS TO O-0030-COPOS.                 DO0030
IF I-0030-REFCLI NOT = HIGH-VALUE                   DO0030
MOVE I-0030-REFCLI TO O-0030-REFCLI.               DO0030
IF I-0030-DATE NOT = HIGH-VALUE                     DO0030
MOVE I-0030-DATE TO O-0030-DATE.                   DO0030
IF I-0030-CORRES NOT = HIGH-VALUE                   DO0030
MOVE I-0030-CORRES TO O-0030-CORRES.               DO0030
IF E-0030-REMIS NOT = HIGH-VALUE                    DO0030
MOVE E-0030-REMIS TO F-0030-REMIS.                 DO0030
MOVE ZERO TO ICATR.                                 DO0030
F8130-GRP. ADD 1 TO ICATR                            DO0030
MOVE J-0030-LINE (ICATR) TO I-0030-LINE             DO0030
MOVE P-0030-LINE (ICATR) TO O-0030-LINE             DO0030
IF I-0030-CODMVT NOT = HIGH-VALUE                   DO0030
MOVE I-0030-CODMVT TO O-0030-CODMVT.               DO0030
IF I-0030-FOURNI NOT = HIGH-VALUE                   DO0030
MOVE I-0030-FOURNI TO O-0030-FOURNI.               DO0030
IF E-0030-QTMAC NOT = HIGH-VALUE                    DO0030
MOVE E-0030-QTMAC TO F-0030-QTMAC.                 DO0030
IF I-0030-INFOR NOT = HIGH-VALUE                    DO0030
MOVE I-0030-INFOR TO O-0030-INFOR.                 DO0030
MOVE O-0030-LINE TO P-0030-LINE (ICATR).           DO0030
IF ICATR < IRR GO TO F8130-GRP.                    DO0030
IF I-0030-EDIT NOT = HIGH-VALUE                     DO0030
MOVE I-0030-EDIT TO O-0030-EDIT.                   DO0030
F8130-FN.      EXIT.                                DO0030
*              *****                                DO0030
*              *                                      *                                DO0030
*              *   SEARCH FOR DOCUMENTATION REQUEST *                                DO0030
*              *                                      *                                DO0030
*              *****                                DO0030
F8150.
MOVE ZERO TO K-S0030-ERCOD.                          DO0030
IF I-0030-CHOIX = '$'                                DO0030
MOVE HIGH-VALUE TO I-0030-CHOIX                     DO0030
MOVE 001 TO K-S0030-ERCOD GO TO F8150-FN.           DO0030
IF I-0030-CHOIX = '!'                                DO0030
MOVE HIGH-VALUE TO I-0030-CHOIX                     DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN.         DO0030
IF I-0030-MATE = '$'                                  DO0030
MOVE HIGH-VALUE TO I-0030-MATE                       DO0030
MOVE 002 TO K-S0030-ERCOD GO TO F8150-FN.           DO0030
IF I-0030-MATE = '!'                                  DO0030
MOVE HIGH-VALUE TO I-0030-MATE                       DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN.         DO0030
IF I-0030-RELEA = '$'                                 DO0030
MOVE HIGH-VALUE TO I-0030-RELEA                     DO0030
MOVE 003 TO K-S0030-ERCOD GO TO F8150-FN.           DO0030
IF I-0030-RELEA = '!'                                  DO0030
MOVE HIGH-VALUE TO I-0030-RELEA                     DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN.         DO0030
IF I-0030-RUE = '$'                                   DO0030
MOVE HIGH-VALUE TO I-0030-RUE                       DO0030
MOVE 004 TO K-S0030-ERCOD GO TO F8150-FN.           DO0030
IF I-0030-RUE = '!'                                  DO0030
MOVE HIGH-VALUE TO I-0030-RUE                       DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN.         DO0030
IF I-0030-COPOS = '$'                                 DO0030
MOVE HIGH-VALUE TO I-0030-COPOS                     DO0030
MOVE 005 TO K-S0030-ERCOD GO TO F8150-FN.           DO0030
IF I-0030-COPOS = '!'                                  DO0030
    
```

GENERATED PROGRAM: PROCEDURE DIVISION
 PERFORMED VALIDATION FUNCTIONS (F81)

4

18

```

MOVE HIGH-VALUE TO I-0030-COPOS DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF I-0030-REFCLI = '$' DO0030
MOVE HIGH-VALUE TO I-0030-REFCLI DO0030
MOVE 006 TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF I-0030-REFCLI = '!' DO0030
MOVE HIGH-VALUE TO I-0030-REFCLI DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF I-0030-DATE = '$' DO0030
MOVE HIGH-VALUE TO I-0030-DATE DO0030
MOVE 007 TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF I-0030-DATE = '!' DO0030
MOVE HIGH-VALUE TO I-0030-DATE DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF I-0030-CORRES = '$' DO0030
MOVE HIGH-VALUE TO I-0030-CORRES DO0030
MOVE 008 TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF I-0030-CORRES = '!' DO0030
MOVE HIGH-VALUE TO I-0030-CORRES DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF E-0030-REMIS = '$' DO0030
MOVE HIGH-VALUE TO E-0030-REMIS DO0030
MOVE 009 TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF E-0030-REMIS = '!' DO0030
MOVE HIGH-VALUE TO E-0030-REMIS DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN. DO0030
MOVE ZERO TO ICATR. DO0030
F8150-GRP. ADD 1 TO ICATR DO0030
MOVE J-0030-LINE (ICATR) TO I-0030-LINE DO0030
IF I-0030-CODMVT = '$' DO0030
MOVE HIGH-VALUE TO I-0030-CODMVT DO0030
MOVE 010 TO K-S0030-ERCOD GO TO F8150-A. DO0030
IF I-0030-CODMVT = '!' DO0030
MOVE HIGH-VALUE TO I-0030-CODMVT DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-A. DO0030
IF I-0030-FOURNI = '$' DO0030
MOVE HIGH-VALUE TO I-0030-FOURNI DO0030
MOVE 011 TO K-S0030-ERCOD GO TO F8150-A. DO0030
IF I-0030-FOURNI = '!' DO0030
MOVE HIGH-VALUE TO I-0030-FOURNI DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-A. DO0030
IF E-0030-QTMAC = '$' DO0030
MOVE HIGH-VALUE TO E-0030-QTMAC DO0030
MOVE 012 TO K-S0030-ERCOD GO TO F8150-A. DO0030
IF E-0030-QTMAC = '!' DO0030
MOVE HIGH-VALUE TO E-0030-QTMAC DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-A. DO0030
IF I-0030-INFOR = '$' DO0030
MOVE HIGH-VALUE TO I-0030-INFOR DO0030
MOVE 013 TO K-S0030-ERCOD GO TO F8150-A. DO0030
IF I-0030-INFOR = '!' DO0030
MOVE HIGH-VALUE TO I-0030-INFOR DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-A. DO0030
MOVE I-0030-LINE TO J-0030-LINE (ICATR). DO0030
IF ICATR < IRR GO TO F8150-GRP. DO0030
IF I-0030-EDIT = '$' DO0030
MOVE HIGH-VALUE TO I-0030-EDIT DO0030
MOVE 014 TO K-S0030-ERCOD GO TO F8150-FN. DO0030
IF I-0030-EDIT = '!' DO0030
MOVE HIGH-VALUE TO I-0030-EDIT DO0030
MOVE SPACE TO K-S0030-ERCOD GO TO F8150-FN. DO0030
GO TO F8150-B. DO0030
F8150-A. MOVE I-0030-LINE TO J-0030-LINE (ICATR). DO0030
F8150-B. EXIT. DO0030
F8150-FN. EXIT. DO0030
F81-FN. EXIT. DO0030

```


4.19. CALLED USER FUNCTIONS

```

*          +-----+
* LEVEL 10  I ZIP CODE VALIDATION          I
*          +-----+
F93CP.
  MOVE 1 TO      IWP20R.
F93CP-100. IF    IWP20R NOT >  IWP20L
  AND          WP20-COPOS  (IWP20R)
  NOT =        WP30-COPOS
  ADD 1 TO     IWP20R      GO TO F93CP-100.
  IF          IWP20R > IWP20L
  MOVE        '5' TO DEL-ER
  GO TO F93CP-FN.
F93CP-FN.      EXIT.
*          +-----+
* LEVEL 10  I DMS ERROR                    I
*          +-----+
F98ER.
  IF          ERROR-STATUS = ZERO
  MOVE        ZERO TO IK
  GO TO F98ER-FN.
  IF          ERROR-CODE = '05'
  AND          ERROR-FUNCTION = '12'
  MOVE        '1' TO IK
  GO TO F98ER-FN.
  IF          ERROR-CODE = '06'
  AND          ERROR-FUNCTION = '03'
  MOVE        '2' TO IK
  GO TO F98ER-FN.
  IF          (ERROR-CODE = '07' OR '13')
  AND          ERROR-FUNCTION = '03'
  MOVE        '3' TO IK
  GO TO F98ER-FN.
  IF          ERROR-CODE = '15'
  AND          ERROR-FUNCTION = '02'
  MOVE        '4' TO IK
  GO TO F98ER-FN.
  MOVE        '5' TO IK.
F98ER-FN.      EXIT.
*          +-----+
* LEVEL 10  I ROLL-BACK ERROR              I
*          +-----+
F99RB.
  MOVE        RB-ERROR-CODE TO 7-WW00-RBCODE
  MOVE        ERROR-FUNCTION TO 7-WW00-FUNCT
  MOVE        ERROR-CODE TO 7-WW00-ERCOD
  MOVE        ERROR-NUM TO 7-WW00-NUM
  MOVE        7-WW00 TO ERROR-MESSAGE
*----->  --->  DISPLAY DMS ERROR  <---
  DISPLAY      '***** DMS ERROR *****'
  UPON PRINTER
  DISPLAY      'PROGRAM      : ' PROGE
  UPON PRINTER
  DISPLAY      'ERROR-STATUS : ' ERROR-STATUS
  UPON PRINTER
  DISPLAY      'ERROR-NUM    : ' ERROR-NUM
  UPON PRINTER
  DISPLAY      'ERROR-AREA   : ' ERROR-AREA
  UPON PRINTER
  DISPLAY      'ERROR-RECORD : ' ERROR-RECORD
  UPON PRINTER
  DISPLAY      'ERROR-SET    : ' ERROR-SET
  UPON PRINTER
  DISPLAY      'IK OPER CATX CATM ICATR ICF '
  'OCF I-PFKEY'
  UPON PRINTER
  DISPLAY      ' ' IK ' ' OPER ' ' CATX
  ' ' CATM ' ' ICATR
  ICF ' ' OCF ' ' I-PFKEY
  UPON PRINTER.
*----->  --->  ROLLBACK  <---
  IF          IMPART-DEPART = '1'
  DEPART WITH ROLLBACK.

```

GENERATED PROGRAM: PROCEDURE DIVISION
CALLED USER FUNCTIONS

PAGE

130

4

19

CALL	'D\$CLCONV' USING STATUS-WORD	P510
CALL	'D\$USERMSG' USING STATUS-WORD	P520
	ERROR-MESSAGE	P530
MOVE	SPACE TO COMMON-AREA	P540
CALL	'D\$PUTSCR' USING STATUS-WORD	P550
	COMMON-AREA	P560
CALL	'D\$TERM' USING STATUS-WORD.	P580
F99RB-FN.	EXIT.	P000

5. HELP FUNCTION

5.1. INTRODUCTION

INTRODUCTION

End users dynamically access the HELP documentation of a screen or of a data element called in a screen, through the activation of the "HELP" function program.

The purpose of the Help function is to display information of various types contained in the Error Message file.

For information on the character used to call the HELP documentation of a given screen or data element, refer to Chapter "DIALOGUE OR SCREEN DEFINITION" in the ON-LINE SYSTEMS DEVELOPMENT Reference Manual.

USING THE "HELP" PROGRAM

To use the specifications of the "HELP" function in a dialogue, the definition of an additional screen is required.

This "HELP" screen belongs to the Dialogue. Thus, the first two characters of the code must be the same as those of the corresponding dialogue, followed by the "HELP" screen code. For dialogue XX, the "HELP" screen would have the following code: 'XXHELP'.

The 'XXHELP' screen must be defined, but not described. (only the Screen Definition must be created). It must use the same variants as the dialogue. Coding the external names (PROGRAM and MAP) is not restricted and is up to the user.

The user must generate and compile the 'XXHELP' program (the generated COBOL program has the same structure as an on-line screen program).

NOTE

A "HELP" program generated from a dialogue can be used by 'n' dialogues. It is generated once, and the 'XXHELP' screens of the various dialogues must have the same external names (PROGRAM and MAP).

The "HELP" program also ensures the display of the documentation, as follows :

- For screen HELP documentation:
 - . Screen-related documentation (texts and comments),
 - . Segment access error messages.
- For the data element HELP documentation:
 - . Standard error messages, automatically generated,
 - . User-defined error messages,
 - . Data element General Documentation lines (CH: E.....G),
 - . Screen General Documentation lines, associated with the data element (CH: O.....G).

For complete details, refer to Chapter "ERROR MESSAGES - HELP FUNCTION", Subchapter "HELP MESSAGES: CODING" in the ON-LINE SYSTEMS DEVELOPMENT Reference Manual.

The 'HELP' program does not ensure the backup of fields entered before it is called.

This backup may be written by the user, by means of a monorecord database in which all the fields of the screen will be stored. The user may use the terminal code as the access key for this database.

The physical accesses to this database can be written as a macro-structure inserted in function F8095, using the labels F80-HELP-....

NOTE

If the Error Message file is generated with the 'C1' option, only the error messages appear. If it is generated with the 'C2' option, the comments and the documentation associated with the screen are also generated.

HELP FUNCTION

5

INTRODUCTION

1

```

-----
!                APPLICATION UNISYS 2200                *PDSG.NDOC.AU1.9!
! ON-LINE SCREEN DEFINITION.....: DOHELP                !
! !                                                    !
! SCREEN NAME.....: HELP FUNCTION SCREEN                !
! !                                                    !
! SCREEN SIZE (LINES, COLUMNS) .....: 24      080      !
! LABEL TYPE, TABS, INITIALIZATION...: L        01      !
! HELP CHARACTER SCREEN, DATA ELEMENT: !          $      !
! !                                                    !
!                LABELS  DISPLAY  INPUT  ER.MESS.  ER.FLD!
! INTENSITY ATTRIBUTE .....: N        N        N        N        N !
! PRESENTATION ATTRIBUTE .....: N        N        N        N        N !
! COLOR ATTRIBUTE .....: W          W          W          W          W !
! !                                                    !
! TYPE OF COBOL AND MAP TO GENERATE..: U    0      UNISYS 2200    !
! CONTROL CARD OPTIONS FRONT & BACK..:          (PROGRAM)    $$      (MAP) !
! EXTERNAL NAMES .....: WWD050    (PROGRAM)    18      (MAP) !
! TRANSACTION CODE.....: * WWD050                        !
! !                                                    !
! EXPLICIT KEYWORDS..: DO                                !
! SESSION NUMBER.....: 0006          LIBRARY.....: AU1    LOCK....: !
! !                                                    !
! O: C1 CH: Odohelp                ACTION:                !
-----

```

HELP FUNCTION

5

INTRODUCTION

1

```
-----
!
!DOCUMENTATION OF THE SCREEN : *** ORDER INPUT SCREEN ***
!
!
! ON THIS SCREEN YOU ENTER AN ORDER FOR DOCUMENTATION
! FOR ANY GIVEN CLIENT.
! EACH ACCESSIBLE FIELD OF THIS SCREEN IS DOCUMENTED. IN
! ORDER TO OBTAIN THIS DOCUMENTATION, PLACE THE CURSOR
! UNDER THE CHOSEN FIELD AND USE THE PROGRAMMABLE FUNC-
! TION KEY PF11.
! FROM THIS SCREEN, IT IS POSSIBLE TO ACCESS ANY SCREEN
! TRANSACTION BY USING THE OFFERED CHOICES WHICH APPEAR
! AT THE BOTTOM OF THE SCREEN.
! THE UPDATE IS VALIDATED BY THE PROGRAMMABLE FUNCTION
! KEY PF07. IF THE SCREEN APPEARS INSUFFICIENT; IT IS
! POSSIBLE TO SCROLL FORWARD BY USING THE PF08 KEY.
!
! F019 UNKNOWN ZIP CODE.
!
! F028 TECHNICAL PROBLEM CALL E.D.P. DEPT.(CODE 030-CD05 F8)
!
!
!CHOICE.....: S (E: END - T: TOP - S: NEXT)
!
-----
```


5.2. GENERATED 'HELP' PROGRAM

```
IDENTIFICATION DIVISION.  
PROGRAM-ID. WWDO50. DOHELP  
AUTHOR. HELP FUNCTION SCREEN. DOHELP  
DATE-COMPILED. 04/05/94. DOHELP  
ENVIRONMENT DIVISION. DOHELP  
CONFIGURATION SECTION. DOHELP  
SOURCE-COMPUTER. UNIVAC-1100-80. DOHELP  
OBJECT-COMPUTER. UNIVAC-1100-80. DOHELP  
SPECIAL-NAMES. DOHELP  
    DECIMAL-POINT IS COMMA. DOHELP  
INPUT-OUTPUT SECTION. DOHELP  
FILE-CONTROL. DOHELP  
DATA DIVISION. D100  
SUBSCHEMA SECTION. D120  
INVOKE SUBSCHEMA SWWDO D140  
    IN FILE SCH OF SCHEMA WWDO D150  
    SAVE DATA INCLUDES RUN-UNIT QUICK-BEFORE-LOOKS D170  
    DMCA AND RUN-UNIT-STATISTICS ARE WORKING D180  
    ROLLBACK F99RB. D190  
FILE SECTION. DOHELP  
WORKING-STORAGE SECTION. DOHELP  
01 WSS-BEGIN. DOHELP  
    05 FILLER PICTURE X(7) VALUE 'WORKING'. DOHELP  
    05 IK PICTURE X. DOHELP  
    05 BLANC PICTURE X VALUE SPACE. DOHELP  
    05 OPER PICTURE X. DOHELP  
    05 OPERD PICTURE X VALUE SPACE. DOHELP  
    05 CATX PICTURE X. DOHELP  
    05 CATM PICTURE X. DOHELP  
    05 ICATR PICTURE 99. DOHELP  
    05 SCR-ER PICTURE X. DOHELP  
    05 FT PICTURE X. DOHELP  
    05 ICF PICTURE X. DOHELP  
    05 OCF PICTURE X. DOHELP  
    05 CAT-ER PICTURE X. DOHELP  
    05 I-PFKEY. DOHELP  
    10 I-FONCT PICTURE 99 VALUE ZERO. DOHELP  
    05 INA PICTURE 999 VALUE 000. DOHELP  
    05 INR PICTURE 999 VALUE 000. DOHELP  
    05 INZ PICTURE 999 VALUE 001. DOHELP  
    05 IRR PICTURE 99 VALUE 17. DOHELP  
    05 INT PICTURE 999 VALUE 001. DOHELP  
    05 IER PICTURE 99 VALUE 01. DOHELP  
    05 DEL-ER PICTURE X. DOHELP  
01 PACBASE-CONSTANTS. DOHELP  
* OLSD DATES PACE30 : 28/10/93 DOHELP  
* PACE80 : 04/01/94 PAC7SG : 931207 DOHELP  
    05 SESSI PICTURE X(5) VALUE '0382 '. DOHELP  
    05 LIBRA PICTURE X(3) VALUE 'AU1'. DOHELP  
    05 DATGN PICTURE X(8) VALUE '04/05/94'. DOHELP  
    05 PROGR PICTURE X(6) VALUE 'DOHELP'. DOHELP  
    05 PROGE PICTURE X(8) VALUE 'WWDO50 '. DOHELP  
    05 TIMGN PICTURE X(8) VALUE '17:57:32'. DOHELP  
    05 USERCO PICTURE X(8) VALUE 'PDSG '. DOHELP  
    05 5-HELP-PROGE PICTURE X(8). DOHELP  
01 DATCE. DOHELP  
    05 CENTUR PICTURE XX VALUE '19'. DOHELP  
    05 DATOR. DOHELP  
    10 DATOA PICTURE XX. DOHELP  
    10 DATOM PICTURE XX. DOHELP  
    10 DATOJ PICTURE XX. DOHELP  
01 DAT6. DOHELP  
    10 DAT61. DOHELP  
    15 DAT619 PICTURE 99. DOHELP  
    10 DAT62. DOHELP  
    15 DAT629 PICTURE 99. DOHELP  
    10 DAT63 PICTURE XX. DOHELP  
01 DAT7. DOHELP  
    10 DAT71 PICTURE XX. DOHELP  
    10 DAT72 PICTURE XX. DOHELP  
    10 DAT73 PICTURE XX. DOHELP  
01 DAT8. DOHELP
```

HELP FUNCTION

5

GENERATED 'HELP' PROGRAM

2

10	DAT81	PICTURE XX.	DOHELP
10	DAT8S1	PICTURE X.	DOHELP
10	DAT82	PICTURE XX.	DOHELP
10	DAT8S2	PICTURE X.	DOHELP
10	DAT83	PICTURE XX.	DOHELP
01	DATSEP	PICTURE X VALUE '/'. DOHELP	
01	DATSET	PICTURE X VALUE '-'. DOHELP	
01	DATCTY.	DOHELP	
05	DATCTY9	PICTURE 99. DOHELP	
01	DAT6C.	DOHELP	
10	DAT61C	PICTURE XX. DOHELP	
10	DAT62C	PICTURE XX. DOHELP	
10	DAT63C	PICTURE XX. DOHELP	
10	DAT64C	PICTURE XX. DOHELP	
01	DAT7C.	DOHELP	
10	DAT71C	PICTURE XX. DOHELP	
10	DAT72C	PICTURE XX. DOHELP	
10	DAT73C	PICTURE XX. DOHELP	
10	DAT74C	PICTURE XX. DOHELP	
01	DAT8C.	DOHELP	
10	DAT81C	PICTURE XX. DOHELP	
10	DAT8S1C	PICTURE X VALUE '/'. DOHELP	
10	DAT82C	PICTURE XX. DOHELP	
10	DAT8S2C	PICTURE X VALUE '/'. DOHELP	
10	DAT83C	PICTURE XX. DOHELP	
10	DAT84C	PICTURE XX. DOHELP	
01	DAT8G.	DOHELP	
10	DAT81G	PICTURE XX. DOHELP	
10	DAT82G	PICTURE XX. DOHELP	
10	DAT8S1G	PICTURE X VALUE '-'. DOHELP	
10	DAT83G	PICTURE XX. DOHELP	
10	DAT8S2G	PICTURE X VALUE '-'. DOHELP	
10	DAT84G	PICTURE XX. DOHELP	
01	TIMCO.	DOHELP	
02	TIMCOG.	DOHELP	
05	TIMCOH	PICTURE XX. DOHELP	
05	TIMCOM	PICTURE XX. DOHELP	
05	TIMCOS	PICTURE XX. DOHELP	
02	TIMCOC	PICTURE XX. DOHELP	
01	TIMDAY.	DOHELP	
05	TIMHOU	PICTURE XX. DOHELP	
05	TIMS1	PICTURE X VALUE ':'. DOHELP	
05	TIMMIN	PICTURE XX. DOHELP	
05	TIMS2	PICTURE X VALUE ':'. DOHELP	
05	TIMSEC	PICTURE XX. DOHELP	
01	CONFIGURATIONS.	DOHELP	
05	EM00-CF	PICTURE X. DOHELP	
01	K-HELP-CLE.	*AA010	
03	K-RHELP-LIGNE	OCCURS 1. *AA010	
10	K-REM00-EMKEY	PICTURE X(17). *AA010	
COPY	SCREEN-DOHELP-18.	*AA040	
01	FIELD-CONTROL-TABLE REDEFINES	*AA040	
	SCREEN-DOHELP-18-FCA.	*AA040	
05	FILLER	PICTURE XX. *AA040	
05	S-LIBEC-YCO	PICTURE 9(2) COMP. *AA040	
05	S-LIBEC-XCO	PICTURE 9(2) COMP. *AA040	
05	FILLER	PICTURE X(5). *AA040	
05	S-LIBEC-DYN	PICTURE X. *AA040	
05	S-LIBEC-BACK	PICTURE X. *AA040	
05	S-LIBEC-FORE	PICTURE X. *AA040	
05	S-LIBEC-INT	PICTURE X. *AA040	
05	S-LIBEC-HIGH	PICTURE X. *AA040	
05	FILLER	PICTURE X. *AA040	
05	S-LIBEC-EMPH	PICTURE X. *AA040	
05	FILLER	PICTURE XX. *AA040	
05	S-LIENT-YCO	PICTURE 9(2) COMP. *AA040	
05	S-LIENT-XCO	PICTURE 9(2) COMP. *AA040	
05	FILLER	PICTURE X(5). *AA040	
05	S-LIENT-DYN	PICTURE X. *AA040	
05	S-LIENT-BACK	PICTURE X. *AA040	
05	S-LIENT-FORE	PICTURE X. *AA040	
05	S-LIENT-INT	PICTURE X. *AA040	
05	S-LIENT-HIGH	PICTURE X. *AA040	
05	FILLER	PICTURE X. *AA040	
05	S-LIENT-EMPH	PICTURE X. *AA040	
05	J-LIGNE	OCCURS 17. *AA040	
10	FILLER	PICTURE X(16). *AA040	

HELP FUNCTION
GENERATED 'HELP' PROGRAM

PAGE

139

5
2

```
05 FILLER PICTURE XX. *AA040
05 S-LICHOI-YCO PICTURE 9(2) COMP. *AA040
05 S-LICHOI-XCO PICTURE 9(2) COMP. *AA040
05 FILLER PICTURE X(5). *AA040
05 S-LICHOI-DYN PICTURE X. *AA040
05 S-LICHOI-BACK PICTURE X. *AA040
05 S-LICHOI-FORE PICTURE X. *AA040
05 S-LICHOI-INT PICTURE X. *AA040
05 S-LICHOI-HIGH PICTURE X. *AA040
05 FILLER PICTURE X. *AA040
05 S-LICHOI-EMPH PICTURE X. *AA040
05 FILLER PICTURE XX. *AA040
05 S-OPDOC-YCO PICTURE 9(2) COMP. *AA040
05 S-OPDOC-XCO PICTURE 9(2) COMP. *AA040
05 FILLER PICTURE X(5). *AA040
05 S-OPDOC-DYN PICTURE X. *AA040
05 S-OPDOC-BACK PICTURE X. *AA040
05 S-OPDOC-FORE PICTURE X. *AA040
05 S-OPDOC-INT PICTURE X. *AA040
05 S-OPDOC-HIGH PICTURE X. *AA040
05 FILLER PICTURE X. *AA040
05 S-OPDOC-EMPH PICTURE X. *AA040
05 FILLER PICTURE XX. *AA040
05 S-LIOPT-YCO PICTURE 9(2) COMP. *AA040
05 S-LIOPT-XCO PICTURE 9(2) COMP. *AA040
05 FILLER PICTURE X(5). *AA040
05 S-LIOPT-DYN PICTURE X. *AA040
05 S-LIOPT-BACK PICTURE X. *AA040
05 S-LIOPT-FORE PICTURE X. *AA040
05 S-LIOPT-INT PICTURE X. *AA040
05 S-LIOPT-HIGH PICTURE X. *AA040
05 FILLER PICTURE X. *AA040
05 S-LIOPT-EMPH PICTURE X. *AA040
05 FILLER OCCURS 1. *AA040
15 FILLER PICTURE XX. *AA040
15 S-ERMSG-YCO PICTURE 9(2) COMP. *AA040
15 S-ERMSG-XCO PICTURE 9(2) COMP. *AA040
15 FILLER PICTURE X(5). *AA040
15 S-ERMSG-DYN PICTURE X. *AA040
15 S-ERMSG-BACK PICTURE X. *AA040
15 S-ERMSG-FORE PICTURE X. *AA040
15 S-ERMSG-INT PICTURE X. *AA040
15 S-ERMSG-HIGH PICTURE X. *AA040
15 FILLER PICTURE X. *AA040
15 S-ERMSG-EMPH PICTURE X. *AA040
COPY SCREEN-DOHELP-18-DATA. *AA050
01 INPUT-SCREEN-FIELDS REDEFINES *AA050
   SCREEN-DOHELP-18-DATA. *AA050
   I-HELP. *AA050
05 I-HELP-LIBEC PICTURE X(30). *AA050
05 FILLER PICTURE X(02). *AA050
05 I-HELP-LIENT PICTURE X(36). *AA050
05 J-HELP-LIGNE OCCURS 17. *AA050
10 FILLER PICTURE X(76). *AA050
05 I-HELP-LICHOI PICTURE X(19). *AA050
05 FILLER PICTURE X(01). *AA050
05 I-HELP-OPDOC PICTURE X. *AA050
05 FILLER PICTURE X(03). *AA050
05 I-HELP-LIOPT PICTURE X(30). *AA050
05 FILLER PICTURE X(02). *AA050
05 I-HELP-ERMS. *AA050
10 FILLER OCCURS 1. *AA050
15 I-HELP-ERMSG PICTURE X(72). *AA050
01 OUTPUT-SCREEN-FIELDS. *AA050
02 O-HELP. *AA050
05 O-HELP-LIBEC PICTURE X(30). *AA050
05 FILLER PICTURE X(02). *AA050
05 O-HELP-LIENT PICTURE X(36). *AA050
05 P-HELP-LIGNE OCCURS 17. *AA050
10 FILLER PICTURE X(76). *AA050
05 O-HELP-LICHOI PICTURE X(19). *AA050
05 FILLER PICTURE X(01). *AA050
05 O-HELP-OPDOC PICTURE X. *AA050
05 FILLER PICTURE X(03). *AA050
05 O-HELP-LIOPT PICTURE X(30). *AA050
05 FILLER PICTURE X(02). *AA050
05 O-HELP-ERMS. *AA050
```

HELP FUNCTION
 GENERATED 'HELP' PROGRAM

PAGE

140

5
 2

	10	FILLER OCCURS 1.	*AA050
	15	O-HELP-ERMSG PICTURE X(72).	*AA050
01		REPEAT-LINE.	*AA050
	02	I-HELP-LIGNE.	*AA050
	05	I-HELP-ERMSGD PICTURE X(74).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	02	O-HELP-LIGNE.	*AA050
	05	O-HELP-ERMSGD PICTURE X(74).	*AA050
	05	FILLER PICTURE X(02).	*AA050
	02	I-LIGNE.	*AA050
	05	FILLER PICTURE XX.	*AA050
	05	S-ERMSGD-YCO PICTURE 9(2) COMP.	*AA050
	05	S-ERMSGD-XCO PICTURE 9(2) COMP.	*AA050
	05	FILLER PICTURE X(5).	*AA050
	05	S-ERMSGD-DYN PICTURE X.	*AA050
	05	S-ERMSGD-BACK PICTURE X.	*AA050
	05	S-ERMSGD-FORE PICTURE X.	*AA050
	05	S-ERMSGD-INT PICTURE X.	*AA050
	05	S-ERMSGD-HIGH PICTURE X.	*AA050
	05	FILLER PICTURE X.	*AA050
	05	S-ERMSGD-EMPH PICTURE X.	*AA050
01		EM00.	*AA100
	05	EM00-EMKEY.	*AA100
	10	EM00-LIBRA PICTURE X(3).	*AA100
	10	EM00-ENTYP PICTURE X.	*AA100
	10	EM00-XEMKY.	*AA100
	15	EM00-PROGR PICTURE X(6).	*AA100
	15	EM00-ERCOD.	*AA100
	20	EM00-ERCOD9 PICTURE 9(3).	*AA100
	15	EM00-ERTYP PICTURE X.	*AA100
	10	EM00-LINUM PICTURE 9(3).	*AA100
	05	EM00-ERLVL PICTURE X.	*AA100
	05	EM00-ERMSG PICTURE X(66).	*AA100
	05	FILLER PICTURE X(6).	*AA100
01		VALIDATION-TABLE-FIELDS.	*AA150
	02	DE-ERR.	*AA150
	05	DE-ER PICTURE X OCCURS 001.	*AA150
	02	DE-E REDEFINES DE-ERR.	*AA150
	03	ER-HELP-ENDRE.	*AA150
	05	ER-HELP-OPDOC PICTURE X.	*AA150
01		TT-DAT.	*AA200
	05	T-DAT PICTURE X OCCURS 5.	*AA200
01		USERS-ERROR.	*AA200
	05	XEMKY.	*AA200
	10	XPROGR PICTURE X(6).	*AA200
	10	XERCD PICTURE X(4).	*AA200
	05	T-XEMKY OCCURS 01.	*AA200
	10	T-XPROGR PICTURE X(6).	*AA200
	10	T-XERCD PICTURE X(4).	*AA200
01		PACBASE-INDEXES COMPUTATIONAL.	*AA200
	05	TALLI PICTURE S9(4) VALUE ZERO.	*AA200
	05	K01 PICTURE S9(4).	*AA200
	05	K02 PICTURE S9(4).	*AA200
	05	K03 PICTURE S9(4).	*AA200
	05	K04 PICTURE S9(4).	*AA200
	05	K50R PICTURE S9(4) VALUE ZERO.	*AA200
	05	K50L PICTURE S9(4) VALUE ZERO.	*AA200
	05	K50M PICTURE S9(4) VALUE +01.	*AA200
	05	5-CA00-LTH PICTURE S9(4) VALUE +0147.	*AA200
	05	5-EM00-LTH PICTURE S9(4) VALUE +0090.	*AA200
	05	LTH PICTURE S9(4) VALUE ZERO.	*AA200
	05	5-HELP-LENGTH PICTURE S9(4) VALUE +0880.	*AA200
01		TABLE-OF-ATTRIBUTES.	*AA250
	02	DE-ATT.	*AA250
	03	DE-ATT1 OCCURS 4.	*AA250
	05	DE-AT PICTURE X OCCURS 001.	*AA250
	02	DE-A REDEFINES DE-ATT.	*AA250
	03	DE-ATT2 OCCURS 4.	*AA250
	04	A-HELP-ENDRE.	*AA250
	05	A-HELP-OPDOC PICTURE X.	*AA250
01		FIRST-ON-SEGMENT.	*AA301
	05	EM00-FST PICTURE X.	*AA301
01		5-HELP-TRX PICTURE X(6).	*AA400

HELP FUNCTION

5

GENERATED 'HELP' PROGRAM

2

01	END-MESSAGE	PICTURE X(80) VALUE SPACE.	*AA400
01	STOP-FIELDS-HELP.		*AA400
02	C-HELP-LE.		*AA400
05	C-HELP-LIBRA	PICTURE XXX.	*AA400
05	C-HELP-ERCOD	PICTURE XXX.	*AA400
05	C-HELP-PROGR	PICTURE X(6).	*AA400
05	C-HELP-ENTYP	PICTURE X.	*AA400
02	HELP-LIENT	PICTURE X(36) VALUE SPACE.	*AA400
02	HELP-LIBEC	PICTURE X(30) VALUE SPACE.	*AA400
01	7-HELP-LIBEL.		*AA400
05	7-HELP-ERMS.		*AA400
10	7-HELP-ERMSG.		*AA400
15	7-HELP-ERMSG1	PICTURE X(12).	*AA400
15	7-HELP-ERMSG2	PICTURE X(18).	*AA400
10	7-HELP-ERMSC	PICTURE X(36).	*AA400
01	SCREEN-LIGNE.		*AA400
05	7-HELP-ERMSGD	PICTURE X(74).	*AA400
05	7-HELP-CODIF	REDEFINES 7-HELP-ERMSGD.	*AA400
10	7-HELP-VALRU	PICTURE X(12).	*AA400
10	FILLER	PICTURE X.	*AA400
10	7-HELP-SIGNI.		*AA400
15	FILLER	PICTURE X(18).	*AA400
15	7-HELP-ERMSC1	PICTURE X(43).	*AA400
05	7-HELP-DOCUM	REDEFINES 7-HELP-ERMSGD.	*AA400
10	7-HELP-XEMKY.		*AA400
15	FILLER	PICTURE XXX.	*AA400
15	7-HELP-ERTYP	PICTURE X.	*AA400
15	FILLER	PICTURE X.	*AA400
10	7-HELP-LITAC	PICTURE X(69).	*AA400
01	XZ00.		*AA400
10	XZ00-EMKEY	PICTURE X(17).	*AA400
10	XZ00-ERLVL	PICTURE X.	*AA400
10	XZ00-ERMSG	PICTURE X(66).	*AA400
10	FILLER	PICTURE X(6).	*AA400
	COPY INFO-BUFFER.		*AA400
	COPY SENDERROR.		*AA400
	COPY STATUS-WORD.		*AA400
	COPY PAGE-STATUS.		*HE010
	COPY SCREEN-BUFFER.		*HE020
	COPY SCREEN-BUFFER REPLACING SCREEN-BUFFER BY OP-BUFFER.		*HE030
01	CLE-PAGE	PICTURE 9(10) COMP.	*HE040
01	7-WW00.		*WW100
05	FILLER	PICTURE X(6) VALUE 'FNPAC '.	*WW120
05	7-WW00-FONCT	PICTURE X(6).	*WW130
05	FILLER	PICTURE X(7) VALUE ' ORDRE '.	*WW140
05	7-WW00-ORDRE	PICTURE X(8).	*WW150
05	FILLER	PICTURE X(7) VALUE ' RBCOD '.	*WW160
05	7-WW00-RBCODE	PICTURE X(2).	*WW170
05	FILLER	PICTURE X(4) VALUE ' FN '.	*WW180
05	7-WW00-FUNCT	PICTURE X(2).	*WW190
05	FILLER	PICTURE X(7) VALUE ' ERCOD '.	*WW200
05	7-WW00-ERCOD	PICTURE X(2).	*WW210
05	FILLER	PICTURE X(7) VALUE ' ERNUM '.	*WW220
05	7-WW00-NUM	PICTURE X(4).	*WW230
01	DBK-REC USAGE	IS DATABASE-KEY.	*WX200
01	COMMON-AREA.		*00000
02	K-SHELP-PROGR	PICTURE X(6).	*00000
02	CA00.		*00001
10	CA00-CLECD.		*00001
15	CA00-NUCOM	PICTURE 9(5).	*00001
10	CA00-CLECL1.		*00001
15	CA00-NUCLIE	PICTURE 9(8).	*00001
10	CA00-ME00.		*00001
15	CA00-CLEME.		*00001
20	CA00-COPERS	PICTURE X(5).	*00001
20	CA00-NUMORD	PICTURE XX.	*00001
15	CA00-MESSA	PICTURE X(75).	*00001
10	CA00-PREM	PICTURE X.	*00001
10	CA00-LANGU	PICTURE X.	*00001
10	CA00-RAISOC	PICTURE X(50).	*00001
02	K-SHELP-CDOC	PICTURE X.	*00002
02	K-SHELP-PROGE	PICTURE X(8).	*00002
02	K-SHELP-LIBRA	PICTURE XXX.	*00002
02	K-SHELP-PROHE	PICTURE X(8).	*00002
02	K-SHELP-ERCOD.		*00002
05	K-SHELP-ERCOD9	PICTURE 999.	*00002
02	K-SHELP-ERTYP	PICTURE X.	*00002

HELP FUNCTION
 GENERATED 'HELP' PROGRAM

PAGE

142

5
2

```

02      K-SHELP-NULIX.                *00002
05      K-SHELP-LINUM  PICTURE 999.    *00002
02      FILLER          PICTURE X(0700). *00002
PROCEDURE DIVISION.                  *99999
*      *****
*      *
*      *   INITIALIZATIONS           *
*      *
*      *****
F01.      EXIT.                        DOHELP
F0105.    CALL 'D$INIT' USING STATUS-WORD, INFO-BUFFER. DOHELP
          IF STATUS-FATAL      GO TO F81ER. DOHELP
          IF INFO-PREVIOUS-PROGRAM-ID NOT = SPACE DOHELP
          AND INFO-PREVIOUS-PROGRAM-ID NOT = 'LOGON' DOHELP
          CALL 'D$GETSCR' USING STATUS-WORD, COMMON-AREA. DOHELP
          IF STATUS-FATAL      GO TO F81ER. DOHELP
F0105-FN. EXIT.                        DOHELP
F0110.    MOVE ZERO TO CATX FT K50L. DOHELP
          MOVE '1' TO ICF OCF SCR-ER. DOHELP
          MOVE ZERO TO VALIDATION-TABLE-FIELDS. DOHELP
          MOVE SPACE TO CATM OPER OPERD CAT-ER. DOHELP
          MOVE SPACE TO TABLE-OF-ATTRIBUTES. DOHELP
          MOVE ZERO TO CONFIGURATIONS. DOHELP
          MOVE SPACE TO XEMKY. DOHELP
          IF INFO-CONVERSATION NOT = 'Y' DOHELP
          MOVE ZERO TO ICF DOHELP
          CALL 'D$OPEN' USING STATUS-WORD DOHELP
          SCREEN-DOHELP-18. DOHELP
          IF STATUS-FATAL      GO TO F81ER. DOHELP
          MOVE SPACE TO I-HELP O-HELP ERROR-MESSAGE. DOHELP
          IF ICF = ZERO PERFORM F8115 THRU F8115-FN. DOHELP
          MOVE 'X' TO DE-AT (4, 001). DOHELP
          MOVE SPACE TO O-HELP-ERMSG (01). DOHELP
F0110-FN. EXIT.                        DOHELP
*      +-----+
* LEVEL 10 I OPEN DATABASE I P000
*      +-----+ P000
F0115.    MOVE 'F0115' TO 7-WW00-FONCT P100
          MOVE 'IMPART' TO 7-WW00-ORDRE P120
          IMPART ON ERROR GO TO F99RB. P140
          MOVE 'OPEN' TO 7-WW00-ORDRE P200
          OPEN WWA21E USAGE-MODE IS P220
          RETRIEVAL P230
          WWA81E USAGE-MODE IS P240
          RETRIEVAL. P250
          IF ERROR-CODE NOT = ZERO P300
          GO TO F99RB. P300
F0115-FN. EXIT.                        P000
F0120.    MOVE '1' TO OCF. DOHELP
          IF K-SHELP-CDOC = 'D' OR K-SHELP-CDOC = 'R' DOHELP
          MOVE '1' TO ICF GO TO F0120-FN. DOHELP
          MOVE 'A' TO OPER DOHELP
          MOVE SPACE TO K-SHELP-ERTYP DOHELP
          MOVE ZERO TO K-SHELP-LINUM DOHELP
          MOVE 'D' TO K-SHELP-CDOC GO TO F3999-ITER-FT. DOHELP
F0120-FN. EXIT.                        DOHELP
F01-FN.   EXIT.                        DOHELP
*      *****
*      *
*      *   RECEPTION                 *
*      *
*      *****
F05.     IF ICF = ZERO GO TO END-OF-RECEPTION. DOHELP
F0510.   MOVE INFO-FUNCTION-KEY TO I-FONCT. DOHELP
          IF I-PFKEY NOT = ZERO GO TO F0510-FN. DOHELP
          CALL 'D$READ' USING STATUS-WORD DOHELP
          SCREEN-DOHELP-18. DOHELP
          IF STATUS-FATAL      GO TO F81ER. DOHELP
          MOVE SCREEN-DOHELP-18-DATA TO DOHELP
          O-HELP DOHELP
          MOVE 'A' TO OPER MOVE SPACE TO OPERD. DOHELP
F0510-FN. EXIT.                        DOHELP

```

HELP FUNCTION
 GENERATED 'HELP' PROGRAM

PAGE

143

5
 2

```

*          *****
*          *
*          *   VALIDATION OF OPERATION CODE   *
*          *
*          *****
F0520.
      IF I-HELP-OPDOC = 'E' OR 'F'
      MOVE K-SHELP-PROGE TO 5-HELP-PROGE
      MOVE 'O' TO OPER OPERD GO TO F0520-900.
      IF I-HELP-OPDOC = 'T' OR 'D'
      MOVE SPACE TO K-SHELP-ERCOD K-SHELP-ERTYP
      MOVE ZERO TO K-SHELP-LINUM
      MOVE 'A' TO OPER GO TO F0520-900.
      IF I-HELP-OPDOC = 'S'
      MOVE 'A' TO OPER GO TO F0520-900.
      MOVE '5' TO ER-HELP-OPDOC MOVE '4' TO SCR-ER
      GO TO F3999-ITER-FT.
F0520-900.
      IF OPER NOT = 'A' AND OPER NOT = 'O'
      GO TO F3999-ITER-FT.
F0520-FN.      EXIT.
F05-FN.      EXIT.
*          *****
*          *
*          *   CATEGORY PROCESSING LOOP   *
*          *
*          *****
F10.      EXIT.
F1010.      MOVE SPACE TO CATM.
      IF CAT-ER = 'E' MOVE '4' TO SCR-ER GO TO F3999-ITER-FT.
      MOVE SPACE TO CAT-ER.
      IF CATX = '0' MOVE 'Z' TO CATX GO TO F1010-FN.
F1010-A.      GO TO F3999-ITER-FT.
F1010-FN.      EXIT.
F10-FN.      EXIT.
*          *****
*          *
*          *   DATA ELEMENT VALIDATION   *
*          *
*          *****
F20.      EXIT.
F20Z.      IF CATX NOT = 'Z' GO TO F20Z-FN.
F20A7.
      IF I-HELP-OPDOC NOT = SPACE
      MOVE '1' TO ER-HELP-OPDOC.
F20A7-FN.      EXIT.
F20Z-FN.      EXIT.
F20-FN.      EXIT.
F3999-ITER-FI.      GO TO F10.
F3999-ITER-FT.      EXIT.
F3999-FN.      EXIT.
F40.      IF SCR-ER > '1' MOVE 'A' TO OPER GO TO F40-FN.
F40-A.      IF OPERD NOT = SPACE MOVE OPERD TO OPER.
F4005.      IF OPER NOT = 'O' GO TO F4005-FN.
      IF K-SHELP-CDOC = 'D'
      MOVE '2' TO K-SHELP-CDOC.
      IF K-SHELP-CDOC = 'R'
      MOVE '3' TO K-SHELP-CDOC.
      MOVE ZERO TO K-SHELP-LINUM.
      IF K-SHELP-ERCOD = SPACE
      OR K-SHELP-ERCOD NOT NUMERIC
      MOVE '001' TO K-SHELP-ERCOD.
      IF K-SHELP-ERCOD > '001'
      SUBTRACT 1 FROM K-SHELP-ERCOD9.
F4005-FN.      EXIT.
F4010.      IF OPER NOT = 'A' GO TO F4010-FN.
      MOVE SPACE TO EM00-EMKEY
      MOVE K-SHELP-LIBRA TO EM00-LIBRA
      MOVE 'H' TO EM00-ENTYP
      MOVE K-SHELP-PROGR TO EM00-PROGR
      MOVE K-SHELP-ERCOD TO EM00-ERCOD
      MOVE K-SHELP-ERTYP TO EM00-ERTYP
      MOVE K-SHELP-LINUM TO EM00-LINUM
      MOVE EM00-EMKEY TO K-REM00-EMKEY (1).
F4010-FN.      EXIT.
*          +-----+
* LEVEL 10      I END OF TRANSACTION      I

```

HELP FUNCTION
 GENERATED 'HELP' PROGRAM

PAGE

144

5
2

```

*          +-----+
F4029.    IF      OPER = 'E'                                P000
          NEXT SENTENCE ELSE GO TO      F4029-FN.          P000
          MOVE    '** END OF TRANSACTION **' TO           P100
          END-MESSAGE.                                     P110
F4029-FN. EXIT.                                           P000
*          *****                                       DOHELP
*          *                                           DOHELP
*          * END OF TRANSACTION                          DOHELP
*          *                                           DOHELP
*          *****                                       DOHELP
F4030.    IF OPER NOT = 'E' GO TO F4030-FN.                DOHELP
          PERFORM F81FI THRU F81FI-FN.                    DOHELP
          CALL 'D$CLCONV' USING STATUS-WORD.              DOHELP
          IF STATUS-FATAL GO TO F81ER.                   DOHELP
          CALL 'D$ENDMSG' USING STATUS-WORD, END-MESSAGE. DOHELP
          IF STATUS-FATAL GO TO F81ER.                   DOHELP
          CALL 'D$CLOSE' USING STATUS-WORD.              DOHELP
          IF STATUS-FATAL GO TO F81ER.                   DOHELP
          STOP RUN.                                       DOHELP
F4030-FN. EXIT.                                           DOHELP
*          +-----+
* LEVEL 10  I END OF HELP PROGRAM                          I      P000
*          +-----+
*          +-----+
F4040.    IF      OPER = 'O'                                P000
          NEXT SENTENCE ELSE GO TO      F4040-FN.          P000
          MOVE    'F4040' TO 7-WW00-FONCT                 P100
          PERFORM F81FI THRU F81FI-FN                     P110
          MOVE    ZERO TO K-SHELP-CDOC                   P120
          MOVE    ZERO TO K-SHELP-NUERR                  P130
          MOVE    'D$PUTSCR' TO 7-WW00-ORDRE              P140
          CALL    'D$PUTSCR' USING STATUS-WORD            P150
          COMMON-AREA.                                    P160
          IF      STATUS-FATAL                            P170
          GO TO  F81ER.                                    P170
          MOVE    'D$PAGEST' TO 7-WW00-ORDRE              P200
          CALL    'D$PAGEST' USING STATUS-WORD            P210
          PAGE-STATUS-BUFFER.                            P220
          IF      STATUS-FATAL                            P230
          GO TO  F81ER.                                    P230
          IF      NO-DATA-PAGES                           P240
          GO TO  F81ER.                                    P240
          MOVE    1 TO CLE-PAGE                           P260
          MOVE    'D$RETR' TO 7-WW00-ORDRE                P270
          CALL    'D$RETR' USING STATUS-WORD              P280
          SCREEN-BUFFER.                                  P290
          CLE-PAGE.                                       P300
          IF      STATUS-FATAL                            P310
          GO TO  F81ER.                                    P310
          MOVE    'D$RELEAS' TO 7-WW00-ORDRE              P330
          CALL    'D$RELEASE' USING STATUS-WORD.          P340
          IF      STATUS-FATAL                            P350
          GO TO  F81ER.                                    P350
          MOVE    SCREEN-BUFFER TO OP-BUFFER              P400
          MOVE    'D$OPEN' TO 7-WW00-ORDRE                P410
          CALL    'D$OPEN' USING STATUS-WORD              P420
          OP-BUFFER.                                       P430
          IF      STATUS-FATAL                            P440
          GO TO  F81ER.                                    P440
          MOVE    5-HELP-PROGE TO 5-HELP-TRX              P450
          MOVE    'D$CLCONV' TO 7-WW00-ORDRE              P460
          CALL    'D$CLCONV' USING STATUS-WORD            P470
          OP-BUFFER.                                       P480
          IF      STATUS-FATAL                            P490
          GO TO  F81ER.                                    P490
          MOVE    'D$SETRX' TO 7-WW00-ORDRE               P500
          CALL    'D$SETRX' USING STATUS-WORD             P510
          5-HELP-TRX.                                       P520
          IF      STATUS-FATAL                            P530
          GO TO  F81ER.                                    P530
          MOVE    'D$SEND' TO 7-WW00-ORDRE                P540
          CALL    'D$SEND' USING STATUS-WORD              P550
          SCREEN-BUFFER.                                  P560
          IF      STATUS-FATAL                            P570
          GO TO  F81ER.                                    P570
          MOVE    'D$TERM' TO 7-WW00-ORDRE                P600
          CALL    'D$TERM' USING STATUS-WORD              P610
  
```


HELP FUNCTION
 GENERATED 'HELP' PROGRAM

PAGE

145

5
 2

```

                IF      STATUS-FATAL
GO TO F81ER.
F4040-FN.      EXIT.
F40-FN.        EXIT.
END-OF-RECEPTION.  EXIT.
*             *****
*             *
*             * DISPLAY PREPARATION
*             *
*             *****
F50.           IF OCF = '0' GO TO END-OF-DISPLAY.
F5010.
                MOVE ZERO TO CATX.
                MOVE ZERO TO CONFIGURATIONS.
                MOVE ALL '1' TO FIRST-ON-SEGMENT.
                IF SCR-ER > '1' GO TO F6999-ITER-FT.
                MOVE SPACE TO O-HELP.
                PERFORM F8115 THRU F8115-FN.
F5010-FN.      EXIT.
F5020.         IF K-SHELP-ERTYP NOT = SPACE
                NEXT SENTENCE ELSE GO TO F5020-FN.
                MOVE SPACE TO EM00-ERTYP.
                IF K-SHELP-ERCOD < '001'
                MOVE SPACE TO EM00-ERCOD.
                MOVE ZERO TO EM00-LINUM
                PERFORM F80-EM00-P THRU F80-FN.
                IF IK = '1' GO TO F5020-FN.
                IF EM00-ERCOD NOT = SPACE
                MOVE EM00-ERMSG TO 7-HELP-ERMS
                MOVE 7-HELP-ERMSC TO HELP-LIENT
                MOVE 'DOCUMENTATION OF DATA ELEMENT '
                  TO HELP-LIBEC ELSE
                MOVE EM00-ERMSG TO HELP-LIENT
                MOVE 'DOCUMENTATION OF THE SCREEN '
                  TO HELP-LIBEC.
F5020-FN.      EXIT.
F50-FN.        EXIT.
*             *****
*             *
*             * CATEGORY PROCESSING LOOP
*             *
*             *****
F55.           EXIT.
F5510.
                MOVE SPACE TO CAT-ER.
                IF CATX = '0' MOVE ' ' TO CATX GO TO F5510-FN.
                IF CATX = ' ' MOVE 'R' TO CATX MOVE ZERO TO ICATR.
                IF CATX NOT = 'R' OR ICATR > IRR GO TO F5510-R.
                IF ICATR > ZERO
                MOVE O-HELP-LIGNE TO
                  P-HELP-LIGNE (ICATR).
                ADD 1 TO ICATR.
                IF ICATR NOT > IRR
                MOVE P-HELP-LIGNE (ICATR) TO
                  O-HELP-LIGNE.
                GO TO F5510-FN.
F5510-R.       EXIT.
F5510-Z.
                IF CATX = 'R' MOVE 'Z' TO CATX GO TO F5510-FN.
F5510-900.    GO TO F6999-ITER-FT.
F5510-FN.      EXIT.
F55-FN.        EXIT.
*             *****
*             *
*             * SEGMENT ACCESS FOR DISPLAY
*             *
*             *****
F60.           EXIT.
F60R.         IF CATX NOT = 'R' OR FT = '1' GO TO F60R-FN.
F60R-FN.       EXIT.
F6010.        IF CATX NOT = 'R' OR FT = '1' GO TO F6010-FN.
                MOVE '0' TO EM00-CF.
                IF EM00-FST = '1'
                MOVE K-REM00-EMKEY (1) TO EM00-EMKEY
                MOVE EM00-LIBRA TO C-HELP-LIBRA
                MOVE EM00-ENTYP TO C-HELP-ENTYP
                MOVE EM00-PROGR TO C-HELP-PROGR

```

HELP FUNCTION
GENERATED 'HELP' PROGRAM

PAGE

146

5
2

```
MOVE          EM00-ERCOD   TO C-HELP-ERCOD           DOHELP
PERFORM F80-EM00-P   THRU F80-FN                     DOHELP
MOVE ZERO TO EM00-FST   ELSE                          DOHELP
PERFORM F80-EM00-RN THRU F80-FN.                     DOHELP
IF IK = '0'                                             DOHELP
  IF          EM00-LIBRA NOT = C-HELP-LIBRA          DOHELP
  OR          EM00-ENTYP NOT = C-HELP-ENTYP          DOHELP
  OR          EM00-PROGR NOT = C-HELP-PROGR          DOHELP
MOVE '1' TO IK.                                        DOHELP
IF IK = '1' MOVE 'G109' TO XERCD MOVE '1' TO FT      DOHELP
PERFORM F81UT THRU F81UT-FN      GO TO F6010-FN.     DOHELP
MOVE '1' TO EM00-CF.                                  DOHELP
MOVE EM00-ERCOD   TO K-SHELP-ERCOD                   DOHELP
MOVE EM00-ERTYP   TO K-SHELP-ERTYP                   DOHELP
MOVE EM00-LINUM   TO K-SHELP-LINUM.                  DOHELP
IF EM00-ERCOD NOT = C-HELP-ERCOD                     DOHELP
AND EM00-ERCOD > '000'                               DOHELP
MOVE '1' TO FT      GO TO F6010-FN.                  DOHELP
IF EM00-ERTYP = SPACE                               DOHELP
NEXT SENTENCE ELSE GO TO F6010-FN.                   DOHELP
IF EM00-ERCOD > ZERO                                DOHELP
MOVE EM00-ERMSG   TO 7-HELP-ERMS                     DOHELP
MOVE 7-HELP-ERMSC TO HELP-LIENT                      DOHELP
MOVE 'DOCUMENTATION OF DATA ELEMENT '              DOHELP
      TO HELP-LIBEC                                   DOHELP
      ELSE                                           DOHELP
MOVE EM00-ERMSG   TO HELP-LIENT                      DOHELP
MOVE 'DOCUMENTATION OF THE SCREEN '                  DOHELP
      TO HELP-LIBEC.                                  DOHELP
GO TO F6010.                                          DOHELP
F6010-FN.      EXIT.                                  DOHELP
F60-FN.        EXIT.                                  DOHELP
*              *****                               DOHELP
*              *                                     DOHELP
*              *   DATA ELEMENT TRANSFER           DOHELP
*              *                                     DOHELP
*              *****                               DOHELP
F65.           EXIT.                                  DOHELP
F6520. IF FT = '1' OR EM00-ERTYP = ' ' GO TO F6520-FN. DOHELP
IF ICATR > IRR GO TO F6520-FN.                       DOHELP
MOVE SPACE TO 7-HELP-ERMSGD.                         DOHELP
IF EM00-ERTYP = '1'                                  DOHELP
MOVE EM00-ERMSG   TO 7-HELP-ERMS                     DOHELP
MOVE 7-HELP-ERMSG2 TO 7-HELP-SIGNI                   DOHELP
MOVE 7-HELP-ERMSC TO 7-HELP-ERMSC1                   DOHELP
MOVE 7-HELP-ERMSG1 TO 7-HELP-VALRU                   DOHELP
GO TO F6520-900.                                     DOHELP
IF EM00-ERTYP = '0'                                  DOHELP
MOVE SPACE        TO 7-HELP-XEMKY                     DOHELP
MOVE EM00-ERMSG   TO 7-HELP-LITAC                     DOHELP
GO TO F6520-900.                                     DOHELP
MOVE EM00-ERMSG   TO 7-HELP-LITAC.                   DOHELP
IF EM00-LINUM NOT = ZERO                             DOHELP
GO TO F6520-900.                                     DOHELP
MOVE EM00-ERCOD   TO 7-HELP-XEMKY                     DOHELP
MOVE EM00-ERTYP   TO 7-HELP-ERTYP.                   DOHELP
F6520-900.                                           DOHELP
MOVE 7-HELP-ERMSGD TO O-HELP-ERMSGD.                 DOHELP
F6520-FN.      EXIT.                                  DOHELP
F6530. IF CATX NOT = 'Z' GO TO F6530-FN.             DOHELP
MOVE HELP-LIENT   TO O-HELP-LIENT                     DOHELP
MOVE HELP-LIBEC   TO O-HELP-LIBEC.                   DOHELP
MOVE 'CHOICE.....:' TO O-HELP-LICHOI                 DOHELP
MOVE '(E: END - T: TOP - S: NEXT) '                 DOHELP
      TO O-HELP-LIOPT.                                DOHELP
IF XERCD NOT = 'G109'                                 DOHELP
MOVE 'S' TO O-HELP-OPDOC GO TO F6530-FN.             DOHELP
MOVE 'E' TO O-HELP-OPDOC.                             DOHELP
IF K-SHELP-ERCOD NUMERIC AND K-SHELP-ERCOD > ZERO   DOHELP
ADD 1 TO K-SHELP-ERCOD9.                              DOHELP
F6530-FN.      EXIT.                                  DOHELP
F65-FN.        EXIT.                                  DOHELP
F6999-ITER-FI. GO TO F55.                             DOHELP
F6999-ITER-FT. EXIT.                                  DOHELP
F6999-FN.      EXIT.                                  DOHELP
F70.           GO TO F7020.                           DOHELP
```


HELP FUNCTION

5

GENERATED 'HELP' PROGRAM

2

```

*          *          *          DOHELP
*          *  DISPLAY          *          DOHELP
*          *          *          *          DOHELP
*          *          *          *          DOHELP
*          *          *          *          DOHELP
F8Z10.
  IF SCR-ER NOT > '1'          DOHELP
  AND DE-AT (4, 001) = 'X'     DOHELP
  PERFORM F7020 THRU F7020-FN. DOHELP
  CALL 'D$SETCV' USING STATUS-WORD. DOHELP
  IF STATUS-FATAL GO TO F81ER. DOHELP
  CALL 'D$PUTSCR' USING STATUS-WORD, COMMON-AREA. DOHELP
  IF STATUS-FATAL GO TO F81ER. DOHELP
  MOVE ZERO TO S18-OUT-FID DOHELP
  MOVE ERROR-X TO S18-OUT-XCO DOHELP
  MOVE ERROR-Y TO S18-OUT-YCO. DOHELP
  MOVE O-HELP TO SCREEN-DOHELP-18-DATA DOHELP
  CALL 'D$SEND' USING STATUS-WORD DOHELP
  SCREEN-DOHELP-18. DOHELP
  IF STATUS-FATAL GO TO F81ER. DOHELP
F8Z10-FN. EXIT. DOHELP
*          *          *          DOHELP
*          *          *          *          DOHELP
*          *  END OF PROGRAM          *          DOHELP
*          *          *          *          DOHELP
*          *          *          *          DOHELP
F8Z20.
  PERFORM F81FI THRU F81FI-FN. DOHELP
  CALL 'D$CLOSE' USING STATUS-WORD. DOHELP
  IF STATUS-FATAL GO TO F81ER. DOHELP
  STOP RUN. DOHELP
F8Z20-FN. EXIT. DOHELP
F8Z-FN. EXIT. DOHELP
*          *          *          DOHELP
*          *          *          *          DOHELP
*          *  PHYSICAL SEGMENT ACCESS ROUTINES *          DOHELP
*          *          *          *          DOHELP
*          *          *          *          DOHELP
F80.
*          +-----+          P000
* LEVEL 10  I ACCESS ERROR FILE          I          P000
*          +-----+          P000
F8098. EXIT. P000
F80-EM00-P. EXIT. P005
F80-EM00-R. P010
  MOVE 'F80EM00' TO 7-WW00-FONCT P100
  MOVE 'FETCH' TO 7-WW00-ORDRE P120
  MOVE EM00-EMKEY TO ER00-EMKEY P130
  FETCH ER00 RECORD. P150
  IF IK = ZERO P200
  MOVE ER00 TO EM00 P200
  GO TO F80-OK. P220
  IF IK NOT > 2 P250
  GO TO F80-KO. P250
  PERFORM F99RB THRU F99RB-FN. P260
F80-EM00-RN. P300
  MOVE 'FETCH NX' TO 7-WW00-ORDRE P320
  FETCH NEXT ER00 RECORD P330
  PERFORM F98ER THRU F98ER-FN. P340
  IF IK = ZERO P350
  MOVE ER00 TO EM00 P350
  GO TO F80-OK. P360
  IF IK NOT > 2 P370
  GO TO F80-KO. P370
  PERFORM F99RB THRU F99RB-FN. P380
F8098-FN. EXIT. P000
F80-OK. MOVE '0' TO IK MOVE PROGR TO XPROGR GO TO F80-FN. DOHELP
F80-KO. MOVE '1' TO IK MOVE PROGR TO XPROGR. DOHELP
F8099-FN. EXIT. DOHELP
F80-FN. EXIT. DOHELP
F81. DOHELP
*          +-----+          P000
* LEVEL 10  I DPS ERROR          I          P000
*          +-----+          P000
F81ER. P000
  PERFORM F81FI THRU F81FI-FN P100
  MOVE STATUS-FONCTION TO 7-WW00-FUNCT P110
  MOVE STATUS-CODE TO 7-WW00-ERCOD P120

```

HELP FUNCTION
 GENERATED 'HELP' PROGRAM

PAGE

149

5
2

```

      MOVE          7-WW00 TO END-MESSAGE.                                P140
*      +-----+
* LEVEL 15        I ERROR DPS MANAGEMENT                                I      P000
*      +-----+
F81ES.
*----->        FUNCTION KEY MSG-WAIT                                  P010
                IF STATUS-FUNCTION = 05                               P100
                AND (STATUS-CODE = 31 OR 34)                          P110
      CALL        'D$RESET' USING STATUS-WORD.                         P100
                IF STATUS-FATAL                                       P120
                GO TO F81ER-FN.                                       P120
F81ES-FN.        EXIT.                                               P000
*      +-----+
* LEVEL 15        I DISPLAY DPS ERROR                                I      P000
*      +-----+
F81EV.
      DISPLAY     '*****  DPS      ERROR  *****'                    P100
                UPON PRINTER                                          P110
      DISPLAY     'PROGRAM          : ' PROGR                          P120
                UPON PRINTER                                          P130
      DISPLAY     'FUNCT. PACBASE   : '                                P140
                7-WW00-FONCT                                          P150
                UPON PRINTER                                          P160
      DISPLAY     'DPS ORDER       : '                                P170
                7-WW00-ORDRE                                          P180
                UPON PRINTER                                          P190
      DISPLAY     'STATUS-FUNCTION : '                                P200
                STATUS-FUNCTION                                       P210
                UPON PRINTER                                          P220
      DISPLAY     'STATUS-CODE     : '                                P230
                STATUS-CODE                                           P240
                UPON PRINTER                                          P250
                IF IMPART-DEPART = '1'                                P300
      DEPART WITH ROLLBACK.                                           P300
      CALL        'D$CLCONV' USING STATUS-WORD                         P310
      CALL        'D$ERRMSG' USING STATUS-WORD                         P320
      MOVE        SPACE TO COMMON-AREA                                P340
      CALL        'D$PUTSCR' USING STATUS-WORD                         P360
                COMMON-AREA                                           P370
      CALL        'D$TERM' USING STATUS-WORD.                          P400
F81EV-FN.        EXIT.                                               P000
F81ER-FN.        EXIT.                                               P000
*      +-----+
* LEVEL 10        I CLOSE DATABASE                                I      P000
*      +-----+
F81FI.
      CLOSE ALL ON ERROR GO TO F99RB.                                  P100
      DEPART     ON ERROR GO TO F99RB.                                  P200
F81FI-FN.        EXIT.                                               P000
*      *****
*      *
*      *   MEMORIZATION OF USER'S ERRORS
*      *
*      *****
F81UT.          IF K50L < K50M ADD 1 TO K50L                            DOHELP
                MOVE XEMKY TO T-XEMKY (K50L). MOVE 'E' TO CAT-ER.    DOHELP
F81UT-FN.        EXIT.                                               DOHELP
F8115.
F8115-FN.        EXIT.                                               DOHELP
F81-FN.          EXIT.                                               DOHELP
*      +-----+
* LEVEL 10        I DMS ERROR                                I      P000
*      +-----+
F98ER.
      IF          ERROR-STATUS = ZERO                                  P100
      MOVE        ZERO TO IK                                          P100
      GO TO      F98ER-FN.                                           P110
      MOVE        '5' TO IK.                                          P120
*      +-----+
* LEVEL 15        I FETCH COMMAND ERROR                                I      P000
*      +-----+
F98ES.          IF ERROR-FUNCTION = '03'                                P000
                NEXT SENTENCE ELSE GO TO F98ES-FN.                    P000
      IF          ERROR-CODE = '06'                                    P100
      MOVE        '4' TO IK                                          P100
      GO TO      F98ER-FN.                                           P110
      IF          ERROR-CODE = '07'                                    P120

```

HELP FUNCTION
 GENERATED 'HELP' PROGRAM

PAGE

150

5
 2

MOVE	'1' TO IK	P120
	GO TO F98ER-FN.	P130
MOVE	ERROR-FUNCTION TO 7-WW00-FUNCT	P200
MOVE	'2' TO IK	P210
	GO TO F98ER-FN.	P220
F98ES-FN.	EXIT.	P000
F98ER-FN.	EXIT.	P000
*	+-----+	P000
* LEVEL 10	I ROLL-BACK ERROR I	P000
*	+-----+	P000
F99RB.		P000
MOVE	RB-ERROR-CODE TO 7-WW00-RBCODE	P100
MOVE	ERROR-FUNCTION TO 7-WW00-FUNCT	P110
MOVE	ERROR-CODE TO 7-WW00-ERCOD	P120
MOVE	ERROR-NUM TO 7-WW00-NUM	P130
MOVE	7-WW00 TO ERROR-MESSAGE	P140
*----->	--> DISPLAY DMS ERROR <----	P200
DISPLAY	'***** DMS ERROR *****'	P210
	UPON PRINTER	P220
DISPLAY	'PROGRAM : ' PROGE	P300
	UPON PRINTER	P310
DISPLAY	'ERROR-STATUS : ' ERROR-STATUS	P320
	UPON PRINTER	P330
DISPLAY	'ERROR-NUM : ' ERROR-NUM	P340
	UPON PRINTER	P350
DISPLAY	'ERROR-AREA : ' ERROR-AREA	P360
	UPON PRINTER	P370
DISPLAY	'ERROR-RECORD : ' ERROR-RECORD	P380
	UPON PRINTER	P390
DISPLAY	'ERROR-SET : ' ERROR-SET	P400
	UPON PRINTER	P410
DISPLAY	'IK OPER CATX CATM ICATR ICF '	P420
	'OCF I-PFKEY'	P425
	UPON PRINTER	P430
DISPLAY	' ' IK ' ' OPER ' ' CATX	P440
	' ' CATM' ' ICATR	P445
	ICF ' ' OCF ' ' I-PFKEY	P447
	UPON PRINTER.	P450
*----->	--> ROLLBACK <----	P490
IF	IMPART-DEPART = '1'	P500
DEPART WITH	ROLLBACK.	P500
CALL	'D\$CLCONV' USING STATUS-WORD	P510
CALL	'D\$USERMSG' USING STATUS-WORD	P520
	ERROR-MESSAGE	P530
MOVE	SPACE TO COMMON-AREA	P540
CALL	'D\$PUTSCR' USING STATUS-WORD	P550
	COMMON-AREA	P560
CALL	'D\$TERM' USING STATUS-WORD.	P580
F99RB-FN.	EXIT.	P000

6. CHART OF VARIABLES AND CONSTANTS

```

+-----+
!           CHART OF ON-LINE CONSTANTS AND VARIABLES           !
+-----+
!           !           !
! CURPOS ! CURSOR POSITIONING IN RECEPTION SCREEN WHERE !
!           ! CPOSL = LINE NUMBER & CPOSC = COLUMN NUMBER !
!           ! (except for DPS7 FORMS). !
!           !           !
! CPOSN  ! "ABSOLUTE" CURSOR POSITIONING WHERE CPOSL = 1 !
!           ! AND CPOSC = 1 !
!           ! (except for DPS7 FORMS). !
!           !           !
! INA    ! NUMBER OF DATA ELEMENTS IN SCREEN-TOP CATEGORY !
!           !           !
! INR    ! INA + NUMBER OF DATA ELEMENTS IN REPETITIVE !
!           ! CATEGORY !
!           !           !
! INZ    ! INR + NUMBER OF DATA ELEMENTS IN SCREEN-BOTTOM !
!           ! CATEGORY !
!           !           !
! IRR    ! NUMBER OF REPETITIONS IN REPETITIVE CATEGORY !
!           !           !
! INT    ! NUMBER OF INPUT FIELDS IN SCREEN !
!           !           !
! IER    ! NUMBER OF SCREEN-RELATED ERROR MESSAGES !
!           !           !
! SESSI  ! SESSION NUMBER OF GENERATED PROGRAM !
!           !           !
! LIBRA  ! LIBRARY CODE !
!           !           !
! USERCO ! USER CODE !
!           !           !
! DATGN  ! DATE OF GENERATED PROGRAM !
!           !           !
! TIMGN  ! TIME OF GENERATED PROGRAM !
!           !           !
! PROGR  ! PROGRAM CODE !
!           !           !
! PROGE  ! PROGRAM EXTERNAL NAME !
!           !           !
! PRCOC  ! HELP PROGRAM EXTERNAL NAME !
!           !           !
+-----+

```



```
+-----+
!      CHART OF ON-LINE CONSTANTS AND VARIABLES  (CONT'D)  !
+-----+
!      !      !
! DATOR ! YEAR-MONTH-DAY FORMATTED MACHINE DATE      !
!      !      !
! DATSEP ! SEPARATOR USED IN DATES                    !
!      !      !
!      !      !
! DAT6   ! DATE FORMATTING: DDMYY OR YMMDD        !
! DAT7   ! ALSO OUTPUT FORMATS (DD/MM/YY FOR INSTANCE) IF !
! DAT8   ! A VARIABLE DATA ELEMENT (V) HAS A DATE FORMAT !
!      !      !
! DATCTY ! FIELD FOR CENTURY LOAD                    !
!      !      !
! DAT6C  ! NON-FORMATTED DATE WITH CENTURY        !
! DAT7C  !                                         !
!      !      !
! DAT8C  ! FORMATTED DATE WITH CENTURY: MM/DD/CCYY !
!      !      !
! DAT8G  ! GREGORIAN FORMATTED DATE: CCYY/MM/DD   !
!      !      !
! TIMCO  ! TIME                                    !
!      !      !
! TIMDAY ! FORMATTED TIME: HH:MM:SS              !
!      !      !
! 5-scrn-! THIS FIELD CONTAINS THE NAME OF THE     !
! PROGE  ! PROGRAM TO BRANCH TO                 !
!      !      !
+-----+
```

```

+-----+
!           CHART OF VALIDATION VARIABLES AND INDICATORS           !
+-----+
!           !
! ICF      ! CONFIGURATION VARIABLE                                     !
!           ! '1' = SCREEN IN INPUT                               !
!           ! '0' = NO SCREEN IN INPUT                               !
!           !
! OCF      ! CONFIGURATION VARIABLE                                     !
!           ! '1' = SCREEN IN OUTPUT                               !
!           ! '0' = NO SCREEN IN OUTPUT                               !
!           !
! OPER     ! OPERATION CODE                                           !
!           ! 'A' = INQUIRY                                           !
!           ! 'M' = UPDATE                                           !
!           ! 'S' = SCREEN CONTINUATION                             !
!           ! 'E' = CONVERSATION END                               !
!           ! 'P' = PREVIOUS DISPLAY                                 !
!           ! 'O' = TRANSFER TO ANOTHER SCREEN                     !
!           !
! OPERD    ! OPERATION CODE FOR DEFERRED BRANCHING                       !
!           ! 'O' = DEFERRED CALL OF ANOTHER SCREEN                 !
!           ! INITIALIZED IN F0520 AND MOVED INTO OPER IN F40     !
!           !
! CATX     ! CATEGORY BEING PROCESSED                                   !
!           ! '0' = BEGINNING OF RECEPTION OR DISPLAY             !
!           ! ' ' = SCREEN TOP                                   !
!           ! 'R' = REPETITIVE CATEGORY                         !
!           ! 'Z' = SCREEN BOTTOM                                 !
!           !
! CATM     ! TRANSACTION CODE                                           !
!           ! 'C' = CREATION                                           !
!           ! 'M' = MODIFICATION                                       !
!           ! 'A' = DELETION                                           !
!           ! 'X' = IMPLICIT UPDATE                                       !
!           !
! ICATR    ! INDICATOR OF CATEGORY BEING PROCESSED                   !
!           ! (REPETITIVE CATEGORY ONLY)                           !
!           !
! FT       ! END OF REPETITIVE CATEGORY INDICATOR                   !
!           ! '0' LINES TO DISPLAY                                   !
!           ! '1' NO MORE LINES TO DISPLAY                           !
!           !
! ddss-CF ! SEGMENT CONFIGURATION INDICATOR (seg. ddss)                 !
!           ! '1' THE SEGMENT IS PROCESSED                           !
!           ! '0' THE SEGMENT IS NOT PROCESSED                       !
!           !
+-----+

```

```
+-----+
!      CHART OF VALIDATION VARIABLES AND INDICATORS (CONT'D) !
+-----+
! IK      ! PHYSICAL FILE ACCESS ERROR INDICATOR      !
!         ! '0' NO ERROR                                !
!         ! '1' ERROR                                    !
!         !                                         !
+-----+

+-----+
!                      ERROR VARIABLES          !
+-----+
!         !
! SCR-ER ! STORAGE OF SCREEN ERROR                  !
!         ! '1' NO ERROR                          !
!         ! '4' ERROR                            !
!         !
! CAT-ER ! STORAGE OF ERROR ON CURRENT CATEGORY   !
!         ! ' ' NO ERROR                        !
!         ! 'E' ERROR                          !
!         !
!ER-scrn-! MEMORIZATION OF DATA ELEMENT ERROR      !
! delcod ! '0' DATA ELEMENT ABSENT              !
!         ! '1' DATA ELEMENT PRESENT        !
!         ! '2' INVALID ABSENCE              !
!         ! '4' INVALID CLASS                !
!         ! '5' INVALID VALUE                !
!         !
+-----+
```