

# **HELP-WINDOWS™**

**LEARNING HELP-WINDOWS**

**Release 3.0**

**WORLD HEADQUARTERS**

**UNICOM Systems Inc.**  
1032 Cove Way  
Beverly Hills, CA 90210  
(818) 838-0326

**UNICOM Systems, Inc**

**UNICOM PLAZA, Suite 310**  
15535 San Fernando Mission Blvd.  
Mission Hills, CA 91345  
(818) 838-0606, Fax: (818) 838-0776

**FAR EAST DIVISION**

**UNICOM Systems Int'l Korea, Ltd.**  
Hawangshimni Dong 301 Suite 201  
Songdong-ku Seoul, Korea  
(02) 296-5476, Fax: (02) 294-5446



Unicom Systems, Inc.  
15535 San Fernando Mission Blvd.  
Mission Hills, California 91345

*First Edition – March 1, 1999*

*The information in this document applies to Release 3.0  
Of the Help-Windows™ product for all operating environments.*

*Published by UNICOM SYSTEMS Inc.  
15535 San Fernando Mission Blvd. Suite 310  
Mission Hills, California 91345*

*© Copyright UNICOM Systems, Inc.*

*All rights reserved*

# HELP-WINDOWS

## LEARNING HELP-WINDOWS

### Release 3.0

#### TABLE OF CONTENTS

<b>TABLE OF CONTENTS</b> .....	<b>0</b>
<b>PREFACE</b> .....	<b>1</b>
<b>GETTING STARTED WITH HELP-WINDOWS</b> .....	<b>2</b>
TYPES OF HELP TEXT .....	2
THE TEXT MAINTENANCE TRANSACTION .....	4
CREATING A STRUCTURED TEXT DOCUMENT .....	5
CREATING A HELP WINDOW WITH STRUCTURED TEXT .....	25
CREATING A HELP WINDOW USING THE FAST-PATH METHOD .....	48
SUMMARY OF THE GETTING-STARTED EXERCISES .....	58
<b>HELP ACCESS MODE</b> .....	<b>58</b>
THE EXIT PROCESSOR .....	59
ACTIVATION OF HELP ACCESS MODE .....	59
RETRIEVING HELP DISPLAYS .....	63
DIRECTIONAL OPTIONS AT A HELP DISPLAY .....	65
ENTERING COMMANDS AT A HELP DISPLAY .....	66
RETRIEVING A SCREEN-LEVEL HELP DISPLAY .....	68
RETRIEVING A FIELD-LEVEL HELP DISPLAY .....	69
CHAINED HELP WINDOWS .....	70
HELP DISPLAYS WITHIN HELP TEXT (NESTED HELP) .....	72
EXITING FROM A HELP DISPLAY .....	74
VIEWING DOCUMENTS WITHOUT USING HWIN .....	74
USING THE CUT/PASTE FEATURE .....	77
DEACTIVATING HELP ACCESS MODE AT A TERMINAL .....	78
DISPLAYING THE CURRENT TERMINAL STATUS .....	78
<b>PLANNING FOR THE USE OF HELP-WINDOWS</b> .....	<b>82</b>
TYPICAL SCENARIOS IN DOCUMENTATION PLANNING .....	82
SUMMARY OF TYPICAL SCENARIOS .....	96
CUSTOMIZATION OPTIONS .....	97
REFERENCE GUIDE TO PRODUCT FEATURES .....	98
<b>INDEX</b> .....	<b>100</b>

# **PREFACE**

This preface is designed to provide a quick overview of the contents of LEARNING HELP-WINDOWS. Each major section of the manual is briefly described, along with the starting page number for that section.

## **GETTING STARTED WITH HELP WINDOWS (Section L1)**

**Page 3**

This section is designed to help the new user very quickly produce results with the HELP-WINDOWS product. It makes no pretense of covering all of the subject matter. A series of exercises are presented which the new user can follow exactly as described to successfully define four help displays. A demonstration transaction, provided with the product is used for each exercise.

## **HELP ACCESS MODE - USING ON-LINE HELP DISPLAYS (Section L2)**

**Page 47**

This section of the Learning HELP-WINDOWS manual deals with the use of HELP-WINDOWS by the .i.end user:use of HELP-WINDOWS;. That is, the retrieval of help displays while in a user transaction. This is called Help Access Mode, or simply '.i.help mode:description of;'. This section should be used by the text-writer for a deeper understanding of the kinds of specifications that can be made when defining help displays that will produce the various results discussed herein.

## **PLANNING FOR THE USE OF HELP-WINDOWS (Section L3)**

**Page 65**

This section is designed to provide helpful insight into the preparation for developing an effective and manageable on-line documentation and/or help system using HELP-WINDOWS. In addition, we will attempt to guide you in the planning stage, where you decide what sort of documentation system to implement, who will do the work, and what subjects of the reference manuals must be understood.

## **INDEX Page 77**

The index arranges key words and phrases of the reference guide in alphabetical order and provides the page number reference to their location.

## GETTING STARTED WITH HELP-WINDOWS

This section is designed to help the new user very quickly produce results with the HELP-WINDOWS product. It makes no pretense of covering all of the subject matter. HELP-WINDOWS is a large and comprehensive package, with many options in the ways that it can be used. For the serious user, therefore, it is critically important that most or all of the accompanying user's guide be read.

This section will lead the new user through three exercises of the product. These are:

- 1) Creating a fully structured on-line document.
- 2) Creating a help window which references a portion of the text in the on-line document. For this exercise we will use what is known as the "structured" method.
- 3) Creating a help window which displays text that is not part of a larger document. For this exercise we will use what is known as the "fast-path" method.

The procedures for these exercises are described in tutorial form, with appropriate illustrations which the user can simulate on a terminal. As each subject is discussed, footnote references will direct you to the topic in the User's Guide where a more complete explanation of each function or feature can be found.

## TYPES OF HELP TEXT

Help text, which is the words that you will write for your on-line help displays, can take two different forms in HELP-WINDOWS. The type of help text that you use is dependent on the method used to create it.

### STRUCTURED TEXT METHOD

In this case, the text to be used for a help display must be part of what is called a "structured hierarchy". There must be a SYSTEM record, a DOCUMENT record and at least a SECTION record defined. The SUBJECT record is optional. Then there must be one or more TEXT records which contain the actual text to be displayed in the help screen or window. `.i.structured help text:description of; .i.text:structured:description of;`

The SYSTEM can be thought of as a library of related books (payroll, general ledger, etc.). The DOCUMENT can be compared to a book in the library. The SECTION is like a chapter in a book and the SUBJECT can be compared to a topic within a chapter. `.i.hierarchy:description of;`

There can be many documents in a system, many sections in a document and many subjects in a section.

The text records fit at the next level, after the subject (or section, if subject is omitted).

The structured text method is what you must use if you want to create full documentation manuals. It is also the recommended method to use for on-line help displays, although there is another method, as described next.

### FAST-PATH METHOD

- 2) **The "fast-path" method.** This method allows you to create up to one full page (screen display) of text for use only as a help display. The text can be keyed directly in a window or a full-screen. With this method, the text does not need to have a full hierarchical structure. That is, it does not need to be part of a SYSTEM, DOCUMENT, SECTION or SUBJECT.

The "fast-path" method is intended for creating text where its only purpose is to be used as an on-line help display; that is, it is not part of a larger on-line document.

## CONSIDERATIONS IN METHOD SELECTION

While the fast path method is quicker and requires less control records than the structured text approach, there are some limitations, as follows: .i.fast-path help text:limitations of;.i.text:fast-path:limitations of;

- Only one text record (20 lines of text) can be created per help definition. If more text is needed there is a method called "chaining" which allows you to connect two or more fast-path definitions together.
- Fast-path help text cannot be printed or manipulated with the batch processing programs.
- Fast-path help text cannot be referenced (included) from other text.
- You cannot create a table of contents or index for fast-path text.

The structured text approach requires more control records to accomplish a help display, but it offers the following advantages: .i.structured help text:advantages of;.i.text:structured:advantages of;

- There is no limit to the amount of text that can be part of the help display.
- The help display may be an extract of a larger document, thereby eliminating the need to define text in two places (both the help display and a procedure manual).
- The text can be printed and manipulated by the batch processing programs.
- You can have several different help displays all pointing to the same text, if desired.
- You can create modules of structured text that can be "included" in other text members.
- You can create a table of contents in structured text, which makes locating sections of text easier.
- Both the table of contents and index can be printed with the document.

## THE TEXT MAINTENANCE TRANSACTION

All documents, whether full manuals or stand-alone help displays are created and maintained using the maintenance transaction, HWIN<sup>1</sup>. The HWIN transaction can be operated from a menu of functions or from a clear screen using "direct-access" commands. Integral to the operation of HWIN is the maintenance "conversation", which always remembers your position in the document and returns you there with a minimum of keystrokes<sup>2</sup>.

HWIN also provides "directories" of each of the various record types which can be used to view a document in outline form or quickly move directly to desired text.<sup>3</sup>

You can also insert special text records in your document, using HWIN, which will create a table of contents and/or an index for the document. The table of contents can be displayed on-line and used to position the operator directly to the desired text.

## CREATING A STRUCTURED TEXT DOCUMENT

Documents must be organized in a four or five-level hierarchical structure. Each level of the hierarchy is identified by a control record which names that level and provides a brief description for use in the directory display and print. The four levels are:

- 1) The SYSTEM definition. There may be any number of systems in the file.
- 2) The DOCUMENT definition. There may be many documents in a system.
- 3) The SECTION definition. There may be many sections in each document.
- 4) The SUBJECT definition. There may be many subjects in each section. Subject is optional; only the first three levels are required. You should avoid subject definitions for short documents and use them for long documents.
- 5) Finally, the actual documentation text is the lowest level in the hierarchy, being connected to the subject or section definition.<sup>4</sup>

### ADDING THE HIERARCHY RECORDS

We will now begin to add the hierarchy records. To begin, you can enter just the transaction code, HWIN, from a clear screen and press Enter. This will display the maintenance selection menu, which appears as follows:  
*.i.hierarchy:creation of; .i.screen displays:maintenance selection menu;*

**Help-Windows Release 3.0.xxxxxx On-line Documentation System**  
**Maintenance Selection Menu**

Record Type	Add	Delete	Inquire Update Directory	Table Of Conts	Active Key
System Header	1	8	15	22	_____
Document Header	2	9	16		_____
Section Header	3	10	17		_____
Subject Header	4	11	18		_____
Text Record	5	12	19		000000
Transaction Key	6	13	20		_____
Cross-Document Field	7	14	21		_____
23 Search Text _____					
24 User Profile					
25 Quick Directory					
26 Mass Copy/Delete					

**Modify active key fields if needed and make menu selection.**

**Enter Menu Selection ==> \_\_**  
**Or Press PF1 for HELP or PF3 to QUIT**

The ACTIVE KEY fields contain the currently active names of each of the record types. When you enter the menu for the first time, these fields will all be blank. After you have created one or more records, or any time you access a record at any level in the hierarchy, the identifiers of all of the records in that hierarchy are saved in the active key fields.

<sup>4</sup>See *ORGANIZATION OF YOUR MANUAL* in section G2.



This means that you can leave HELP-WINDOWS at any time, and when you return to the maintenance selection menu, the identifiers of the last record that you accessed (even if CICS has been down and back up) will be provided for you. This is referred-to as the maintenance 'conversation', in HELP-WINDOWS<sup>5</sup>.

If the current active key fields do not contain the identifiers of the record that you wish to access, you can enter new identifiers in any or all of the fields. If there are no active key fields and you select a menu option such as INQUIRE, which requires key fields, you will be prompted to enter them.

You respond to the menu by ensuring that the active key fields contain the desired identifiers (if any), then entering the one or two-digit number which corresponds to the desired function following the selection pointer (==>). Then press ENTER.

### STEP 1: ADDING THE SYSTEM RECORD

From the Maintenance Selection Menu, enter number 01 (to Add a System Header), and press ENTER; or, alternatively, from a clear screen enter

HWIN,A,SY

and press ENTER. 'A' is the command for Add, and SY is the minimum abbreviation for SYSTEM. You could enter any number of the characters of the word, up to the complete word, SYSTEM.

The next screen to display, for either method of entry, is the SYSTEM addition screen:

HWIN,A,SY

**Help-Windows On-Line Documentation/Help System  
System Record Maintenance**

System ID \_\_\_\_\_

System Description  
\_\_\_\_\_

System Page Heading  
\_\_\_\_\_

Enter ">" in position one to center heading.

[ PF Key prompts for directional options. ]

HW10008. Enter data and press "ENTER" to add system or "PF3" to exit

Now, key a one to eight-character alphanumeric name in the SYSTEM ID field. For our exercise, we're going to use the name TESTSYS.

<sup>5</sup>See *FINDING YOUR PLACE IN THE TEXT FILE , THE MAINTENANCE CONVERSATION* in section G7.

The SYSTEM DESCRIPTION field is not required, but we recommend that you always supply descriptions, as it makes it much easier to find things later.

The SYSTEM PAGE HEADING is not required and won't be used for our exercise. This is the running header that you want to print on every page of this system when the batch print program is used.

Fill in all desired fields and press ENTER. This will write the system definition to the file and automatically produce the DOCUMENT addition screen<sup>6</sup>.

## STEP 2: ADDING THE DOCUMENT RECORD

The document addition screen will automatically appear following the addition of a system record. It can be directly produced by selecting number 2 from the menu or entering

HWIN,A,SY=TESTSYS,DO

from a clear screen. In this example, the presence of 'DO', which is the minimum abbreviation for DOCUMENT, along with a supplied system name, tells HWIN that document addition is what is desired. The document addition screen appears as follows:

```
HWIN,A,SY=TESTSYS,DO

      Help-Windows On-Line Documentation/Help System
      Document Record Maintenance

System ID   TESTSYS
Document ID  _____

Document Description
_____

Document Heading
_____

Enter ">" in position one to center heading.

[ PF Key prompts for directional options ]
HW10008. Enter data and press "ENTER" to add document or "PF3" to exit
```

The SYSTEM ID already contains the name TESTSYS.

The DOCUMENT DESCRIPTION field is not required, but we recommend that you always supply descriptions, as it makes it much easier to find things later.

---

<sup>6</sup>See *RECORD MAINTENANCE BY RECORD TYPE, THE SYSTEM RECORD, ADDING A NEW SYSTEM RECORD* in section G2.

The DOCUMENT PAGE HEADING is not required and won't be used for our exercise. This is the running header that you want to print on every page of this system when the batch print program is used.

Now, fill in the DOCUMENT ID field (we'll use TESTDOC) and the DOCUMENT DESCRIPTION field, then press ENTER. This will automatically produce the SECTION addition screen<sup>7</sup>.

---

<sup>7</sup>See *RECORD MAINTENANCE BY RECORD TYPE, THE DOCUMENT RECORD, ADDING A NEW DOCUMENT RECORD* in section G2.

### STEP 3: ADDING THE SECTION RECORD

The section addition screen will automatically appear following the addition of a document record. It can be directly produced by selecting number 3 from the menu or entering

HWIN,A,SY=TESTSYS,DO=TESTDOC,SE

from a clear screen. In this example, the presence of 'SE', which is the minimum abbreviation for SECTION, along with a supplied system and document name, tells HWIN that section addition is what is desired.

The section addition screen appears as follows:

```
HWIN,A,SY=TESTSYS,DO=TESTDOC,SE

      Help-Windows On-Line Documentation/Help System
      Section Record Maintenance

System ID      TESTSYS
Document ID    TESTDOC
Section Number 001

Section Description
_____

Section Heading
_____

Enter ">" in position one to center heading.

      [ PF Key prompts for directional options ]
HW10008. Enter data and press "ENTER" to add section or "PF3" to exit
```

The SYSTEM ID and DOCUMENT ID are already filled-in with TESTSYS and TESTDOC.

Section 001 is already filled-in, since sections are normally defined as incrementing numbers. You can, however, enter an alphanumeric section identifier if desired.

Fill in the section description and press Enter. This will automatically produce the SUBJECT addition screen<sup>8</sup>.

### STEP 4: ADDING THE SUBJECT RECORD

The subject addition screen will automatically appear following the addition of a section record. It can be directly produced by selecting number 4 from the menu or entering

HWIN,A,SY=TESTSYS,DO=TESTDOC,SE=001,SU

<sup>8</sup>See *RECORD MAINTENANCE BY RECORD TYPE, THE SECTION RECORD, ADDING A NEW SECTION RECORD* in section G2.

from a clear screen. In this example, the presence of 'SU', which is the minimum abbreviation for SUBJECT, along with a supplied system, document and section identifier, tells HWIN that subject addition is what is desired.

The subject addition screen appears as follows:

```
HWIN,A,SY=TESTSYS,DO=TESTDOC,SE=001,SU
```

**Help-Windows On-Line Documentation/Help System**  
**Subject Record Maintenance**  
**(OPTIONAL - not required)**

```
System ID      TESTSYS
Document ID    TESTDOC
Section Number 001
Subject        _____
```

```
Subject Description
_____
```

```
Subject Heading
_____
```

Enter ">" in position one to center heading.

[ PF Key prompts for directional options ]

HW10008. Enter data and press "ENTER" to add subject or "PF3" to exit

For our exercise we're going to bypass the subject record by pressing ENTER without keying a SUBJECT ID. The subject record is not necessary and should only be used for larger documents. If you want to use a subject, key a subject ID, a description, and press ENTER. This will automatically produce the TEXT addition screen<sup>9</sup>.

## ENTERING DOCUMENTATION TEXT

We are now at the point where we are ready to use the text editor to create the actual wording of the document that we are writing. Remember that with the structured text method, it does not matter at this point whether we are creating a help display or a manual to be printed and/or viewed on-line. For the purpose of this demonstration, we will be creating a help display.

The text that we will enter will belong to the structured hierarchy known as system TESTSYS, document TESTDOC and section 001.

### STEP 5: ADDING TEXT RECORDS

The full-screen editor for adding text will automatically appear after pressing ENTER for the subject record. It can be directly produced by selecting number 5 from the menu or entering

```
HWIN,A,SY=TESTSYS,DO=TESTDOC,SE=001,REC
```

from a clear screen. In this example, the presence of 'REC', which is the minimum abbreviation for RECORD, along with a supplied system, document and section identifier, tells HWIN that text record addition is what is desired.

---

<sup>9</sup>See *RECORD MAINTENANCE BY RECORD TYPE, THE SUBJECT RECORD, ADDING A NEW SUBJECT RECORD* in section G2.



If no changes are needed and you wish to add another text record, press the ADD key or change the command coded at the top of the screen to 'A' and press ENTER. This will produce the next TEXT entry screen, with the record number assigned and displayed at the top of the screen.



## MANAGING TEXT RECORDS

Once you have completed addition of more than one text record, you can use the NEXT and PREVIOUS commands to browse forward and backward through the text records. Both NEXT and PREVIOUS appear as one of the PF key prompts at the bottom of the screen. To move forward, press the NEXT key; to move backward, press the PREV key. Of course, you can also enter the 'N' or 'P' command on the command line.

Two other methods are available to move around in the text records:

- 1) You can tab the cursor to the point in the command line following REC=, enter a different text record number and press ENTER. This will take you directly to that text record.
- 2) You can press the PF key for the QUIT command (usually PF3) and return to the maintenance selection menu. There you can change the currently displayed text record number in the active key fields to the desired number and select menu function 12.

## MAKING CHANGES

If changes are needed after adding a text record, you can simply move the cursor to the area to be changed and type over any existing data. You can use the DEL key to delete characters in a line. To insert characters on a line, position the cursor where the insertion is to be made, press the INS key and type the desired characters. The ERASE EOF key may be used to erase all or part of a line at any time.

## LINE DELETION

To delete an entire line, TAB to the COMMAND FIELD at the end of the line (where the three asterisks are) and enter a 'D' over any of the asterisks, then press ENTER. The entire line will be deleted and all following lines will shift up one line.

To delete more than one line at a time, enter a 'D' in more than one COMMAND FIELD before pressing ENTER or enter 'Dnn' on one or more COMMAND FIELDS, where nn is the number of consecutive lines (including the current line) to be deleted. You can also enter 'DD' on the first line to be deleted and another 'DD' on the last line, then press ENTER to delete all text from the first 'DD' through the second.

## LINE INSERTION

To insert a line between two text lines, Tab to the COMMAND FIELD of the preceding line and enter an 'I' over any one of the asterisks, then press ENTER. This will cause a blank line to be inserted following that line, and all subsequent lines will shift down one line. You may then key any desired text on that line.

To insert multiple lines in one spot, enter 'Inn' in the preceding COMMAND FIELD, where nn is the number of lines to insert, then press ENTER. This will cause nn blank lines to be inserted after the current line, and all subsequent lines will shift down nn lines.

Note that when inserting lines, if the number of insertions causes any lines containing text to shift beyond the 20th (or last) line of the screen, that text will not roll off of the screen. The number of blank lines inserted will be the maximum number until the last line of text reaches the bottom line. This is because text records are maintained in screen-image form (one record per screen display), and text will not automatically be rolled forward into the next record. Additional space in a

TEXT record can be obtained by use of the MOVE/COPY commands or by the SPLIT command prior to making the insertion.

## MOVING AND COPYING TEXT LINES

One or more consecutive text lines can be moved or copied to another location. To move text lines, enter 'M' or 'Mnn' (nn = number of lines to move) in a COMMAND FIELD and press ENTER. This causes the specified line(s) to be deleted from that location. Now place an 'A' (insert-after) in the COMMAND FIELD on the line preceding the new location, which may be on the same or a different TEXT record, and press ENTER. The MOVED lines will be inserted following that line. You can also place a 'B' (insert-before) on the line following the new location and press ENTER. This will insert the text immediately preceding this line.

To copy text lines, enter 'C' or 'Cnn' (nn = number of lines to copy) in a COMMAND FIELD and press ENTER. Now place an 'A' (insert-after) or 'B' (insert-before) in the COMMAND FIELD on the line preceding or following the new location, which may be on the same or a different TEXT record, and press ENTER. The COPIED lines will be inserted at that location.

You may also perform block moves and copies by placing 'MM' or 'CC' on the first line to be moved or copied and another 'MM' or 'CC' on the last line to be moved/copied.

You may enter more than one 'M' or 'C' command on one edit screen, if desired. The result is that all selected lines are grouped together consecutively and inserted as one set wherever the 'A' or 'B' command is placed. For instance, if 'C2' is entered in the command area at line 3, then 'C4' is entered in the command area at line 16, all six text lines will be grouped together and inserted as one set following the line where the 'A' command is placed.

## ADDITIONAL EDITOR OPERATIONS

There are several other operations that can be done in edit mode<sup>11</sup>. Among them are:

- 1) Line spacing and page-break control.
- 2) Using different display attributes and colors.
- 3) Splitting a text record.
- 4) Using the INCLUDE statement to insert blocks of text from another hierarchy.
- 5) Controlling the display when using word-wrap.
- 6) Setting upper and lower case options.
- 7) Inserting table of contents and index entries.
- 8) Defining a user exit for programmed text displays.

Existing text in another form can be imported into the HELP-WINDOWS text file by means of a batch program. Thus, if you already have a lot of documentation created on some other medium or system, you will not need to re-key it to use it in HELP-WINDOWS<sup>12</sup>.

### STEP 6: KEYING THE FIRST PAGE OF TEXT

For this exercise, we need to key at least two screens of text data. You may not want to do all of the keying that is illustrated here, and it is not necessary in order to gain an understanding of the

---

<sup>11</sup>See *RECORD MAINTENANCE BY RECORD TYPE, THE TEXT RECORD, ADDING TEXT RECORDS, USING THE TEXT EDITOR* in section G2.

<sup>12</sup>See *REFORMATTING EXISTING TEXT* in section G8.

functions involved. You may key several lines of nonsense information, if desired, but you should enter at least two text records.

The transaction screen that we will be documenting in the next exercise is the HDMO transaction, which is supplied with HELP-WINDOWS. This transaction simply displays a generic application screen that can be used for practice. The text we shall key, therefore, will apply to the HDMO screen. That way, we can use this text in the next exercise, which involves creating a help window using structured text.

**[Note].** The text illustrated below has already been keyed and is present in the text file under another system name. You should key a couple of practice text records to get the feel of the text editor, then, if you want to use the pre-keyed text to complete the next exercise, proceed to Step 9, *COPYING THE PRE-KEYED TEXT RECORDS*.

We will key the following text in the text addition screen, as follows:

```

HWIN,E,SY=TESTSYS,DO=TESTDOC,SE=001,REC=01000
<STRTFD HDMO TRAN HELP>                                     ***
The HDMO transaction is a sample CICS application that produces a screen ***
display similar to many industry applications.               ***
                                                             ***
The screen is divided into two parts:                         ***
  The upper half is used for entry of name and address information for ***
  each customer. The fields in this area are:                 ***
<STRTFD HDMO FIELD HELP1>                                    ***
CUSTOMER NUMBER -      Enter a 5-digit number to be assigned to this ***
                        customer. It must be a unique number.  ***
CUSTOMER NAME -        The full customer or company name.     ***
ADDRESS LINES 1&2 -    The street and mailing address.        ***
CITY, STATE, ZIP -     Full city name, state code, zip code.   ***
<STOPFLD HDMO FIELD HELP1>                                    ***
<SPACE 1>                                                      ***
  The lower half of the screen contains the product information for this ***
  customer. Key the following information for each product:   ***
<STRTFD HDMO FIELD HELP2>                                    ***
PRODUCT CODE -        Enter the 4-digit code that identifies this product. ***
DATE SOLD -           The date (MMDDYY) of the sale.          ***
[ PF Key prompts for directional options ]

```

To use structured text in a help display, you must identify the text to be used by assigning it a name. This is accomplished by inserting special text commands before and after the text to be used. These commands are known as STRTFD and STOPFLD statements. In this example, we have identified three portions of the text to be used as help text:

- 1) The STRTFD statement on line 1 identifies all text that follows with the name 'HDMO TRAN HELP'. The STOPFLD for this text will be entered on the next page.
- 2) The STRTFD statement on line 8 identifies the text that follows with the name 'HDMO FIELD HELP1'. The STOPFLD for this text is on line 14.
- 3) The STRTFD statement on line 18 identifies the text that follows with the name 'HDMO FIELD HELP2'. The STOPFLD for this text will be entered on the next page<sup>13</sup>.

Upon filling the first screen, we press ENTER to update this text record. The record will be written to the file, then re-displayed. We could now make any desired changes to the text.

Since we have no changes to be made, but there is more text to add, we now press the AC6 key to add another record. We could also change the 'E' command to 'A' and press ENTER, if desired. This will cause another empty edit screen to appear, this one with record number 002000.

<sup>13</sup>See *THE AUTO-DEFINE PROCESS - STRUCTURED TEXT METHOD, IDENTIFYING THE TEXT TO BE DISPLAYED*, in section G5.





We now have room to key two additional lines of text at that location, as follows:

```
HWIN,E,SY=TESTSYS,DO=TESTDOC,SE=001,REC=02000
ORIGINAL COST -          This is the sales price of the product.      ***
VENDOR NUMBER -        The number assigned to the vendor for this    ***
                        product.                                       ***
MAINTENANCE AMOUNT -   If the customer purchased a maintenance      ***
                        agreement, this is the amount of the annual    ***
                        maintenance fee.                               ***
EXPIRATION DATE -      This is the date that the maintenance agree-  ***
                        ment will expire.                              ***
<STOPFLD HDMO FIELD HELP2>                                         ***
Upon completion of entry for all fields, press the ENTER key to add this ***
record to the file.                                                 ***
Your input will be edited for field validity and valid customer/vendor ***
numbers. If errors exist, an error message will display for correction. ***
If there are no errors, the screen should re-display with the message ***
"RECORD ADDED SUCCESSFULLY"                                         ***
on the bottom row.                                                 ***
<STOPFLD HDMO TRAN HELP>                                           ***
[ PF Key prompts for directional options ]                          ***
```

Upon pressing ENTER, the changes are recorded in the text record and our text addition is complete.

## STEP 8: DISPLAYING TEXT IN LIST FORM

It is usually a good idea after keying some text records to display the text as it will be seen by the end-user. To do this, press the LIST key (usually PF5) or change the command from 'E' to 'L' and press ENTER. Record 001000 appears as:

HWIN,L,SY=TESTSYS,DO=TESTDOC,SE=001,REC=01000

The HDMO transaction is a sample CICS application that produces a screen display similar to many industry applications.

The screen is divided into two parts:

The upper half is used for entry of name and address information for each customer. The fields in this area are:

CUSTOMER NUMBER -	Enter a 5-digit number to be assigned to this customer. It must be a unique number.
CUSTOMER NAME -	The full customer or company name.
ADDRESS LINES 1&2 -	The street and mailing address.
CITY, STATE, ZIP -	Full city name, state code, zip code.

The lower half of the screen contains the product information for this customer. Key the following information for each product:

PRODUCT CODE -	Enter the 4-digit code that identifies this product.
DATE SOLD -	The date (MMDDYY) of the sale.
ORIGINAL COST -	This is the sales price of the product.
VENDOR NUMBER -	The number assigned to the vendor for this product.
MAINTENANCE AMOUNT -	If the customer purchased a maintenance

You will note that in the list mode display, the two text records flow together and it all appears as contiguous text. The STRTFLD, STOPFLD and SPACE commands do not appear, the command fields disappear, and all text is protected from modification.

After viewing the text in list mode, if there are corrections to be made, simply press the EDIT key (or change the command to 'E') and press ENTER to return to edit mode, where the changes can be made<sup>14</sup>.

## STEP 9: COPYING THE PRE-KEYED TEXT RECORDS

The two text records illustrated in steps 6 and 7, above, have been pre-keyed for you and are present in the text file under a different system name. If you keyed the text records exactly as illustrated in steps 6 and 7 you can skip this step, but it is recommended that you do this, as it will ensure that the text in use agrees with the instructions herein. This step will also give you some practice using the DELETE and COPY commands.

The pre-keyed text exists in system DEMOSYS. The document name is the same as used above - TESTDOC. It contains one section record (number 001) and two text records, numbered 1000 and 2000, which appear exactly as illustrated in steps 6 and 7.

---

<sup>14</sup>See *RECORD MAINTENANCE BY RECORD TYPE, THE TEXT RECORD, DISPLAYING TEXT RECORDS IN LIST MODE* in section G2.



We can easily copy the two text records into system TESTSYS by replacing the entire subject 001 in TESTSYS, TESTDOC with section 001 in DEMOSYS, TESTDOC. To do this, we must first do a global deletion of section 001 in TESTSYS, TESTDOC. Global deletion means that the hierarchy record and all subordinate records are deleted with one command<sup>15</sup>.

---

<sup>15</sup>You could also copy each text record individually, but copying the entire section is easier.

## DELETING THE TEST SECTION AND TEXT RECORDS

First, return to the maintenance selection menu by pressing the QUIT key (PF3), or, if you are currently at a blank screen, enter HWIN and press ENTER.

Make sure that the active key fields of the selection menu contain TESTSYS for the system, TESTDOC for the document and 001 for the section. Now enter '10' as the menu selection number and press ENTER. This will display the section record for system TESTSYS, document TESTDOC.

Now press the DELETE Pf key, or change the command from 'I' to 'D' and press ENTER. You should now receive the following message at the bottom of the screen:

*HW10012. THIS SECTION HAS SUBORDINATE RECORDS ASSOCIATED WITH IT. PRESS "PF5" TO GLOBALLY DELETE THE ENTIRE SECTION OR "PF3" TO END DELETE.*

This message is telling you that text records exist for this section. It is referring to the two practice records that you keyed earlier<sup>16</sup>.

Now, press PF5, which will delete the section record and all text records connected to this section<sup>17</sup>. Upon completion, you should receive the message

*HW10022. GLOBAL SECTION DELETION HAS BEEN COMPLETED.*

## COPYING THE DEMO SECTION AND TEXT RECORDS

Now we need to display the section record for system DEMOSYS. The easiest way to do this at this point is to press the TAB key, which skips the cursor to the beginning of the system name, TESTSYS. Now overwrite the 'TEST' with 'DEMO', changing the system name to DEMOSYS. Now press ENTER<sup>18</sup>.

The section 001 record for system DEMOSYS, document TESTDOC should now display. Now change the 'I' command to 'C' (COPY) and press ENTER<sup>19</sup>. The following message will appear at the bottom of the screen:

*HW10017. PRESS "ENTER" TO COPY THIS SECTION OR "PF3" TO EXIT COPY MODE.*

Press ENTER to begin the copy function. You now receive the message

*HW10018. ENTER NEW SECTION INFORMATION AND PRESS ENTER TO COMPLETE COPY.*

---

<sup>16</sup>If you did not key any text records, the message will read *HW14007. PRESS "ENTER" TO DELETE SECTION OR "PF3" TO EXIT DELETE MODE*).

<sup>17</sup>See *LOCAL AND GLOBAL DELETION*, in section G2. Also see the DELETE command in section G6.

<sup>18</sup>You could also return to the menu, change the system name to DEMOSYS and press ENTER.

<sup>19</sup>For a discussion of the command line format, see *MAINTENANCE COMMANDS IN SECTION G6*.

Overtyping the system name (currently DEMOSYS) with TESTSYS. Do not change any other fields, just press ENTER. This will cause the section 001 record and both text records to be copied from system DEMOSYS and attached to system TESTSYS, document TESTDOC. The message

*HW10020. SECTION HAS BEEN COPIED.*

will display when the copy is complete<sup>20</sup>. If you would like to view the copied text records, press PF5 (SUB). PF5 will proceed down the hierarchy to the subject record, and since there is no subject record, the first text record will display<sup>21</sup>.

You have now successfully copied the pre-keyed text records, which appear exactly as described in steps 6 and 7. These can now be used in the next exercise, *CREATING A HELP WINDOW WITH STRUCTURED TEXT*.

## SUMMARY OF THE DOCUMENT BUILDING EXAMPLE

We have now completed entry for a complete document with two screens full of text. There is no limit to the amount of text that can be associated with a section or subject. Likewise, there is no limit to the number of subjects in a section, sections within a document, or documents within a system.

There are several things that could be done with this document now that we have created it. We won't examine all of these in this overview, but the following subjects are suggested reading:

- 1) Adding color and extended highlighting to your help text. (See *THE TEXT RECORD, CREATING COLOR AND HIGHLIGHTING*, section G2).
- 2) Insert table of contents entries into the text. (See *CREATING AND USING A TABLE OF CONTENTS* in section G7).
- 3) Insert index entries into the text. (See *CREATING AND USING AN INDEX* in section G7).
- 4) Viewing the directory display to find text records. (See *USING DIRECTORIES*, section G3).
- 5) Printing the document. (See *PRINTING DOCUMENTATION* in *BATCH PROCESSING*, section G8).
- 6) Load additional text into this document. (See *BATCH PROCESSING, REFORMATTING EXISTING TEXT*, section G8).
- 7) Unload text to a sequential file for processing on another system. (See *BATCH PROCESSING, EXPORTING THE TEXT FILE*, section G8).
- 8) Set up dummy transactions for full-screen help text. (See *DEFINING A HELP MENU, USING DUMMY TRANSACTIONS*, section G5).

---

<sup>20</sup>For more information on COPY, see section G6, *COMMANDS AND KEYWORDS IN HELP-WINDOWS*.

<sup>21</sup>Proceeding through the hierarchy is explained more fully in the various *DIRECTIONAL OPTIONS* in section G2.

## CREATING A HELP WINDOW WITH STRUCTURED TEXT

As described earlier, "structured text" is text that is created by the method used in the first exercise, where it belongs to a full hierarchy containing a system, document, section and optional subject. This is opposed to "fast-path" text, which is created by the method used in the next exercise, where the text is attached directly to a transaction key record<sup>22</sup>, and no structured hierarchy exists.

In this exercise, we will use the structured text which was created in the first exercise and associate it with a user transaction screen.

### ESTABLISHING THE "HOT" KEYS

The first step in using auto-define is to establish PF or PA keys to be used as the help definition key and help access keys. You need two keys when you are creating help. End-users will only need one key, the help access key.

### STEP 1: DEFINING THE "HELP DEFINITION" KEY

To enter auto-define mode, key the following command from a clear screen:

HWIX,DEFINE

and press ENTER. Or, if desired, enter only HWIX, which produces a screen display with all of the available commands to the HWIX transaction. Then enter DEFINE in the command area and press ENTER. The following screen will display in response to the command:

**HELP-WINDOWS Release 3.0.xxxxxx**  
**On-Line Documentation/Help System**  
**Activation of On-line Help Definition Mode**

To activate Help-Windows automatic definition mode, press the PF or PA key to be used as the **Help Definition** key.

To perform an automatic help definition, display the user transaction screen for which help is to be defined, position the cursor to the field to be documented, and press the help definition key.

A window will appear on the screen with further instructions for creating the definition.

An understanding of **Transaction Keys**, **Cross-Document Fields** and **Text Entry** required to use this feature. Please refer to the reference manual for detailed instructions.

**The Help Definition key is ==>**

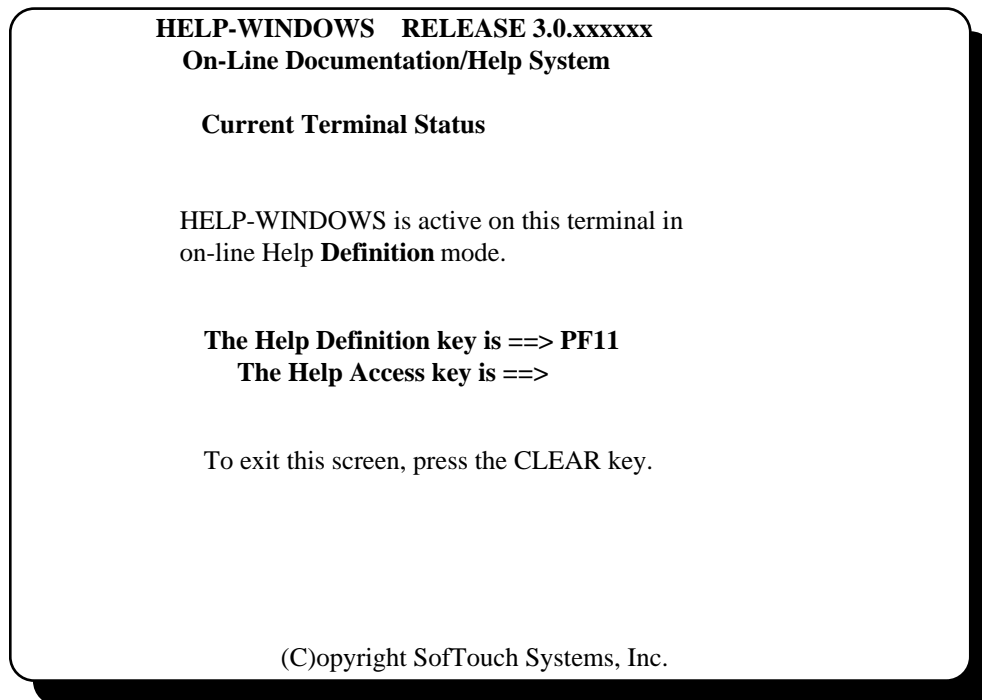
(C)opyright SofTouch Systems, Inc. 1988

Press the PF key or PA key to be used as the Help Definition key. Any of the available function keys, PF1 through PF24, PA1, PA2 or PA3 can be used. Be sure you choose a function key that

<sup>22</sup>Help definitions are stored in a special record called a 'transaction key record'.

is not needed by the application program that you are preparing to document. For our exercise, let's use PF11 as the help definition key.

When you press the desired function key, the following screen will display:



You have now successfully entered Help Definition mode and established the designated PF or PA key as your "hot" key for help definition<sup>23</sup>.

## STEP 2: DEFINING THE HELP ACCESS KEY

In addition to the help definition key, you should also define the help access key. This is the key to press to retrieve a completed help display. When creating a help display, you will want to test the final result as the end user will see it. The help access key is used for this purpose.

Establishing the help access key is done using the same procedure as the help definition key. The only difference is the command to be used. To define a help access key, enter the command

HWIX,ON

from a clear screen. Or, if desired, enter only HWIX, which produces a screen display with all of the available commands to the HWIX transaction. Then enter ON in the command area and press ENTER.

<sup>23</sup>See *ENTERING AUTO-DEFINE MODE* in section G5.

This will produce the following display:

**HELP-WINDOWS Release 3.0.xxxxxx  
On-Line Documentation/Help System**

**Activation of On-line Help Access Mode**

To activate Help-Windows for your terminal, press the PF or PA key to be used as the default help key.

If a particular key has already been associated with the transaction(s) to be used, you may not need a default help key, in which case you can press CLEAR.

If you have both a default and a transaction-level help key, the transaction-level key will be used when you are in that transaction.

Press the desired PF/PA key for the default help key or press CLEAR to use only transaction-level keys.

**The default help key is ==>**

(C)opyright SofTouch Systems, Inc. 1988

Press the PF key or PA key to be used as the help access key. Any of the available function keys, PF1 through PF24, PA1, PA2 or PA3 can be used as long as it is a different key from the help definition key. Be sure you choose a function key that is not needed by the application program that you are preparing to document. For this exercise, let's use PF12 as the help access key. When you press PF12, the following display is produced:

**HELP-WINDOWS RELEASE 3.0.xxxxxx  
On-Line Documentation/Help System**

**Current Terminal Status**

HELP-WINDOWS is active on this terminal in on-line Help **Definition** mode.

**The Help Definition key is ==> PF11**

**The Help Access key is ==> PF12**

To exit this screen, press the CLEAR key.

(C)opyright SofTouch Systems, Inc.

We now have both a help definition key and a help access key defined.

### STEP 3: INVOKING THE USER TRANSACTION

We're now ready to create a help display in auto-define mode which will reference our structured text. For this exercise, we will use the HDMO transaction display.

Clear the screen and enter HDMO. Upon keying HDMO and pressing ENTER, the HDMO screen display will appear as follows:

```
Screen: CM010N                               Operator: XXX
                SAMPLE CICS APPLICATION SCREEN
                FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number   _____
Customer Name     _____
Address Line 1    _____
Address Line 2    _____
City              _____
State __ Zip Code  _____

..... PRODUCTS INSTALLED .....
Product   Date   Original   Vendor   Maintenance   Expiration
Code     Sold    Cost       Number   Amount        Date
____     _____ _____   _____   _____   _____
____     _____ _____   _____   _____   _____
____     _____ _____   _____   _____   _____
____     _____ _____   _____   _____   _____

Use this screen for testing help displays with HELP-WINDOWS.
Refer to "LEARNING HELP-WINDOWS" for step-by-step instructions.
```

### **CREATING THE FIRST FIELD-LEVEL HELP DISPLAY**

We can define help at any of four levels of screen recognition<sup>24</sup>. These are:

- 1) Transaction level - this help display applies to any screen and any field of the transaction.
- 2) Screen level - this help display applies to any field of this particular screen of this transaction.
- 3) Field level - this help display applies to a particular field which could appear on more than one screen of this transaction.
- 4) Screen level/Field-level - this help display applies to a particular field of a particular screen of this transaction.

First, we will create field level help for the CUSTOMER NUMBER and PRODUCT CODE fields. We will then create transaction level help which will apply to all other fields.

### STEP 4: POINT TO THE DESIRED FIELD

The cursor should already be positioned in the CUSTOMER NUMBER field. If it is not, use the TAB key to put it there.

<sup>24</sup>See *LEVELS OF HELP ASSOCIATION*, section G4.



## STEP 5: PRESS THE HELP-DEFINITION HOT KEY

Now press the PF key that you designated as your "help definition" key when the HWIX,DEFINE command was done. This was PF11 in our exercise.

Upon pressing the help definition key, the following window will appear on the screen:

```
Screen: CM010N                               Operator: XXX
                SAMPLE CICS APPLICATION SCREEN
                FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number   _____
Customer Name     _____
Address Line 1    _____
Address Line 2    _____
City              _____
State __ Zip Code _____

==> AUTODEF
Tran   HDMO Desc Default_Window_for_Auto-Define_____
Screen 99999999
Field  06023_____
Seq    01
System _____
Xdoc   _____
STATUS=NEW RECORD

Use this screen for testing help displays with HELP-WINDOWS.
Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID
```

### EXPLANATION OF RESULTS:

The size and position of the window is determined from a generic model transaction key record which was provided with the HELP-WINDOWS system<sup>25</sup>.

The ==> in the window is the window command area. It currently contains the AUTODEF command, which must be there in order to respond to the display using the PF key prompts on the bottom row.

The TRANSACTION is filled in with HDMO, the transaction in progress.

The SCREEN ID is all nines, indicating that a unique screen identification is currently not defined.

FIELD contains the word '06023' which is row 06, column 23 of the screen<sup>26</sup>.

The SEQUENCE field contains 01 since this is the first transaction key record to be defined at this position of this screen.

The STATUS information is provided to indicate the current status of this transaction key record. NEW RECORD indicates that a new record is being created and has not yet been added to the file.

<sup>25</sup>See *TRANSACTION KEY RECORD MAINTENANCE, USING MODEL TRANSACTION KEY RECORDS* in section G7.

<sup>26</sup>Row/column is not the only way to identify help fields. You can also use text on the screen. See *THE AUTO-DEFINE PROCESS - STRUCTURED TEXT METHOD, DEFINING TEXT-IDENTIFIED FIELD-LEVEL HELP*, section G5, for more information.

The bottom row of the user screen is replaced with the directional options (key prompts) which are available at this stage of auto-define.

## STEP 6: CONNECTING THE HELP TEXT

For the first field help, we will leave the window where it is. We are now ready to complete the definition by connecting the help text. To do this, we simply key the system identifier (TESTSYS) in the SYSTEM field and the name that we used on the STRTFLD statement in the XDOC field<sup>27</sup>. We will also key a description of this help display in the description field following the transaction code.

The help text that we want for customer number is the one we called HDMO FIELD HELP1. Your screen should appear as follows:

```
Screen: CM010N                               Operator: XXX
      SAMPLE CICS APPLICATION SCREEN
      FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number   _____
Customer Name     _____
Address Line 1    _____
Address Line 2    _____
City              _____
State __ Zip Code _____

==> AUTODEF
Tran   HDMO Demonstration Help Screen Number 1_____
Screen 99999999
Field  06023_____
Seq    01
System TESTSYS_
Xdoc   HDMO FIELD HELP1__
STATUS=NEW RECORD

Use this screen for testing help displays with HELP-WINDOWS.
Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID
```

Now, press the ENTER key to complete the addition of this help definition. After pressing ENTER, if there are no errors or additional information needed, the STATUS field should display as "RECORD ADDED".

We have now completed the creation of a field-level help display using the structured text method.

## STEP 8: EXIT AUTO-DEFINE MODE

To exit auto-define mode, you may either press the "help definition" key again, press the CLEAR key or press PF3. The response to all entries is the same - the original HDMO screen will be re-displayed with no windows present.

## STEP 9: TEST THE HELP DISPLAY

<sup>27</sup>XDOC stands for CROSS-DOCUMENT. Names used in STRTFLD commands automatically create a special record called a cross-document record. For more information, see *CROSS-DOCUMENT RECORD MAINTENANCE* in section G7.

Now, with the cursor remaining in the CUSTOMER NUMBER field, press the help access key. For this exercise, we defined this as PF12.

Upon pressing the help key, you should see the help display appear as the end-user will see it. That is, only the text displays in the window, with all control statements removed.

It should appear as follows:

```
Screen: CM010N                               Operator: XXX
      SAMPLE CICS APPLICATION SCREEN
      FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number   _____
Customer Name     _____
Address Line 1    _____
Address Line 2    _____
City              _____
State __ Zip Code  _____

==>
CUSTOMER NUMBER -   Enter a 5-digit number to be assigned to this
                    customer. It must be a unique number.
CUSTOMER NAME -    The full customer or company name.
ADDRESS LINES 1&2 - The street and mailing address.
CITY, STATE, ZIP - Full city name, state code, zip code.

Use this screen for testing help displays with HELP-WINDOWS.
PF8 =Next PF7 =Prev PF3=Quit PF4 =Left PF5 =Right PF6 =Cut/Paste  LAST PAGE
```

Press the help key (PF12) again or press the CLEAR key to return to the HDMO transaction screen with no windows present.

## CREATING THE SECOND FIELD-LEVEL HELP DISPLAY

We are now ready to create the second field-level help display, which will be placed on the first PRODUCT field, in the lower half of the screen<sup>28</sup>.

### STEP 10: POINT TO THE DESIRED FIELD

Use the TAB or new-line key to position the cursor in the first PRODUCT field, on row 17.

### STEP 11: PRESS THE HELP-DEFINITION HOT KEY

Now press the PF key that you designated as your "help definition" key when the HWIX,DEFINE command was issued. This was PF11 in our exercise.

Upon pressing the help definition key, the following window will appear on the screen:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number \_\_\_\_\_  
Customer Name \_\_\_\_\_  
Address Line 1 \_\_\_\_\_  
Address Line 2 \_\_\_\_\_  
City \_\_\_\_\_  
State \_\_ Zip Code \_\_\_\_\_

==> AUTODEF  
Tran HDMO Desc Demonstration Help Screen Number 1 \_\_\_\_\_  
Screen 99999999  
Field 17005 \_\_\_\_\_  
Seq 01  
System \_\_\_\_\_  
Xdoc \_\_\_\_\_  
STATUS=NEW RECORD

Use this screen for testing help displays with HELP-WINDOWS.  
Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID

### EXPLANATION OF RESULTS:

The size and position of the window is now determined by using the transaction key record previously created with the first field-level help display. You will note that the DESC field contains the same description information that was keyed for the previous help display.

The ==> in the window is the window command area. It currently contains the AUTODEF command, which must be there in order to respond to the display using the PF key prompts on the bottom row.

The TRANSACTION is filled in with HDMO, the transaction in progress.

<sup>28</sup>It is possible to create a single help definition to apply to all of the PRODUCT fields. See *TRANSACTION KEY RECORD MAINTENANCE, EXAMPLES OF FIELD LOCATOR INFORMATION, MULTIPLE-OCCURRING FIELDS*, in section G7.

The SCREEN ID is all nines, indicating that a unique screen identification is currently not defined.

FIELD contains the word '17005' which is row 07, column 05 of the screen.

The SEQUENCE field contains 01 since this is the first transaction key record to be defined at this position of this screen.

The STATUS information again indicates NEW RECORD, as this help definition has not yet been added to the file.

The bottom row of the user screen is replaced with the directional options (key prompts) which are available at this stage of auto-define.

## STEP 12: CHANGING THE WINDOW CONFIGURATION

For this exercise, we're going to change the size and position of the help window. To do this, press PF2 (CHANGE WINDOW).

Upon pressing PF2, the display will change to:

Screen: CM010N	Operator: XXX
SAMPLE CICS APPLICATION SCREEN FOR HELP-WINDOWS AUTO-DEFINE TESTING	
Customer Number	_____
Customer Name	_____
Address Line 1	_____
Address Line 2	_____
City	_____
State __ Zip Code	_____
==> AUTODEF	
Starting row/col	12 003
Number of rows	08
Number of columns	075
Word wrap?	N (Y/N)
Refresh screen?	Y (Y/N)
Full screen?	N (Y/N)
STATUS=NEW RECORD	
Use this screen for testing help displays with HELP-WINDOWS.	
PF1=Left 2=Right 3=Quit 4=Up 5=Down 6=Taller 7=Shorter 8=Wider PF9=Narrow	

We're now viewing the current configuration of the window. All of the row and column information displayed is taken from the current position and size of the window, which originated with the model transaction key record.

There is a new set of PF key options at the bottom of the change-window screen. They are used to alter the window appearance as follows:

PF1 - Move the entire window to the left one column.

- PF2 - Move the entire window to the right one column.
- PF3 - Quit the screen-change function and return to the main AUTODEF display.
- PF4 - Move the entire window up one row.
- PF5 - Move the entire window down one row.
- PF6 - Deepen the window by moving the bottom bar down one row.
- PF7 - Shorten the window by moving the bottom bar up one row.
- PF8 - Widen the window by moving the right bar to the right one column.
- PF9 - Narrow the window by moving the right bar to the left one column.

There are three methods of changing or moving the window.

- 1) We can use the PF key prompts as described above to move one row or one column at a time. For instance, if we wanted a taller window, we could press PF3 a few times to move the entire window up a few rows, then press PF5 to extend the bottom bar down a few rows. With these PF key options, you can configure the window to any size and locate it anywhere on the screen<sup>29</sup>.
- 2) Another choice is to use the window adjustment commands. Any of the following can be entered in the command area (after the ==>) of the window<sup>30</sup>:

- WU=nn - Move the window UP nn rows.
- WD=nn - Move the window DOWN nn rows.
- WR=nn - Move the window RIGHT nn columns.
- WL=nn - Move the window LEFT nn columns.
- WT=nn - Make the window TALLER by nn rows.
- WS=nn - Make the window SHORTER by nn rows.
- WW=nn - Make the window WIDER by nn columns.
- WN=nn - Make the window NARROWER nn columns.

**[Note].** The equals sign (=) is not required for these commands; you can enter the number (nn) immediately after the command.

With the window adjustment commands, each time you enter a number of rows or columns, that number remains in force until you change it. Subsequent commands can be entered without the =nn and the same number of rows or columns previously specified will be used. This increment value also applies to the PF key prompts for changing the window. That is, after you once establish a number of rows or columns with a window adjustment command, subsequent pressing of a PF key to change the window will move the same number of rows or columns, rather than only one.

When finished with the window adjustment commands, press PF3 to restore the AUTODEF command.

- 3) The other choice is to enter the new window position and/or size directly in the fields displayed in the window and press ENTER. This will instantly reconfigure the window to the specified position and size.

To shift the window up so that the lower half of the screen is visible, yet keep the width and depth of the window the same, enter WU=8 (move the window up 8 rows) in the command area, erase the remainder of the AUTODEF command and press ENTER.

<sup>29</sup>See *DIRECTIONAL OPTIONS IN AUTO-DEFINE MODE, THE CHANGE-WINDOW DISPLAY*, in section G5.

<sup>30</sup>See *THE AUTO-DEFINE PROCESS - STRUCTURED TEXT METHOD, CHANGING THE APPEARANCE OF THE WINDOW* in section G5.

The screen should now appear as follows:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

==> WUP

Starting row/col 04 003

Number of rows 08

Number of columns 075

Word wrap? N (Y/N)

Refresh screen? Y (Y/N)

Full screen? N (Y/N)

STATUS=NEW RECORD

Product Code	Date Sold	Original Cost	Vendor Number	Maintenance Amount	Expiration Date
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____

Use this screen for testing help displays with HELP-WINDOWS.  
PF1=Left 2=Right 3=Exit 4=Up 5=Down 6=Taller 7=Shorter 8=Wider PF9=Narrow

Now we have the window where we want it and the size that we want. We are ready to exit the change-window screen and define the field-level help for the command field. First press PF3 to restore the AUTODEF command. Then press PF3 again to return to the main AUTODEF window.

Upon pressing PF3, the main AUTODEF information will re-display in the new window, as follows:



Screen: CM010N

Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

==> AUTODEF

Tran HDMO Desc Demonstration Help Screen Number 1\_\_\_\_\_

Screen 99999999

Field 17005\_\_\_\_\_

Seq 01

System \_\_\_\_\_

Xdoc \_\_\_\_\_

STATUS=NEW RECORD

Product	Date	Original	Vendor	Maintenance	Expiration
Code	Sold	Cost	Number	Amount	Date
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____

Use this screen for testing help displays with HELP-WINDOWS.

Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID

### STEP 13: CONNECTING THE HELP TEXT

We are now ready to complete the definition by connecting the help text. To do this, we simply key the system identifier (TESTSYS) in the SYSTEM field and the name that we used on the STRTFLD statement in the XDOC field<sup>31</sup>. We will also change the description of this help display in the description field following the transaction code.

The help text that we want for product code is the one we called HDMO FIELD HELP2. Your screen should appear as follows:

```
Screen: CM010N                               Operator: XXX
                SAMPLE CICS APPLICATION SCREEN
                FOR HELP-WINDOWS AUTO-DEFINE TESTING

==> AUTODEF
Tran   HDMO Desc Demonstration Help Screen Number 2_____
Screen 99999999
Field  17005_____
Seq    01
System TESTSYS_
Xdoc   HDMO FIELD HELP2__
STATUS=NEW RECORD

  Product  Date  Original  Vendor  Maintenance  Expiration
  Code     Sold   Cost      Number   Amount        Date
  _____
  _____
  _____
  _____
  _____

Use this screen for testing help displays with HELP-WINDOWS.
Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID
```

Now, press the ENTER key to complete the addition of this help definition. After pressing ENTER, the STATUS field should display as "RECORD ADDED".

We have now completed the creation of a field-level help display using the structured text method.

### STEP 14: EXIT AUTO-DEFINE MODE

To exit auto-define mode, you may either press the "help definition" key again, press the CLEAR key or press PF3. The response to all entries is the same - the original HDMO screen will be re-displayed with no windows present.

### STEP 15: TEST THE HELP DISPLAY

Now, with the cursor remaining in the PRODUCT field, press the help access key. For this exercise, we defined this as PF12.

<sup>31</sup>XDOC stands for CROSS-DOCUMENT. Names used in STRTFLD commands automatically create a special record called a cross-document record. For more information, see *CROSS-DOCUMENT RECORD MAINTENANCE* in section G7.

Upon pressing the help key, you should see the help display appear as the end-user will see it. That is, only the text displays in the window, with all control statements removed.

It should appear as follows:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

PRODUCT CODE -	Enter the 4-digit code that identifies this product.
DATE SOLD -	The date (MMDDYY) of the sale
ORIGINAL COST -	This is the sales price of the product.
VENDOR NUMBER -	The number assigned to the vendor for this product.
MAINTENANCE AMOUNT -	If the customer purchased a maintenance agreement, this is the amount of the annual maintenance fee.

Product Code	Date Sold	Original Cost	Vendor Number	Maintenance Amount	Expiration Date
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____

Use this screen for testing help displays with HELP-WINDOWS.  
PF8 =Next PF7 =Prev PF3=Quit PF4 =Left PF5 =Right PF6 =Cut/Paste

You will note that all of the text cannot fit in the depth of this window. In order to see the remainder of the text, press the NEXT key or enter 'N' in the command area and press ENTER. From there, the PREVIOUS command (or key) will return to this page<sup>32</sup>.

Press the HELP key (PF12) again or press the CLEAR key to return to the HDMO transaction screen with no windows present.

<sup>32</sup>When using NEXT, the words 'LAST PAGE' will appear in the lower right-hand corner when all text has been displayed.

## CREATING A TRANSACTION-LEVEL HELP DISPLAY

We are now ready to create the third and last help display of this exercise, which will be a transaction-level help display. Transaction-level help will be invoked any time the operator presses the help key when the cursor is in a field for which no field-level help has been created<sup>33</sup>.

It is a good idea to always create a transaction-level help definition, even though you may be documenting every field of a user screen. That way, something will be displayed, no matter where the cursor may be when the help key is pressed<sup>34</sup>.

### STEP 16: POINT TO THE DESIRED FIELD

Use the TAB or new-line key to position the cursor in any field on the screen except CUSTOMER NUMBER or the first PRODUCT field. For our exercise, place it in CUSTOMER NAME.

### STEP 17: PRESS THE HELP-DEFINITION HOT KEY

Now press the PF key that you designated as your "help definition" key when the HWIX,DEFINE command was done. This was PF11 in our exercise.

Upon pressing the help definition key, you will receive the same window configuration as that used in the first field-level help definition. The only difference will be the row-column identification of the help FIELD.

It will appear as follows:

```
Screen: CM010N                               Operator: XXX
      SAMPLE CICS APPLICATION SCREEN
      FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number   _____
Customer Name     _____
Address Line 1    _____
Address Line 2    _____
City              _____
State __ Zip Code _____

==> AUTODEF
Tran   HDMO Desc Demonstration Help Screen Number 1 _____
Screen 99999999
Field  07023 _____
Seq    01
System _____
Xdoc   _____
STATUS= NEW RECORD

Use this screen for testing help displays with HELP-WINDOWS.
Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID
```

<sup>33</sup>See *LEVELS OF HELP ASSOCIATION* in section G4.

<sup>34</sup>If the help key is pressed where no field help exists and no transaction-help exists, but there are field-level definitions for other fields on the screen, the following message *HX03001. NO HELP TEXT IS AVAILABLE. PRESS "CLEAR" TO RESTORE TRANSACTION* will display.



## STEP 18: IDENTIFYING THIS AS TRANSACTION-LEVEL HELP

As long as there is anything other than all nines in the FIELD identifier, this will be considered to be a field-level help display. To indicate that we want transaction-level help this time, tab to the FIELD identifier, press the ERASE EOF key and press ENTER. The FIELD identifier will now be filled with nines, indicating transaction-level help.

## STEP 19: CHANGING THE WINDOW CONFIGURATION

We also want to change the window to a larger window that covers most of the screen. To do this, press PF2 (CHANGE WINDOW).

Upon pressing PF2, the display will change to:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number \_\_\_\_\_

Customer Name \_\_\_\_\_

Address Line 1 \_\_\_\_\_

Address Line 2 \_\_\_\_\_

City \_\_\_\_\_

State \_\_ Zip Code \_\_\_\_\_

==> AUTODEF

Starting row/col 12 003

Number of rows 08

Number of columns 075

Word wrap? N (Y/N)

Refresh screen? Y (Y/N)

Full screen? N (Y/N)

STATUS=NEW RECORD

Use this screen for testing help displays with HELP-WINDOWS.

PF1=Left 2=Right 3=Quit 4=Up 5=Down 6=Taller 7=Shorter 8=Wider PF9=Narrow

We're now viewing the current configuration of the window, as in the previous exercise

This time we will directly enter the window configuration desired. TAB to the STARTING ROW/COL field and change the row number to '04'. Then change NUMBER OF ROWS to '17' and press ENTER.

The screen should now appear as follows:

```
Screen: CM010N                               Operator: XXX
                SAMPLE CICS APPLICATION SCREEN
                FOR HELP-WINDOWS AUTO-DEFINE TESTING

==> AUTODEF
Starting row/col    04 003
Number of rows     17
Number of columns  075
Word wrap?        N (Y/N)
Refresh screen?   Y (Y/N)
Full screen?      N (Y/N)
STATUS=CHANGES PENDING

Use this screen for testing help displays with HELP-WINDOWS.
PF1=Left 2=Right 3=Quit 4=Up 5=Down 6=Taller 7=Shorter 8=Wider PF9=Narrow
```

Now we have the window where we want it and the size that we want. We are ready to exit the change-window screen. Press ENTER to complete the changes, then PF3 to return to the main AUTODEF window.

Upon pressing PF3, the main AUTODEF information will re-display in the new window, as follows:

```
Screen: CM010N                               Operator: XXX
                SAMPLE CICS APPLICATION SCREEN
                FOR HELP-WINDOWS AUTO-DEFINE TESTING

==> AUTODEF
Tran    HDMO Desc Demonstration Help Screen Number 1_____
Screen  99999999
Field   99999999999999999999
Seq     01
System  _____
Xdoc    _____
STATUS=RECORD CHANGED

Use this screen for testing help displays with HELP-WINDOWS.
Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID
```





It should appear as follows:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

==>

The HDMO transaction is a sample CICS application that produces a screen display similar to many industry applications.

The screen is divided into two parts:

The upper half is used for entry of name and address information for each customer. The fields in this area are:

CUSTOMER NUMBER - Enter a 5-digit number to be assigned to this customer. It must be a unique number.

CUSTOMER NAME - The full customer or company name.

ADDRESS LINES 1&2 - The street and mailing address.

CITY, STATE, ZIP - Full city name, state code, zip code.

The lower half of the screen contains the product information for this customer. Key the following information for each product:

PRODUCT CODE - Enter the 4-digit code that identifies this product.

DATE SOLD - The date (MMDDYY) of the sale.

Use this screen for testing help displays with HELP-WINDOWS.

PF8 =Next PF7 =Prev PF3=Quit PF4 =Left PF5 =Right PF6 =Cut/Paste

In order to see the remainder of the text, press the NEXT key or enter 'N' in the command area and press ENTER. From there, the PREVIOUS command (or key) will return to this page<sup>35</sup>.

Press the help key (PF12) again or press the CLEAR key to return to the HDMO transaction screen with no windows present.

## SUMMARY OF THE STRUCTURED AUTO-DEFINE EXAMPLE

This has been an illustration of creating a help window with structured text that exercised some of the options available when using auto-define or help access. There are many more options which can be performed<sup>36</sup>.

In the next exercise, we will construct a help display using the fast-path method.

<sup>35</sup>When using NEXT, the words 'LAST PAGE' will appear in the lower right-hand corner when all text has been displayed.

<sup>36</sup>For a complete discussion of auto-define mode using normal structured text, see *THE AUTO-DEFINE PROCESS - STRUCTURED TEXT METHOD*, in section G5.

## CREATING A HELP WINDOW USING THE FAST-PATH METHOD

The procedure that we will be using to very quickly create a help window for a field in a user transaction is called the "fast-path" method of auto-define. This feature allows us to enter a user transaction, position the cursor to the desired field, press a PF or PA key and a window will pop up on the screen. We can then key the desired help instructions (text) in the window, press the ENTER key and we are finished.

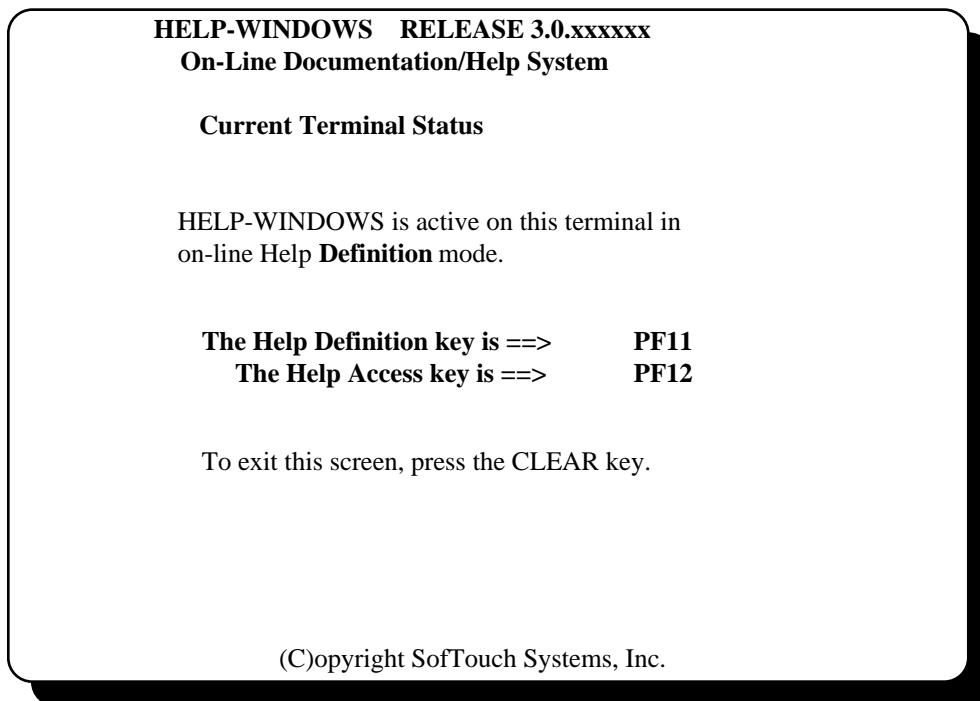
Fast-path help is to be used for "stand-alone" help displays only. That is, the text to be displayed is not part of a larger document.

The advantage of the fast-path method is that it is very quick, and does not require the creation of a structured hierarchy to define the text. It has the following disadvantages, however:

- Only one text record (20 lines of text) can be created per help definition. If more text is needed there is a method called "chaining" which allows you to connect two or more fast-path definitions together<sup>37</sup>.
- Fast-path help text cannot be printed or manipulated with the batch processing programs.
- Fast-path help text cannot be referenced (included) from other text.
- You cannot create a table of contents or index for fast-path text

### STEP 1: DEFINING THE "HELP DEFINITION" KEY

If you have not already done so, key HWIX,DEFINE from a clear screen and press ENTER, then press the desired "help definition" key, as described in step 1 of the first exercise of this manual<sup>38</sup>. Also define your help access key using the HWIX,ON command. If you're not sure if any keys are defined, enter the command, HWIX,STAT, which will produce the following display:



<sup>37</sup>See *THE AUTO-DEFINE PROCESS - FAST-PATH METHOD, ADDING MULTIPLE PAGES OF FAST-PATH HELP TEXT* in section G5.

<sup>38</sup>Or, key HWIX,DEF,PFxx (where PFxx is the key to be used) and press ENTER.

If no keys are displayed, you need to do the DEFINE and ON command to establish them.

## STEP 2: INVOKING THE USER TRANSACTION

Now enter a transaction code which will display a screen that you would like to document. For our purposes in this illustration, we will use the HDMO transaction.

Upon keying HDMO and pressing ENTER, the HDMO display will appear as follows:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number \_\_\_\_\_  
Customer Name \_\_\_\_\_  
Address Line 1 \_\_\_\_\_  
Address Line 2 \_\_\_\_\_  
City \_\_\_\_\_  
State \_\_ Zip Code \_\_\_\_\_

..... PRODUCTS INSTALLED .....

Product Code	Date Sold	Original Cost	Vendor Number	Maintenance Amount	Expiration Date
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____

Use this screen for testing help displays with HELP-WINDOWS.  
Refer to "LEARNING HELP-WINDOWS" for step-by-step instructions.

## STEP 3: POSITIONING THE CURSOR

Now position the cursor to the desired field on the screen where help is to be made available. For this exercise, let's use the CUSTOMER NAME field.

## STEP 4: PRESS THE HELP-DEFINITION HOT KEY

Now, without moving the cursor, press the PF or PA key that you designated as your "help definition" key when the HWIX,DEFINE command was done. In our first exercise, this was PF11.

Upon pressing the help definition key, the following window will appear on the screen:

```
Screen: CM010N                               Operator: XXX
                SAMPLE CICS APPLICATION SCREEN
                FOR HELP-WINDOWS AUTO-DEFINE TESTING
=====
==> AUTODEF
Tran      HDMO Desc Demonstration Help Screen Number 3_____
Screen    99999999
Field     07023_____
Seq       01
System    _____
Xdoc     _____
STATUS= NEW RECORD

Use this screen for testing help displays with HELP-WINDOWS.
Enter=Add 1=Key Screen 2=Change Window 3=Quit 4=Help Field 5=Text 6=Screen ID
```

#### **EXPLANATION OF RESULTS:**

The size and position of the window is determined from the help definition that was created for transaction-level help in the previous exercise.

The ==> in the window is the window command area. It currently contains the AUTODEF command, which must be there in order to respond to the display using the PF key prompts on the bottom row.

The TRANSACTION is filled in with HDMO, the transaction in progress.

The SCREEN ID is all nines, indicating that a unique screen identification is currently not defined.

FIELD contains 07023, which is row 07 column 23.

The SEQUENCE field contains 01 since this is the first transaction key record to be defined at this position of this screen.

The STATUS information is provided to indicate the current status of this transaction key record. NEW RECORD indicates that a new help definition is being created and has not yet been added to the file.

The bottom row of the user screen is replaced with the directional options (key prompts) which are available at this stage of auto-define.

#### **STEP 5: CHANGING THE WINDOW**

Since we're going to do a field help definition, we need to make the window smaller and place it so that the CUSTOMER NAME field can be seen.

As in the previous exercises, press PF2 to enter change-window mode. Then, either use the PF keys, window commands or direct entry of row/column information to size the window like you want it.

By setting the STARTING ROW/COL to '12 003', NUMBER OF ROWS to '08' and NUMBER OF COLUMNS to '075' we will have the same window used in the previous exercises, as follows:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number \_\_\_\_\_  
Customer Name \_\_\_\_\_  
Address Line 1 \_\_\_\_\_  
Address Line 2 \_\_\_\_\_  
City \_\_\_\_\_  
State \_\_ Zip Code \_\_\_\_\_

==> AUTODEF  
Starting row/col 12 003  
Number of rows 08  
Number of columns 075  
Word wrap? N (Y/N)  
Refresh screen? Y (Y/N)  
Full screen? N (Y/N)  
STATUS=NEW RECORD

Use this screen for testing help displays with HELP-WINDOWS.  
PF1=Left 2=Right 3=Quit 4=Up 5=Down 6=Taller 7=Shorter 8=Wider PF9=Narrow

Now, press PF3 until the main AUTODEF display reappears. The next step is to enter the help text.



STEP 6: PREPARE TO ENTER THE HELP TEXT

What we want to do is provide some help instruction text to go into the window with its current configuration. Therefore we will press PF5 (TEXT), which produces the following display:

The screenshot shows a mainframe application window with the following content:

Screen: CM010N Operator: XXX

SAMPLE CICS APPLICATION SCREEN  
FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number \_\_\_\_\_  
Customer Name \_\_\_\_\_  
Address Line 1 \_\_\_\_\_  
Address Line 2 \_\_\_\_\_  
City \_\_\_\_\_  
State \_\_ Zip Code \_\_\_\_\_

==> EDIT TRAN

Use this screen for testing help displays with HELP-WINDOWS.

ENTER=ADD 3=Q 4=WIND 5=LISTT 6=COLOR 7=PREV 8=NEXT 9=UPD 10=ADD 11=DEL 12=...

STEP 7: ENTER THE HELP TEXT

We now proceed to key the text exactly as we want it to appear in the window. The cursor is initially positioned on the first line below the command area, which now contains EDIT TRAN. As we key, when the cursor reaches the right-hand edge of the window it will automatically skip to the next line and when it reaches the end of the last line it skips back to the command area.

When we have keyed all of the text that we want, it might appear as follows:

```
Screen: CM010N                               Operator: XXX
                SAMPLE CICS APPLICATION SCREEN
                FOR HELP-WINDOWS AUTO-DEFINE TESTING

Customer Number   _____
Customer Name     _____
Address Line 1    _____
Address Line 2    _____
City              _____
State __ Zip Code _____

==> EDIT TRAN
The customer name field must contain the full name of the customer. If this
is a company name, use the standard company name format for your entry.

For a person's name, enter the last name first, then a comma, then the first
name, middle (if known) and any titles (Jr. Esq., etc.)

Use this screen for testing help displays with HELP-WINDOWS.
ENTER=ADD 3=Q 4=WIND 5=LISTT 6=COLOR 7=PREV 8=NEXT 9=UPD 10=ADD 11=DEL 12=I
```

Let's assume that this is all we want to say for this help display and it will fit in the window size displayed. Upon completion of keying the text, we press ENTER. This causes the transaction key record<sup>39</sup>, with the text that we keyed, to be added to the file.

Note that we could key more text than would fit in the window, and this is often needed. To do this, we would press PF4, the window-edit key. Pressing PF4 the first time will expand the screen to a full-screen text editor display, where one full page of text can be entered and the editor sub-commands can be used. Pressing PF4 again will return to the window display<sup>40</sup>.

Now press PF3 to exit the EDIT TRAN command and return to the main AUTODEF display. We have now completed the creation of a field-level help display using the fast-path method.

#### STEP 8: EXIT AUTO-DEFINE MODE

To exit auto-define mode, you may either press the "help definition" key again, press the CLEAR key or press PF3. The response to all entries is the same - the original HDMO screen will be re-displayed with no windows present.

#### STEP 9: TEST THE HELP DISPLAY

Now, with the cursor remaining in the CUSTOMER NAME field, press the help access key. For this exercise, we defined this as PF12.

<sup>39</sup>Help definitions are stored in special records called 'transaction key records'.

<sup>40</sup>For more information on this, see *THE AUTO-DEFINE PROCESS - FAST-PATH METHOD, ADDING MORE TEXT THAN CAN FIT IN THE WINDOW*, in section G5.

Upon pressing the help key, you should see the help display appear as the end-user will see it. That is, only the text displays in the window, with all control statements removed.

It should appear as follows:

Screen: CM010N	Operator: XXX
SAMPLE CICS APPLICATION SCREEN FOR HELP-WINDOWS AUTO-DEFINE TESTING	
Customer Number	_____
Customer Name	_____
Address Line 1	_____
Address Line 2	_____
City	_____
State __ Zip Code	_____

==>

The customer name field must contain the full name of the customer. If this is a company name, use the standard company name format for your entry.

For a person's name, enter the last name first, then a comma, then the first name, middle (if known) and any titles (Jr. Esq., etc.)

Use this screen for testing help displays with HELP-WINDOWS.  
PF1 =Next PF2 =Previous PF3 =Left PF4 =Right PF5 =Cut/Paste

#### SUMMARY OF THE FAST-PATH AUTO-DEFINE EXAMPLE

This has been a very simple illustration that did not explore most of the options available when using auto-define or help access. It serves, however, to illustrate how simple it is to define a help display using the fast-path method.

If you plan to document screens for on-line help only, and if you do not intend to have very much documentation for each field, and if you do not need to print the documentation, the fast-path method will work well for you. Otherwise, you should use the structured text method.

## SUMMARY OF THE GETTING-STARTED EXERCISES

If you did all of the exercises herein, you now have created four help displays for the HDMO transaction. Two are structured text field-level help windows, one is a fast-path field-level help window and one is a structured text transaction-level help window. Some of the things that we did not do in these exercises are:

- 1) Create a full-screen help display<sup>41</sup>.
- 2) Create a menu of help topics<sup>42</sup>.
- 3) Use text on the screen to identify a help display<sup>43</sup>.
- 4) Add more than one page of fast-path help text<sup>44</sup>.
- 5) Define a screen-level help display<sup>45</sup>.

The *INTRODUCTION* section of the User's Guide contains a topic called *REFERENCE GUIDE TO PRODUCT FEATURES*. In this, every feature and option of HELP-WINDOWS is listed, along with the section and topic references where they can be found. When you're not sure whether what you want can be done, or do not know how to do it, this section can point you to the location in the manual that describes it.

This ends the exercises for getting started with HELP-WINDOWS. You should know enough about the product at this point to create some realistic help displays or on-line documents for your environment. We urge you, however, in order to fully understand the product, to read the complete reference guide or at least the discussions that were referenced in this manual.

## HELP ACCESS MODE

---

<sup>41</sup>See *THE AUTO-DEFINE PROCESS - STRUCTURED TEXT METHOD, DEFINING A FULL-SCREEN HELP DISPLAY* in section G5.

<sup>42</sup>See *DEFINING A HELP MENU* in section G5.

<sup>43</sup>See *THE AUTO-DEFINE PROCESS - STRUCTURED TEXT METHOD, DEFINING TEXT-IDENTIFIED FIELD-LEVEL HELP* in section G5.

<sup>44</sup>See *THE AUTO-DEFINE PROCESS - FAST-PATH METHOD, ADDING MULTIPLE PAGES OF FAST-PATH HELP TEXT* in section G5.

<sup>45</sup>See *THE AUTO-DEFINE PROCESS - STRUCTURED TEXT METHOD, DEFINING SCREEN-LEVEL HELP* in section G5.

## USING ON-LINE HELP DISPLAYS

This section of the Learning HELP-WINDOWS manual deals with the use of HELP-WINDOWS by the end user. That is, the retrieval of help displays while in a user transaction. This is called Help Access Mode, or simply 'help mode'.

This section should be used by the text-writer<sup>46</sup> for a deeper understanding of the kinds of specifications that can be made when defining help displays that will produce the various results discussed herein.

Command entry in help mode, screen chaining, multiple window displays, nested help and the cut/paste feature are discussed. In addition, a method is discussed for allowing end users to retrieve and browse on-line documents, without the use of the maintenance transaction.

**[Note].** The examples in this section refer to the illustrations used in the section G5 of the User's Guide, which describes the auto-define process in detail.

## THE EXIT PROCESSOR

Once an on-line help definition has been created and connected to a user transaction, you are ready to use HELP-WINDOWS in help access mode.

Help access mode is controlled by the EXIT PROCESSOR program of HELP-WINDOWS. This program performs the activation and deactivation of help access mode at the terminal, as well as the search and retrieval of help displays when the help key is pressed.

The exit processor is the part of HELP-WINDOWS which handles retrieval and display of help screens and windows in help access mode. It operates by being activated or deactivated by the operator at a terminal. Once activated for a terminal, help access is available for that terminal until the exit processor is deactivated for that terminal. It may also be activated automatically during CICS startup.

The exit processor uses a different transaction code than the maintenance processor. This transaction code can be anything defined by the system programmer at installation, but for the purposes of this manual, it will be 'HWIX'.

The HWIX transaction is not used often during the course of the day. It will be used to activate HELP-WINDOWS for help access mode in the morning, and optionally to deactivate it at night.

## ACTIVATION OF HELP ACCESS MODE

Before a help display can be retrieved and displayed on the screen, help access mode must be "activated" for the terminal. This would normally be done once, immediately after signing on.

The activation process can be automated so that help access mode is active as soon as the operator signs on<sup>47</sup>. If the activation process is not automated, the operator must enter the "ON" command at the terminal to activate it. The command is entered immediately after the comma which separates the transaction code from the command. The 'ON' command would be entered from a clear screen as

---

<sup>46</sup>The text-writer is the person creating the on-line help displays.

<sup>47</sup>See *AUTOMATIC ACTIVATION OF HELP ACCESS MODE* in section T2 of the *TECHNICAL REFERENCE MANUAL*.

HWIX,ON or HWIX,INIT,PFxx<sup>48</sup>

---

<sup>48</sup>The INIT command performs the same function as ON but lets you designate your help key with the command.

If the help access key is entered with the command (HWIX,INIT,PFxx), the following prompt screen is bypassed. The value of the PF or PA key entered with the command becomes the help access key.

Upon entry of the ON or INIT command, the following screen will be displayed<sup>49</sup>:

**HELP-WINDOWS Release 3.0.xxxxxx  
On-Line Documentation/Help System**

**Activation of On-line Help Access Mode**

To activate Help-Windows for your terminal, press the PF or PA key to be used as the default help key.

If a particular key has already been associated with the transaction(s) to be used, you may not need a default help key, in which case you can press CLEAR.

If you have both a default and a transaction-level help key, the transaction-level key will be used when you are in that transaction.

Press the desired PF/PA key for the default help key or press CLEAR to use only transaction-level keys.

**The default help key is ==>**

(C)opyright SofTouch Systems, Inc. 1988

This screen is requesting that the PF or PA key to be used as the DEFAULT help key be pressed.

The default help key is the PF or PA key to be used as the "hot" key to retrieve help displays for which no transaction-level help keys have been defined.

It is possible to define a help key to be used as the "hot" key for certain transactions only. If this is the case, the default help key will not function as a "hot" key when you are using that transaction. It will only function in transactions which do not have a transaction-level key defined. If all of the transactions that you will use have transaction-level help keys defined, you may not need a default help key<sup>50</sup>.

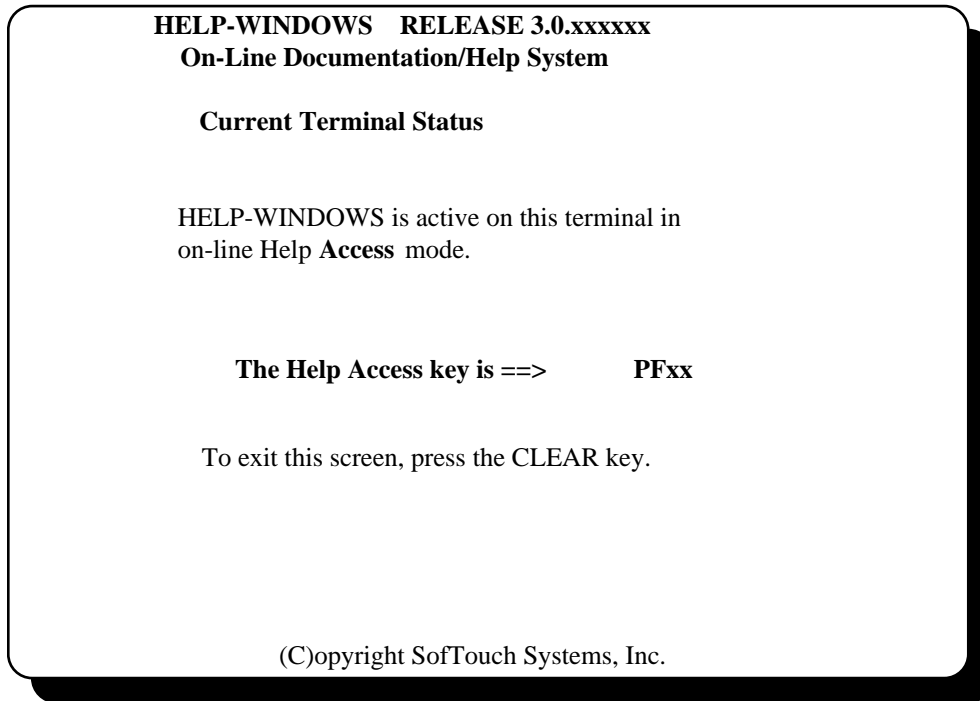
If a default help key is desired, press the PF or PA key to be used. If no default help key is desired, press the CLEAR key.

<sup>49</sup>If *BYPASS STATUS SCRN* is coded YES and the help key has been preset in the customization table, the prompt screen will be bypassed. HWIX,INIT will always produce the prompt screen.

<sup>50</sup>See *TRANSACTION KEY RECORD MAINTENANCE, FIELDS OF THE TRANSACTION KEY RECORD* in section G7 of the User's Guide.



Upon responding to this screen, the following screen will display<sup>51</sup>:



This is a status screen which confirms your entry. The mnemonic for the key that you pressed will be displayed in place of PFxx. If CLEAR was pressed, the same screen will display, but there will be no default help key displayed.

This completes the activation process. Help access mode is now active on this terminal. You may now proceed to use it to retrieve help displays.

---

<sup>51</sup>If *BYPASS STATUS SCRIN* is coded YES in the customization table, this screen will be bypassed.

## RETRIEVING HELP DISPLAYS

To retrieve a help display, first initiate a user transaction for which a help display has been defined. Then proceed to the appropriate screen (if not already there) and press your help key.

We will use the FCTD transaction, as it is used in section G5 of the User's Guide, to illustrate help displays in action. First we enter the FCTD transaction with no operands, resulting in the following opening menu of functions:

CICS-FILESERV, RELEASE 1.3 VSE, PTF LEVEL xxxxxx

This utility controls the definition and maintenance of  
FCT file descriptions for CICS

The following comands are available:

- (H)elp - Request operator instructions
- (I)nquire - Display dynamic FCT status, all or single
- (N)ew - Create a new dynamic FCT entry
- (D)elete - Delete a dynamic FCT entry
- (O)pen - Change an FCT entry to 'active' and open the file
- (A)ctivate - Change a dynamic FCT to 'active'
- (C)lose - Close a file (if open)
- (B)ackout - Remove a previously activated FCT entry

The command may be entered from a clear screen or from this screen.  
Follow the command with at least one space, then the dataset name.

— \_\_\_\_\_

In the examples in section G5, we created a transaction-level help display which applied to the FCTD transaction in general. This means that if the cursor is positioned somewhere on the screen where there are no field-level help definitions, the transaction-level help display will be the one which is retrieved.

Upon pressing the help key at this screen, this help display will display in a window as it was defined:

CICS-FILESERV, RELEASE 1.3 VSE, PTF LEVEL xxxxxx

This utility controls the definition and maintenance of  
FCT file descriptions for CICS

The following comands are available:

==>

CICS-FILESERV is an on-line utility which provides an RDO-like  
facility for the CICS File Control Table.

File Control Definitions can be created, changed, activated and  
deactivated on-line, without the necessity of recycling CICS.

In addition, CICS-FILESERV provides a facility similar to CEMT  
for opening and closing  
or they may be opened automatically upon first access.

PF8 =NEXT PF7 =PREVIOUS PF4 =LEFT PF5 =RIGHT PF6 =CUT/PASTE

As you can see, the text which was written and associated with the transaction key at the transaction level is displayed in a window of the size and shape defined in the transaction key record.

## DIRECTIONAL OPTIONS AT A HELP DISPLAY

The last line of the application screen is used to display the directional options available at this point. You can invoke one of these options by pressing the corresponding PF key or by entering the equivalent command in the command area of the window, which is immediately after the ==> symbol. Note that the PF keys for each of the options could be changed to a different key (see CUSTOMIZATION). The available options are:

- PF8 - (NEXT command) Display the next page of text after this one. The display will move forward in the text for the depth of the window, displaying as much text as can be displayed in the window.
- PF7 - (PREVIOUS command) Display the previous page of text before this one. The display will move backward in the text for the depth of the window, displaying as much text as can be displayed in the window.
- PF4 - (LEFT command) Move the display to the left one or more positions. Pressing PF4 will pan left one position. To pan more than one position, enter LE=nn (or LEFT=nn) in the command area, where nn is the number of positions to pan.

Note that panning left means that you logically move the window to the left, over the text. It will appear as though the text moves to the right.

- PF5 - (RIGHT command) Move the display to the right one or more positions. Pressing PF5 will pan right one position. To pan more than one position, enter RI=nn (or RIGHT=nn) in the command area, where nn is the number of positions to pan.

Note that panning right means that you logically move the window to the right, over the text. It will appear as though the text moves to the left.

**[Note].** If word-wrap is in effect for this display, there is no need to do any panning. Word-wrap will cause sentences of text to break on word boundaries so that all of the text will display, no matter what size the window. With word-wrap, you only need to page forward to read all of the text.

- PF6 - (CUT/PASTE command). To use the cut/paste function, place the cursor anywhere in one of the words in the help display and press PF6. Upon pressing PF6, the help display will disappear and the application screen is re-displayed. The word where the cursor was placed is copied from the help display and placed into the field of the application display where the cursor was positioned when the help key was originally pressed<sup>52</sup>.

---

<sup>52</sup>See USING THE CUT/PASTE FEATURE at the end of this section.

## ENTERING COMMANDS AT A HELP DISPLAY

Following are all of the valid commands that can be entered in help access mode. The commands are listed in alphabetic order with the minimum abbreviation shown in parentheses with each command.

These commands can be entered in the command area (after the ==> symbol) of the help window or screen.

### DOWN (D)

Keywords: LINES (optional)

The DOWN command moves the display forward in the text for the specified number of lines. It may be entered in any of the following ways:

D	To move down one line.
D=nnn	To move down nnn lines.
D LINES=nnn	To move down nnn lines.
Dnnn	To move down nnn lines.

[Note]. The number of lines (nnn) may be entered as 1 to 4 numeric digits.

### LEFT (LE)

Keywords: CHARS (Optional)

The LEFT command moves the display to the left the specified number of character positions. It may be entered in any of the following ways:

LE	To move left 10 positions.
LE=nnn	To move left nnn positions.
LE CHARS=nnn	To move left nnn positions.
Lnnn	To move left nnn positions.

[Note]. The number of positions (nnn) may be entered as 1 to 4 numeric digits.

### RIGHT (RI)

Keywords: CHARS (Optional)

The RIGHT command moves the display to the right the specified number of character positions. It may be entered in any of the following ways:

RI	To move right 10 positions.
RI=nnn	To move right nnn positions.
RI CHARS=nnn	To move right nnn positions.
Rnnn	To move right nnn positions.

[Note]. The number of positions (nnn) may be entered as 1 to 4 numeric digits.

## SEARCH (S)

Keywords: TEXT (Optional)

The SEARCH command is used to find a character string in a text record. SEARCH will start with the first text line currently displayed, then continue through multiple text records until either the character string is found or the end of text for that section/subject is found.

The help mode search operates in formatted mode, that is, the text referenced by INCLUDE statements is expanded and included in the search and text commands such as <SPACE> and <EJECT> are interpreted. In other words, the list mode search will not find any of the text commands enclosed in greater than/less than signs (<>) because it interprets the text command prior to performing the search. If you want to search for a text command, you must perform the search in edit mode.

You can perform a help mode search for a character string in either of the following ways:

- 1) Enter S=xxxxxxx to search forward from this point.
- 2) Enter T=xxxxxxx to search backward from this point.

The character string to be located (indicated by xxxxxxxx in the examples above) is entered with no delimiters. If imbedded spaces are present in the character string, go ahead and enter the space as it appears. However, you can not include a comma in the search string. A comma denotes the end of the string and need not be entered in help mode.

If the character string is found, the command will stop and display the text record containing the search string. The line of text containing the search string will be the first line displayed.

To continue the search, just press ENTER. The SEARCH command will continue forward (or backward) until the next matching character string is found or end of data if reached.

You do not need to be concerned about upper/lower case when doing a search. SEARCH will find the data whether the case of the data matches the case of the search string or not<sup>53</sup>.

## UP (U)

Keywords: LINES (Optional)

The UP command moves the display backward in the text for the specified number of lines. It may be entered in any of the following ways:

U	To move up one line.
U=nnn	To move up nnn lines.
U LINES=nnn	To move up nnn lines.
U <sub>nnn</sub>	To move up nnn lines.

**[Note].** The number of lines (nnn) may be entered as 1 to 4 numeric digits.

---

<sup>53</sup>See the SEARCH command in section G6.

## RETRIEVING A SCREEN-LEVEL HELP DISPLAY

The previous example illustrated the retrieval of a transaction-level help display. That particular display was retrieved because there was neither a screen-level nor a field-level display defined. To continue with our FCTD example, we defined a screen-level help display for the second FCTD screen, as follows:

```

FCTD N      CICS-FILESERV CREATE NEW FILE DEFINITION

Dataset name _____          Original status  CLOSED
Access method VSAM                Current status  CLOSED
File label _____            Automatic open ? Y
Service requests upd new del bro  ___ ___ ___ ___  Date created   _____
Auto journalling (jid) no          Last opened    _____
Journal requests (jreq) ___ ___  Number of opens _____
Automatic logging (log)  -
Resource security (rsl)  -
Variable length records  ___
VSAM access type          KSDS VSAM user catalog  ___  VSAM
VSAM bufnd  ___          VSAM bufni  ___          VSAM bufsp  ___
VSAM password _____          VSAM string no. 01
----- Non VSAM files only -----
Block size  ___          Record size  ___          ___
Channel/units  ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___

F101. KEY DATA FIELDS AND PRESS ENTER TO CREATE DEFINITION
  
```

The display was keyed to the word 'Dataset' which appears at row 3, column 2. When this screen displays, the cursor is placed under the command field on row 1. When the help key is pressed, HELP-WINDOWS will determine that there is not a field-level help display defined for that position; however there is a screen-level display for a screen with the FCTD transaction code and the word 'Dataset' at row 3, column 2. The screen-level text will be retrieved and displayed as follows:

```

FCTD N      CICS-FILESERV CREATE NEW FILE DEFINITION

Dataset name _____
Access method VSAM
File label _____
Service requests upd new del bro _____
Auto journalling (jid) no
Journal requests (jreq) _ _ _
Automatic logging (log) -
Resource security (rsl) -
Variable length records _____
VSAM access type          KSDS V
VSAM bufnd _____     VSAM b
VSAM password _____
----- Non V
Block size _____ Record size _____
Channel/units _____

F101. KEY DATA FIELDS AND PRESS ENTER TO CREATE DEFINITION

```

==>

This is the file definition screen. It has a field corresponding to every keyword which can be defined with the DFHFCT macro.

In addition, it has the necessary fields to allow the file to be dynamically allocated.

To add a new file definition, fill in the required fields and press ENTER. After adding the definition, you can open

## RETRIEVING A FIELD-LEVEL HELP DISPLAY

In our example, we defined a text-identified field-level help window to be associated with the 'Dataset Name' field. If the operator places the cursor anywhere in the data field corresponding to Dataset Name and presses the help key, the field-level display will be retrieved.

It will appear as follows:

```

FCTD N      CICS-FILESERV CREATE NEW FILE DEFINITION

Dataset name _____          Original status  CLOSED
Access method VSAM              Current status  CLOSED
Fi
Se ==>
A   The Dataset Name field corresponds to the DATASET= keyword of the
Jc DFHFCT macro. It also corresponds to the DLBL file ID for DOS users or
A   the DDNAME for MVS users.
R
V   Enter the Dataset Name as up to 8 alphanumeric characters (7 for DOS).
V
VSAM bufnd _____ VSAM bufni _____ VSAM bufsp _____
VSAM password _____ VSAM string no. 01
----- Non VSAM files only -----
Block size _____ Record size _____
Channel/units _____

F101. KEY DATA FIELDS AND PRESS ENTER TO CREATE DEFINITION

```

Now the help window and associated text defined for the Dataset Name field is displayed. The same directional options apply at this point.



## CHAINED HELP WINDOWS

Chained help windows occur when multiple transaction key records are defined with the same transaction code, screen identifier and field identifier, with incrementing sequence numbers.

In this situation, each transaction key references a different section of text to be displayed, and may define a different window configuration.

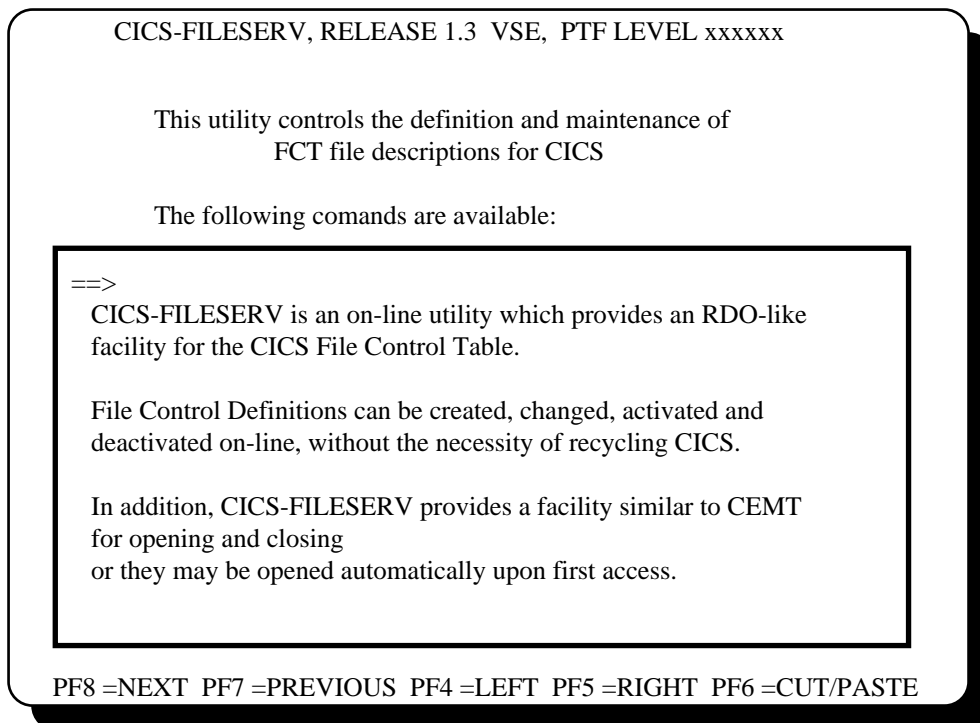
In help access mode, the text associated with the first transaction key will be retrieved when the help key is pressed. Upon browsing forward with the NEXT command or key, all of the text associated with the first transaction key is displayed. When the end of text is reached for the first definition, another NEXT command will cause the second transaction key record to be retrieved, whereupon the text and window configuration for that definition is displayed.

If the window configuration of the second transaction key is the same as the first, you cannot tell that chaining has occurred. If however, the second key has a different size and/or position for the window, you will see a different display.

One of the options for this type of operation is the REFRESH SCREEN option of the transaction key record. If this field is specified as YES, the window of the previous transaction key will disappear when chaining occurs and the new window is displayed. If REFRESH SCREEN is specified as NO, the first window remains on the screen and the second window is displayed in addition to, and sometimes on top of it.

The result is that multiple windows can appear on the screen at one time, each with its associated text. An example of multiple window chaining could appear as follows.

First we have the application screen with a single window displayed:



CICS-FILESERV, RELEASE 1.3 VSE, PTF LEVEL xxxxxx

This utility controls the definition and maintenance of  
FCT file descriptions for CICS

The following commands are available:

==>  
CICS-FILESERV is an on-line utility which provides an RDO-like  
facility for the CICS File Control Table.

File Control Definitions can be created, changed, activated and  
deactivated on-line, without the necessity of recycling CICS.

In addition, CICS-FILESERV provides a facility similar to CEMT  
for opening and closing  
or they may be opened automatically upon first access.

PF8 =NEXT PF7 =PREVIOUS PF4 =LEFT PF5 =RIGHT PF6 =CUT/PASTE

Now we browse forward with PF8. If this is the end of text for the first transaction key, and there is another record with the same transaction code, screen ID and field ID but with a different

sequence number, you would see still a different display. The next display might appear as follows:

CICS-FILESERV, RELEASE 1.3 VSE, PTF LEVEL xxxxxx

This utility controls the definition and maintenance of FCT file descriptions for CICS

The following commands are available:

==> Commands are available to open, close, activate, deactivate, allocate, deallocate and backout a file.

==> In addition, the multiple file display can be used to display information about several files at once, either for all files or generically equivalent files.

==> CICS-FILESERV is an interactive facility for the CICS File Control Definition (FCT) File Control Definition (FCT) deactivated on-line, with the following commands:

In addition, CICS-FILESERV provides a facility similar to CEMT for opening and closing files. Files may be opened explicitly or they may be opened automatically upon first access.

PF8 =NEXT PF7 =PREVIOUS PF4 =LEFT PF5 =RIGHT PF6 =CUT/PASTE

Continuing forward, we might encounter still another transaction key which configures the display as follows:

CICS-FILESERV, RELEASE 1.3 VSE, PTF LEVEL xxxxxx

This utility controls the definition and maintenance of FCT file descriptions for CICS

The following commands are available:

==> Commands are available to open, close, activate, deactivate, allocate, deallocate and backout a file.

==> In addition, the multiple file display can be used to display information about several files at once, either for all files or generically equivalent files.

==> CICS-FILESERV is an interactive facility for the CICS File Control Definition (FCT) File Control Definition (FCT) deactivated on-line, with the following commands:

A GROUP facility is provided to group files of unlike names together. A single command can then be performed for the entire group.

In addition, CICS-FILESERV provides a facility similar to CEMT for opening and closing files. Up to 99 files can be in one group. Files may be opened explicitly or they may be opened automatically upon first access.

PF8 =NEXT PF7 =PREVIOUS PF4 =LEFT PF5 =RIGHT PF6 =CUT/PASTE

The window which contains the cursor is the current window and is the one which responds to any commands or PF key directional options.

Browsing backward from this point with the PREVIOUS command or key will remove the windows one at a time as the beginning of text for each window is passed.

Also, pressing the help key again will revert back to the previous window. Continued pressing of the help key will take you all the way back to the application display with no windows. Pressing the CLEAR key will take you directly back to the application screen, no matter how many windows are on the screen.

## **HELP DISPLAYS WITHIN HELP TEXT (NESTED HELP)**

It is possible to define a help screen or help window, then define another help display to be keyed to one of the words of text in the first display. This is called 'nested' help.

The definition for this is accomplished by coding the word in the first display with an unprotected attribute so that the operator can tab to it (it should probably also be high intensity or a different color so as to be noticed), then defining a transaction key using that word as the field text identifier. For this definition, CURSOR IN FIELD TEXT should be specified as YES.

To retrieve the nested help display while in help mode, you tab to the field, then press the ENTER key instead of the help key.

An example of this would be as follows. On the first display, the word RDO might be set up with a transaction key using 'RDO' as a field text identifier. RDO is coded to be high-intensity, unprotected.

The screen appears as follows:

CICS-FILESERV, RELEASE 1.3 VSE, PTF LEVEL xxxxxx

This utility controls the definition and maintenance of  
FCT file descriptions for CICS

The following comands are available:

==>

CICS-FILESERV is an on-line utility which provides an **RDO**ike facility for the CICS File Control Table.

File Control Definitions can be created, changed, activated and deactivated on-line, without the necessity of recycling CICS.

In addition, CICS-FILESERV provides a facility similar to CEMT for opening and closing files. Files may be opened explicitly or they may be opened automatically upon first access.

PF8 =NEXT PF7 =PREVIOUS PF4 =LEFT PF5 =RIGHT PF6 =CUT/PASTE

If the operator tabs to the RDO field and presses ENTER, the next display could appear as:

CICS-FILESERV, RELEASE 1.3 VSE, PTF LEVEL xxxxxx

This utility controls the definition and maintenance of  
FCT file descriptions for CICS

The following comands are available:

==>

CICS-FILESERV is an on-line utility which provides an **RDO**ike facility for the CICS File Control Table.

File Control Definitions can be created, changed, activated and deactivated on-line, without the necessity of recycling CICS.

In addition, CICS-FILESERV provides a facility similar to CEMT for opening and closing files. Files may be opened explicitly or they may be opened automatically upon first access.

==>

RDO is the CICS facility for on-line definition of table entries such as PCT, PPT and TCT definitions.

PF8 =NEXT PF7 =PREVIOUS PF4 =LEFT PF5 =RIGHT PF6 =CUT/PASTE

The nested window could, of course appear anywhere on the screen, or it could be a full-screen display.

Nested help windows work in the same manner as chained windows. That is, pressing the help key causes the current window to disappear and reverts back to the previous window in the nest.

## **EXITING FROM A HELP DISPLAY**

When you are ready to exit a help display and return to transaction operation, there are two methods of doing so:

- 1) Pressing the help key again will remove the current window or help screen and return to the previous level. If only one window or screen is present, this will return you to the application screen. If multiple help displays are present, either because of screen chaining or nested help displays, pressing the help key takes you back to the previous display. Pressing it again takes you back another level, etc.
- 2) Pressing the CLEAR key always returns you to the application screen, regardless of how many help displays are concurrently active.

Once you return to the application screen you may continue with the next input to that application, just as if you had not interrupted it. The cursor is replaced in whatever position it was in when you first pressed the help key and all attributes are restored to their original value.

## **VIEWING DOCUMENTS WITHOUT USING HWIN**

There are at least two techniques which make it possible for the end user at a terminal to view on-line documents without using the maintenance transaction. In fact, this method is much easier for the end-user since they don't need to know anything about hierarchical structure, conversations, or fast-path commands. The commands available to the end-user would be any of the commands which can be issued in help-access mode.

## **USING A HELP MENU**

There are various ways to create menus of help displays. You can use any existing programming technique to build a menu or you can use HELP-WINDOWS to do it.

If you have an application development system or fourth-generation language package you could create a menu panel using the "paint-the-screen" technique available in most of these packages, then define a transaction key for a word which appears in each of the selection prompts of the menu screen.

You could accomplish the same thing with an application program using BMS mapping or any other screen-building technique.

For both of these techniques you should set an unprotected attribute code somewhere in the text of the selection prompt to allow the operator to tab the cursor to that position and press the help key. You then define a field-level transaction key for that position using either row/column specification or cursor-in-text specification.

You can also use HELP-WINDOWS to build a help menu through the use of nested help displays. The procedure for this is described fully in DEFINING A HELP MENU in section G5 of the User's Guide.

In general, you create a text record which will be your menu display, then define help definitions which are associated with a word in each of the selection prompts. You can use auto-define for these definitions but you will need to supply the transaction code of the dummy transaction (see *USING DUMMY TRANSACTIONS*, on the next page) by typing it at the main AUTODEF

display. Each transaction key either references structured text or fast-path text to be associated with this menu selection. The text may be displayed in full-screen or window mode, at your discretion.

Now you must provide a method for the operator to get to this. You could associate the help menu with some other user menu that the operator always sees, or you could associate it with a transaction screen. This would be done through the normal means, that is, defining a help definition to be recognized on a user transaction or screen, with the text pointing to the help menu text record.

## **USING DUMMY TRANSACTIONS**

Still another method is to use "dummy" transactions. This method provides a way to get directly to a help menu or directly to the text from a clear screen.

Since HELP-WINDOWS will recognize any text on the screen and can associate that with a help display, it is not necessary that the operator be in a real transaction in order to retrieve a help display. You can define a help definition and say that the transaction ID is anything, whether it is a real transaction or not.

For instance, you could define a transaction key with the transaction ID of HELP. HELP is not necessarily defined to CICS as a valid transaction code. From a clear screen, if you enter HELP, then press the help key (don't press ENTER), HELP-WINDOWS will retrieve and display the first page of text associated with that transaction. HELP-WINDOWS does not know or care whether it is a real transaction or not.

Your dummy transaction does not need to point to a menu; it could point directly to the help text. Thus, you could define a series of dummy transactions to invoke different on-line documentation displays. All the end-user needs to know in order to view the displays is what transaction to enter.

## **ACCESSING THE ON-LINE DOCUMENT**

For the end-user, the method you choose for providing access to on-line documents makes a difference as far as the procedure that they use. The available choices and the user's procedures are as follows:

- 1) Using another program to provide a help menu:

With this method, the operator will retrieve the help menu by entering some transaction code which invokes a program to display the menu. The operator will then tab to the desired selection and press the help key. This will produce the associated text.

- 2) Using a HELP-WINDOWS menu which is connected to a user menu or user application screen:

With this method, the operator will retrieve the user menu or screen by entering some transaction code which invokes a program. The operator will then tab to the desired selection and press the help key. This will produce a help menu which was created with HELP-WINDOWS. The operator will then tab to the desired selection and press the ENTER key, because we are now using nested help text. This will produce the associated text.

- 3) Using a HELP-WINDOWS menu which is invoked from a dummy transaction:

With this method, the operator will retrieve a help menu which was created with HELP-WINDOWS by entering a dummy transaction code and pressing the help key. The operator will then tab to the desired selection and press the ENTER key, because we are now using nested help text. This will produce the associated text.

4) Accessing the text directly from a dummy transaction:

With this method, the operator will enter a dummy transaction code and press the help key. This will produce the associated text.

## USING THE CUT/PASTE FEATURE

The cut and paste function is a special feature of HELP-WINDOWS which allows the operator to copy text from the help display into the application screen.

A good use of this feature is when the text in the help window describes all of the valid codes for a given field on the application screen. The operator can place the cursor in the field, press the help key to see all of the valid codes, then, rather than exiting help mode and keying the code in the field, simply place the cursor on the desired code and press the cut/paste key. This will cause the word where the cursor was placed to be copied from the help text and placed into the application field where the cursor was positioned when the help key was originally pressed.

When the word is pasted into the application field, the modified-data-tag for the field is turned on, as it would be if the operator had keyed the word. Thus, when ENTER or a PF key is pressed in the application, the pasted word will transmit to the application program just as if it were keyed.

When defining a help display, you do not need to do anything to enable the cut/paste feature to be used. It will work with any help display. It may be helpful, however, to place unprotected attributes before each word or value which would logically be eligible for a cut/paste operation, thereby allowing the operator to use the TAB key to easily position to the desired entry.

The following limitations and rules apply to the cut/paste feature of HELP-WINDOWS:

- 1) You can only paste into an unprotected field of the application display.
- 2) Only one word will be copied when the cut/paste key is pressed. The cursor may be anywhere within that word. If multiple-word copy is desired you must connect the words with a hyphen or underscore or some other character so that HELP-WINDOWS will consider it to be all one word.
- 3) If the word being pasted is longer than the receiving field, characters will be truncated on the right until the word fits the field.
- 4) If the word being pasted is shorter than the receiving field, no additional characters are padded on the end.

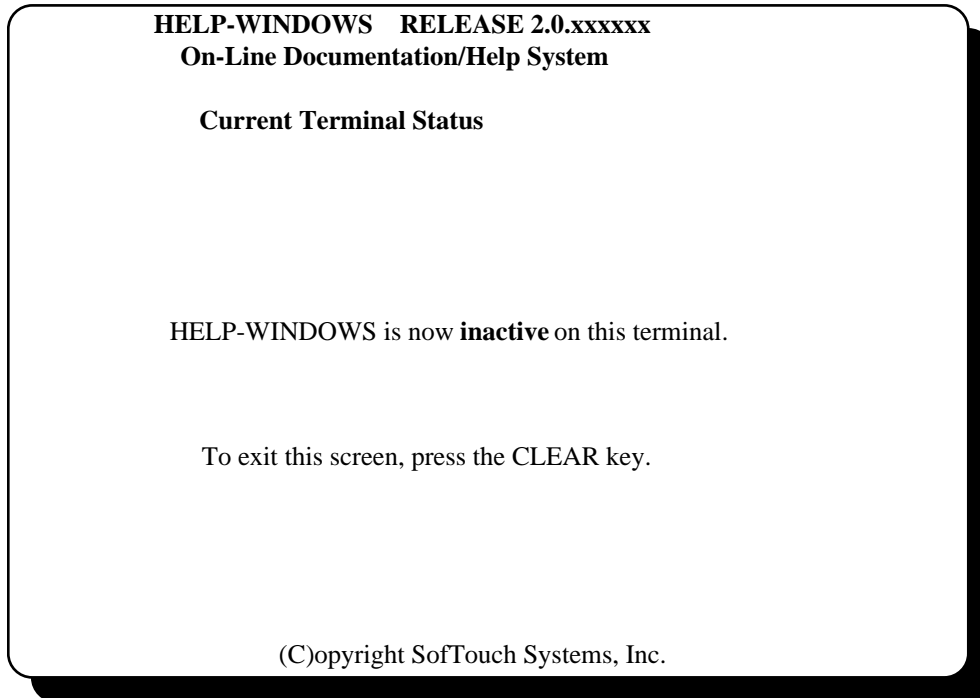


## DEACTIVATING HELP ACCESS MODE AT A TERMINAL

When you want to turn help access mode off for any reason, enter the OFF command, as

```
HWIX,OFF
```

This will result in the following :



After entering the OFF command, the terminal is no longer in help access mode. The help key, if pressed, will be passed to whatever transaction is in progress and will no longer invoke HELP-WINDOWS.

## DISPLAYING THE CURRENT TERMINAL STATUS

If you are not sure whether help access mode is active at your terminal or not, you can inquire about the current status by entering the STAT command. It is entered as follows:

```
HWIX,STAT
```

The STAT command will produce a status display showing the status of help access mode, either active or inactive, along with the default help key if it is active.



## **PLANNING FOR THE USE OF HELP-WINDOWS**

This section is designed to provide helpful insight into the preparation for developing an effective and manageable on-line documentation and/or help system using HELP-WINDOWS. In addition, we will attempt to guide you in the planning stage, where you decide what sort of documentation system to implement, who will do the work, and what subjects of the reference manuals must be understood.

HELP-WINDOWS is a highly flexible product with many options in the way that it can be used. In order to implement an orderly and manageable documentation system, we recommend that you do some up-front planning, adopt some standards for use in your system, then proceed with a structured approach.

This structured approach can be very simple or highly detailed, depending upon the extent of your documentation needs and the complexity of your on-line application systems. We will examine some typical scenarios in people's approach to documentation and try to denote the level of knowledge required of the product in order to satisfy each scenario. That is, we will point out the various features of the product that would be used for each scenario and note the locations in the reference manual where those features are discussed.

Finally, as in everything, of the various scenarios that we present, probably none will be an exact fit for your situation. You must decide what your needs really are, then pick the "best-fit" scenario to meet those needs. Keep in mind also that we at Unicom Systems are available at a phone call to assist in this planning in any way that we can. Also, we can arrange a one or two-day training session for your people if you decide that it is needed. For these training sessions, there is no charge if your people come to our site. If you need us to come to your site we must charge for this service.

## **TYPICAL SCENARIOS IN DOCUMENTATION PLANNING;**

When data processors approach an on-line documentation product, they have various needs in mind. Typically, these needs fall into one of three categories, as follows:

- 1) They are only interested in on-line help displays for their user transactions.
- 2) They are only interested in creating and maintaining procedural manuals in some centralized fashion.
- 3) They want both on-line help and procedural document maintenance.

We will examine each of these three major scenarios and present different approaches to each. The discussion is presented in dialog form, where the statements or questions in quotation marks represent statements or questions from someone who is considering the HELP-WINDOWS product.

### **.c.SCENARIO 1 - "ALL WE NEED IS ON-LINE HELP DISPLAYS";**

The on-line help display feature of HELP-WINDOWS represents roughly one half of the total package. Providing on-line help displays means that a terminal

operator who is working in one of your application transactions, can press a PF or PA key and receive a display, either a full-screen display or a pop-up window, which provides the operator with instructions for operating that transaction correctly.

On-line help displays can be created for individual fields of a display screen, where the position of the cursor on the screen determines what help display will be shown, or they can be created to relate to a particular screen display within a transaction, no matter where the cursor is, or finally they may be created to apply to any screen of the transaction, regardless of the cursor position. Thus, you can have globally defined instructions for operating the transaction, screen-oriented instructions for specific functions of a transaction, and field-oriented instructions for individual data field entry.

The pop-up windows can be created to display at any position on the screen, and the size of the window is entirely up to the person who creates the help display. You may choose to have all windows be the same size and position, or every window can be different. You can have more than one window on the screen at the same time. It is possible to build any number of cascading, overlapping windows that display in different positions on the screen as the operator proceeds through the help text.

The instructions (help text) that display in a window or full-screen display can be created in various ways, all the way from keying it directly into the window to building elaborate chains of text that provide various paths through the documentation. You can import pre-existing help text from some other system into HELP-WINDOWS. You can have help within help. That is, you can specify certain words of the documentation where the operator can position the cursor and retrieve a more detailed explanation of that particular word or phrase. This can take the form of a help menu, if desired, where the first thing the operator sees is a menu of help topics, whereupon they pick the desired topic and display it. One menu can produce still another menu, if desired. There is no limit to the level of nesting within a help chain.

As you can see, the creation of help windows can be done very simply or can be quite complex. The choice is up to you, but as in everything in data processing, the more complex a thing becomes, the harder it is to manage. Therefore, let's address the question of who will create the help displays.

### **.c."WHO IS GOING TO CREATE THE HELP DISPLAYS?"**

HELP-WINDOWS is nothing more than a tool which allows someone in your installation to build a help system. Somebody has to do the actual work of deciding what needs to be documented, then do it. It follows that this needs to be someone who is familiar with the application transactions that are to be documented. Lets examine who that might be.

### **.c.LET THE END USERS DO IT**

This can be a workable approach. You can create some on-line help displays with a bare minimum of working knowledge of the HELP-WINDOWS product and even less knowledge of data processing and technical ramifications. By using a feature of HELP-WINDOWS called 'auto-define', an end user can enter their transaction, position the cursor to the field to be documented, press a pre-defined 'help-definition' key and a window will pop up which leads them through the process of creating a help display. The help text can be keyed directly into the window, requiring no knowledge of structured hierarchies or organizational techniques. Using this method, there is an absolute minimum of control records created in the text file, and naming conventions are not really necessary, since auto-define will provide a unique name for each control record created.

For an example of using this approach, take a look at the third exercise in *GETTING STARTED WITH HELP-WINDOWS* (section L1 of this manual), called *CREATING A HELP WINDOW USING THE FAST-PATH METHOD*. As you can see, the process is really quite simple.

If you choose to pursue this approach, your end-users need to be familiar with the auto-define process, which is described in the User's Guide under *SETTING UP HELP WITH AUTO-DEFINE* (Section G5). You will still need at least one person in the organization who is relatively technical and has a good understanding of the structure of the transaction key record and the operation of the auto-define process. The auto-define/fast-path process is not foolproof and it is very possible for a non-technical person to go astray and either create unnecessary records or destroy good data. The technician in this scenario should be familiar with sections G4, G5 and G7 of the User's Guide. It would also be helpful to understand the workings of the text editor, as described in section G2.

### .c.HAVE THE APPLICATION PROGRAMMER DO IT

This is usually the most ideal approach. The programmer who developed or maintains the application system is the one who usually knows the most as to its inner workings. He or she knows all of the options available for every data field of each screen, and, being of a technical persuasion, is better equipped to take advantage of the various choices in HELP-WINDOWS to construct an easily manageable documentation system.

### .c.HAVE AN ENTRY-LEVEL PROGRAMMER DO IT

For the on-line help only scenario, this is usually the most logical approach. It is usually the case that the programmer who developed or maintains the application system is busy with another project which is too important to interrupt. It's often not the best use of his or her time to be doing documentation.

HELP-WINDOWS is not so technical that an entry-level programmer would have trouble mastering its use. This person could spend some time with the application programmer learning the operation of the application system, then proceed to build the help displays. It will be good experience for the entry-level programmer, both from the standpoint of getting familiar with the application system and of working closely with the end users.

This person will need a good understanding of the structure of the transaction key record and the operation of the auto-define process. The technician in this scenario should be familiar with sections G5 and G7 of the User's Guide, and section L2 of this manual. It would also be helpful to understand the workings of the text editor, as described in section G2 of the User's Guide, the various types of directory displays, described in section G4 and at least a perfunctory acquaintance with the various commands in HELP-WINDOWS, described in section G6. He or she should also be able to run two of the batch reports which will be helpful. These are the directory report and the cross-reference report and are described in section G8.

### **.c."HOW SHOULD THE DOCUMENTATION BE BUILT?"**

Once you have decided who is to do the actual construction of the help displays, the next choice to be made concerns the method used to build the displays. There are two methods of creating help displays in HELP-WINDOWS. You can use either or both, as you choose. These two methods are:

- 1) Fast-path help text.
- 2) Normal structured help text.

Both methods are described in detail in the section G5 of the User's Guide. The choice of which method to use is really a choice of how much wording you want to associate with a field or screen. Fast-path help text should usually be used if the display contains no more than one screen-full of help text. If you have a lot to say, normal structured text should be used. Also, if you choose to have the end-users create their own help displays, the fast-path method should probably be used, as it is quicker and requires less technical understanding.

To understand the difference in the two methods, you first need to understand the workings of a special record in HELP-WINDOWS, called the transaction key record. This record is the "bridge" between the application display and the help text to be associated with it. The transaction key record describes whether the display is to be full-screen or in a window, whether it applies to the transaction, the screen or a field on the screen, the position and size of the window (if used), and the method of identifying the field or screen where this help text applies.



In addition, the transaction key record identifies the actual text to be displayed when the operator presses the help key for this field, screen or transaction. Here is where the difference in the two methods appears, as follows:

#### .c.FAST-PATH HELP TEXT

Fast-path help text is attached directly to the transaction key record. It must be keyed (entered) during the process of building the transaction key record, either using auto-define or transaction key maintenance. The text can be keyed directly into the window or full-screen display. There is a limit of one screen's worth of text that can be attached to a transaction key record. If you need more text than that, you must create a second transaction key record, using a technique called "chaining".

Fast-path help text is designed to be used strictly for help displays. It cannot be printed, using the batch print program, nor can you import text from another system to build it.

#### .c.STRUCTURED HELP TEXT

Structured text is created using the same technique that would be used to build a procedural manual. That is, it must have what we call a "structured hierarchy". This structured hierarchy is a method of organizing text into systems, documents within systems, sections within documents and optionally subjects within sections. You create it by adding a system record, a document record, a section record and optionally a subject record. The text is then the final level of the hierarchy.

Finally, once you have a hierarchy in place, you create another type of record called a "cross-document field" record. This record is nothing more than a pointer to a section of text, by identifying the names of the hierarchy to which it belongs. In the transaction key record, you add the name of the cross-document field record. This identifies the help text using the normal structured approach. Note that you can create the cross-document field record automatically while entering the help text, by using special text commands called 'STRTFD' and 'STOPFD'.

There is no limit to the amount of text that can be associated with a transaction display using the normal structured text approach. It also has the advantage that it can be printed, exported and imported and included in other displays. It takes a little more time to set up, and requires some standards as far as naming conventions, etc., but is more flexible to use and maintain than fast-path text.

#### **.c."WHAT DO I NEED TO READ?"**

If all you are building is on-line help displays, and you want to use only the fast-path text method, the topic in section G5 called *THE AUTO-DEFINE PROCESS - FAST-PATH METHOD* will probably give you sufficient information. You should also read section L2 of this manual (*LEARNING HELP-WINDOWS*) for an understanding of the operations that the end-users can do while in help-access mode.

If you choose to use normal structured help text, you need to have a working knowledge of the creation and maintenance of a structured hierarchy, the use of the text editor, the directory displays, the auto-define process using structured help text, and the operations that the end-users can do while in help-access mode. You should read:

- TEXT FILE MAINTENANCE (section G2)
- USING DIRECTORIES (Section G3)
- UNDERSTANDING ON-LINE HELP DISPLAYS (Section G4)
- SETTING UP HELP WITH AUTO-DEFINE (Section G5)
- HELP ACCESS MODE - USING ON-LINE HELP DISPLAYS (section L2)

It will probably also be helpful to be acquainted with *COMMANDS AND KEYWORDS IN HELP-WINDOWS* and *BATCH PROCESSING* (sections G6 and G8).

## **SCENARIO 2 - "ALL WE NEED IS ON-LINE DOCUMENTS";**

The on-line documentation feature of HELP-WINDOWS allows you to create and maintain full documentation procedural manuals, organized in a way that makes them easy to manage. Your documentation is organized into a 'structured hierarchy' as described earlier in 'normal structured help text'. This structured hierarchy is a method of organizing text into systems, documents within systems, sections within documents and optionally subjects within sections. The actual documentation text becomes the lowest level of the hierarchy.

The documentation text is created and maintained using the HELP-WINDOWS text editor. This is a full-screen editor which allows you to add and delete lines of text, copy and move lines of text, include and insert text, etc. It performs very much like a programmer text editor such as TSO or ICCF. It does not have all of the features of a word processor, such as shifting words off of a line, spell checking, etc. You can do word insertion and deletion by using the insert/delete keys of the terminal.

Modular construction is made possible with the INCLUDE statement. This means that common sections of text can be inserted into multiple documents without duplicating it. This feature makes both the creation and maintenance of common text much easier.

Documentation organized in this fashion can be printed using a batch print program. You can use the print facility supplied with HELP-WINDOWS, which lets you print entire manuals or extracted portions, and provides various options for things like page headers, page numbers, etc. Or, if you choose, you can write your own print program, using the programmer interface supplied in HELP-WINDOWS for retrieving text.

Documentation from other systems can be easily imported into the HELP-WINDOWS text file. It is organized into the desired hierarchical structure by inserting special control statements in the text to identify the system, document, section and subject, prior to running the batch reformat program, which loads the text file.

Extracts or entire systems can be exported to a sequential file from the HELP-WINDOWS text file, to be loaded into some other system. Some people prefer to export their text into a word-processor, possibly a personal computer, do all of their text editing, than import the text back into HELP-WINDOWS.

An on-line directory is available for reviewing and locating documents. You can easily transfer from the directory display directly to the text and back again. The entire directory can be printed for permanent reference.

In addition to the print utility and the batch import/export programs, a cross-reference report is available which can be used to easily identify sections of text that are referenced in multiple places.

### **.c."WHO WILL CREATE THE DOCUMENTATION TEXT?"**

The creation and maintenance of an on-line documentation system requires someone who is at least acquainted with the technical aspects of data

processing. It does not need to be a programmer, necessarily, but should be someone familiar with batch job control, since much of the job involves running batch programs to print the text, import or export text, print directories and cross-references, etc. This person needs to have a logical orientation, have a good command of language structure and grammar, and the ability to organize information in a way that covers the subject adequately and makes it easy for an end-user to locate desired information.

Probably most important, this person must be very familiar with the application systems which are to be documented.

## **"CAN END-USERS DO IT?"**

With certain exceptions, the answer to this is usually "no". Typically, end-users do not have access to job submission in order to run the batch programs, and have little or no understanding of technical ramifications. The result could be a system which is very difficult to manage due to poor naming conventions and inadequate use of a modular structure.

It is sometimes the case, however, that one individual in an end-user department is so conversant with the on-line application that his or her knowledge is essential to developing adequate documentation. It may make sense in this situation to give that person some basic technical training on the product, along with a technician in the data processing department to call on, and give the project to them.

## **"HOW ABOUT ONE OF THE PROGRAMMERS OR ANALYSTS?"**

This is usually a workable approach, although the same difficulty may exist here as noted earlier. That is, the programmer's time may be better spent in program development or analysis. It is also sometimes the case that the programmer, while understanding the internals of the application programs very well, may have little knowledge of departmental procedures and processes. He may know, for instance, that a given field can accept one of five different valid codes, but may have no knowledge of the rules behind those codes, and why one code should be chosen over another.

## **"LET'S HIRE A CONSULTANT."**

If you're looking at a pretty extensive documentation effort, this may be your best choice. Consultants typically have a good technical background and can quickly become conversant with the product. They usually have good people skills, so they can interface well with the end-users who are "in-the-know" in order to get the answers they need. There are many consultants today who specialize in documentation projects. One of these could be an excellent choice.

## **"WE ALREADY HAVE LOTS OF DOCUMENTATION, IT'S JUST NOT ALL IN ONE PLACE"**

If this is the case, and it often is, the person you choose to build your HELP-WINDOWS documentation system may not have to know a great deal about the systems being documented. He or she needs technical knowledge in order to take advantage of the facilities in HELP-WINDOWS for collecting documentation from different media, and organizational ability, in order to put it all together into a manageable and understandable form. This could be an entry-level programmer, someone from the help desk, or even an operator with a good understanding of job control language.

## **"WHAT DO I NEED TO READ?"**

To build a manageable on-line documentation system, you need to have a working knowledge of the creation and maintenance of a structured hierarchy, the use of the text editor, the directory displays, the operations that the end-users

can do while viewing documentation, and a good understanding of the batch system in HELP-WINDOWS. You should read:

- TEXT FILE MAINTENANCE (Section G2)
- USING DIRECTORIES (Section G3)
- UNDERSTANDING ON-LINE HELP DISPLAYS (Section G4)
- SETTING UP HELP WITH AUTO-DEFINE (Section G5)
- HELP ACCESS MODE - USING ON-LINE HELP DISPLAYS (Section L2)
- DEFINING A HELP MENU in SECTION G5
- BATCH PROCESSING (Section G8)

It will probably also be helpful to be acquainted with COMMANDS AND KEYWORDS IN HELP-WINDOWS (section G6) and CUSTOMIZATION (section T2 of the *TECHNICAL REFERENCE MANUAL*).

## "WHAT ABOUT NAMING CONVENTIONS?"

Standard naming conventions are far more important when creating an on-line documentation system, than when only using the help displays. As described before, all documentation text is defined as a part of a SYSTEM, DOCUMENT, SECTION and optional SUBJECT. Each of these entities is identified by a name, assigned by the text writer. These names are 8 characters long in the case of system and document, 3 digits for section, and 16 characters long for subject. In addition, you may have some CROSS DOCUMENT records and TRANSACTION KEY records, which are also identified with a unique name.

In all cases, an optional description field is associated with the name of the record being created. We strongly recommend that you use this description field, even though it is not required. The description displays in the on-line directory, the directory print report and the cross-reference report, and can be very helpful for identifying a particular item.

Even so, for large documentation systems it is easy to generate quite a few records of various types, each of which must be identified uniquely. A convention for assigning standard names to each of these record types will go a long way towards making the system easier to manage.

There are no restrictions on the characters that can be used in names. Any alpha or numeric or special character may be used. There is also no rule against duplicating names at subordinate levels. Thus, a system called ACCOUNTS could also contain a document named ACCOUNTS. This would normally only make sense if that were the only document in that system.

You should take advantage of the hierarchical structure wherever you can. That is, instead of creating four systems called PAYABLES, RECVABLE, PAYROLL and GENLEDGE, make one system called ACCTNG, with each of the above items being documents within that system. Use of the subject identifier, while it is optional, should be used for large systems. This allows easy access to extracted portions of the entire document. You might want to prefix all subject names with 2 or 3-character identifier which associates them with the document to which they belong. Subjects in the PAYROLL document could be prefixed with PR-, for instance.

The method you choose for naming records is entirely up to you, and HELP-WINDOWS does provide several aids for easily locating information. A standard method for assigning names simply makes the job a little easier.

## SCENARIO 3 -

### "WE NEED ON-LINE DOCUMENTS AND ON-LINE HELP DISPLAYS"

For the most part, this scenario is just a combination of the previous two. However, there is one important difference.

HELP-WINDOWS was designed for use in this scenario. It was assumed that most people would want on-line help displays, but would also want the ability to create and maintain procedural manuals on-line. It follows that the text to be used for most help displays might also be a part of a larger procedural document. It would not be desirable to create this text in two different places

and forms - once in a format that would look good in a pop-up window, then again in a form that would look good on a printed page.

## **TEXT EXTRACTION AND WORD-WRAP;**

HELP-WINDOWS provides two features that eliminate the necessity of duplicating text in different formats for different presentation. The first of these is the ability to extract portions of text from a larger document for use as a help display. The second is the ability to "wrap" text in a window by breaking sentences on word boundaries when the sentence approaches the edge of the window.

The result is that you can create your procedural manual with no particular regard to how any of it will look when displayed in a window. You need only concern yourself with the layout of printed pages. You can then select appropriate portions of the text to be associated with a field on a screen by marking the beginning and ending of the text with special HELP-WINDOWS text commands. By activating the "word-wrap" feature, that text can be displayed in a window of any size and the terminal operator will not have to pan left and right in order to read the text.

## **MODULAR CONSTRUCTION;**

Another approach to creating help displays that are a part of larger procedural documents is to use modular construction. With this method, the text used for a help display can be created as a separate unit with its own hierarchy. Maybe each field description of an on-line screen could be defined as a subject within one section of a document which contains nothing but help displays.

You could then create another document, the actual procedural manual, which uses a HELP-WINDOWS text command called the INCLUDE statement to insert those help display text subjects at the appropriate points in the document. By using this method, text that may be common to multiple documents need only be created one time, then included into as many printed documents as desired, or referenced as a help display from several different screens.

## **"WHAT DO I NEED TO READ?"**

For combined on-line documents and help displays, you need to have an understanding of most of the features of the product, so reading the entire manual would be a good idea. In particular, the following subjects are required reading:

- TEXT FILE MAINTENANCE (Section G2)
- USING DIRECTORIES (Section G3)
- UNDERSTANDING ON-LINE HELP DISPLAYS (Section G4)
- SETTING UP HELP WITH AUTO-DEFINE (Section G5)
- HELP ACCESS MODE - USING ON-LINE HELP DISPLAYS (Section L2)
- DEFINING A HELP MENU in SECTION G5
- SPECIAL MAINTENANCE FUNCTIONS AND INSTRUCTIONS (Section G7)
- BATCH PROCESSING (Section G8)



It will probably also be helpful to be acquainted with **COMMANDS AND KEYWORDS IN HELP-WINDOWS** (section G6) and **CUSTOMIZATION** (section T2 of the *TECHNICAL REFERENCE MANUAL*).

Text extraction is performed by means of the **STRTFLD/STOPFLD** statements, which are described in *IDENTIFYING THE TEXT TO BE DISPLAYED* in section G5. It is also described in *CROSS-DOCUMENT RECORD MAINTENANCE* in section G7.

Modular construction and the use of the **INCLUDE** statement is described in *INSERTING TEXT RECORDS, THE INCLUDE STATEMENT*, in the topic entitled *THE TEXT RECORD* in section G2.

Word-wrap is activated when the transaction key record is created either explicitly or using auto-define. Word-wrap is described in *FIELDS OF THE TRANSACTION KEY RECORD* in the topic entitled *TRANSACTION KEY RECORD MAINTENANCE* of section G7.

## **SUMMARY OF TYPICAL SCENARIOS**

The on-line documentation system that you create with **HELP-WINDOWS** can take different forms, according to your needs. You should do some up-front planning as to exactly what you want to accomplish, then decide who will construct the system. That person should learn those parts and features of the product that will enable him or her to accomplish the plan.

## CUSTOMIZATION OPTIONS

There are several options available in HELP-WINDOWS to control or change some of the ways that the product operates. These include such things as resolving conflicts or setting standards for PF key usage within the product, setting different attribute control codes other than the default codes, using multiple text files and setting certain security options, automatic activation of help access mode, etc.

These options are all described in the section entitled *CUSTOMIZATION* of the *TECHNICAL REFERENCE MANUAL*. The choices that can be made are summarized as follows:

- 1) Define alternate names for the VSAM files used by HELP-WINDOWS.
- 2) Specify that the conversation is to be maintained by terminal ID rather than operator ID.
- 3) Establish security for certain HWIX commands.
- 4) Define different characters to be used as attribute codes.
- 5) Suppress the status display at HWIX,ON time.
- 6) Associate a different VSAM file name with an alternate transaction code.
- 7) Set the default upper/lower case option.
- 8) Assign different PF keys for the directional keys used by the product.
- 9) Define a character or string of characters to be recognized as a help code in a field, rather than using a PF or PA key as a help key.

These customization options should be examined and set so that there will be the least amount of interruption possible to normal sign-on and transaction operation procedures.

## REFERENCE GUIDE TO PRODUCT FEATURES

HELP-WINDOWS has many features, some of which you may never use. For the purpose of your planning and evaluation, listed here are all of the things that you can do with the product, with the corresponding subject title in this manual as to where each feature is described.

- **Create and maintain procedural documents and manuals.** (See section G2 - *Record Maintenance by Record Type*).
- **Maintain documentation text with a text editor.** (See *Using the Text Editor* in section G2).
- **Display text as it will appear when printed or viewed in a help window.** (See section G2 - *The Text Record, Displaying Text Records in List Mode*).
- **Associate help text with a transaction, a screen of a transaction or a field of a screen.** (See section G5 - *Setting Up Help With Auto-Define*).
- **Display help text in a window on the screen.** (See section G5 for window creation methods. See section L2 of *Learning Help-Windows* for viewing text in a window).
- **"Wrap" the text to fit in a window.** (See *Text display specifications* within *Transaction Key Maintenance* which appears in section G7).
- **Pan the text right or left in a window.** (See the *RIGHT* and *LEFT* commands in section G6).
- **Scroll forward and backward in a help display.** (See the *UP* and *DOWN* commands in section G6).
- **Search for key words or character strings in help text.** (See the *SEARCH* command in section G6).
- **Use "cut and paste" to pick up words from the help text and insert them into the transaction screen.** (See *Using the cut/paste feature* which appears in section L2 of *Learning Help-Windows*).
- **Associate a help display with field titles or text on the user screen, rather than using absolute row/column address.** (See *Defining Text-Identified Field Help* in section G5).
- **Use extracted portions of a larger document for help displays.** (See *Identifying the Text to be Displayed*, in section G5).
- **Associate help text with a word or phrase within other help text, thereby creating multiple levels of help.** (See *Help displays within help text (nested help)* which appears in section L2 of *Learning Help-Windows*).
- **Create and use help menus for quick access to various sections of help text.** (See *Defining a help menu*, Section G5).
- **"Chain" sections of help text together in various ways to create multiple paths through the documentation.** (See *Help screen chaining* which appears in section L2 of *Learning Help-Windows*).
- **Create multi-window displays. Any number of windows may overlap in any configuration desired.** (See *Text display specifications* within *Transaction Key Maintenance*, section G7. Also see *Chained help windows*, in section L2 of *Learning Help-Windows*).
- **Use a "dummy" transaction to initiate a help display.** (See *Using dummy transactions* in *Defining a help menu*, section G5. Also see *Viewing on-line documentation without using HWIN*, section L2 of *Learning Help-Windows*).
- **Create help displays by invoking a user transaction screen, positioning the cursor as desired and pressing a "help definition" key.** (See section G5 - *Setting Up Help With Auto-Define*).
- **Use different colors and extended highlighting in text displays.** (See Section G2 - *The Text Record, Creating different colors in help displays*).
- **Create documentation text for large-screen terminals.** (See the *ALTSCRN* command in section G6).
- **Enter documentation text in lowercase or uppercase mode. Control the setting of the uppercase translation feature of CICS.** (See the *UC* command in section G6. Also see the *CASE* keyword in section T2 of the *Technical Reference* manual).
- **Construct documentation in modular form, inserting pre-defined modules into other documents.** (See section G2 - *The Text Record, Inserting Text Records, the INCLUDE statement*).

- **Easily locate documents and text using the on-line directory.** (See section G3 - *Using Directories*, also see *Creating and Using a Table of Contents* in section G7).
- **Define a single "hot" key to be used as the help-retrieval key for all transactions, or define a different help key for individual transactions.** (See *Text display specifications* within *Transaction Key Maintenance*, section G7).
- **Define "cross-document" fields.** Fields that appear on multiple screens or multiple transactions, always referencing a common set of help text. (See *Cross-Document Record Maintenance*, in section G7).
- **Create a table of contents to be printed with the document.** (See *Creating and Using a Table of Contents* in section G7).
- **Display the table of contents on-line, for quick access to various parts of the help text.** (See *Creating and Using a Table of Contents* in section G7).
- **Print documents or selections from a document. Add page headers, automatic page numbering and other formatting options.** (See *Printing documentation* in section G8).
- **Print the text directory and cross-reference reports.** (See *Printing documentation* in section G8).
- **Capture a user screen display and store it as text in a document.** (See *Saving a User Screen Display as Text Data* in section G5).
- **Create an index for alphabetical reference to topics.** (See *Creating and Using an Index* in section G7).
- **Import text from some other file into the HELP-WINDOWS text file.** (See *Reformatting existing text*, in section G8).
- **Export HELP-WINDOWS text to a sequential file to be processed by another system.** (See *Exporting the text file*, in section G8).
- **Control or change various operational options in HELP-WINDOWS.** (See *Customization* in section T2 of the *Technical Reference Manual*).
- **Set security options at the system or document level, allowing only authorized personnel to view, update, add or delete text.** (See *Assigning security to documentation text* in section T2 of the *Technical Reference Manual*).
- **Automatically pre-activate HELP-WINDOWS for all or selected users.** (See *Automatic Activation of Help Access Mode* in section T2 of the *Technical Reference Manual*).

## A

activation  
  of help mode, 61  
active key fields  
  of selection menu, 6  
attribute code  
  in help menus, 77  
auto-define  
  changing the window, 36,  
  46, 54  
  connecting help text, 33  
  entering, 26  
  exiting, 33, 41, 48  
  fast-path method, 50  
  field window, 31, 35, 53  
  initial display, 29, 35  
  keying fast-path text, 57

## C

chaining  
  of help displays, 73  
changing the window  
  methods of, 37  
command codes  
  ADD, 7  
  AUTODEF, 31, 36, 53  
  COPY, 24  
  CUT, 67, 80  
  DEFINE, 26  
  DELETE, 24  
  DOWN, 68  
  EDIT, 13  
  EDIT TRAN, 56  
  LEFT, 68  
  NEXT, 15, 43  
  OFF, 81  
  ON, 27, 62  
  PASTE, 67, 80  
  PREV, 15, 43  
  QUIT, 15  
  RENUMBER, 13  
  RIGHT, 68  
  SEARCH, 69  
  STAT, 50, 81  
  STOPFLD, 18  
  STRTFD, 18  
  UP, 69  
command field  
  copying lines, 16  
  in auto-define, 31, 36, 53  
  line deletion, 15  
  line insertion, 15

  moving line, 16  
commands  
  in help mode, 68  
  window adjustment, 37  
conversation  
  with selection menu, 7  
copy line command  
  in text edit, 16  
cursor  
  in help menus, 77  
  updating text, 13, 15  
  with multiple windows, 75  
cut and paste  
  **description of**, 80  
  limitations of, 80  
  use of, 67  
CUT command  
  **description of**, 80  
  use of, 67

## D

DEL KEY  
  IN TEXT EDIT, 15  
DEL key  
  in text edit, 13  
deletion  
  global, 23  
  text line, 15  
directional options  
  in help mode, 67  
document record  
  addition of, 8  
DOWN command  
  description of, 68  
dummy transactions  
  description of, 78  
  illustration of, 78

## E

EDIT mode  
  description of, 13  
editor  
  operation of, 15, 16  
end user  
  document viewing, 77  
  use of HELP-WINDOWS,  
  61  
ERASE EOF KEY  
  IN TEXT EDIT, 15  
ERASE EOF key  
  in text edit, 13  
exercises  
  copying pre-keyed text, 22

  creating a structured  
  document, 6  
  creating fast-path help  
  display, 50  
  creating transaction level  
  help, 44  
  keying text, 17  
  structured help display, 26  
  summary of, 60  
exit processor  
  description of, 61

## F

fast-path  
  text creation. (see fast-path  
  help text)  
fast-path help text  
  advantage of, 50  
  description of, 3, 50  
  disadvantage of, 50  
  keying of, 56  
  using in help display, 50  
field level help  
  creating, 29, 35  
  description of, 29

## H

HDMO  
  description of, 17  
  use of, 29  
help access key  
  defining, 27, 28  
Help Definition key  
  defining, 27  
help displays  
  chaining, 73  
  exiting from, 77  
  help menus, 77  
  levels of, 29  
  retrieval of  
    field level, 72  
    screen level, 71  
    transaction level, 65  
help displays in action  
  illustration of, 65  
help key  
  default  
    defining, 63  
    retrieving help display, 66  
help menu  
  accessing of, 78  
  creation of, 77  
  defining, 77

- help mode
  - description of, 61
- help text
  - types of, 2
- hierarchy
  - creation of
    - document record, 8
    - section record, 10
    - subject record, 11
    - system record, 7
    - text record, 12
  - description of, 6
- hot keys
  - establishing, 26
- HWIN
  - description of, 4
- HWIX transaction
  - activating auto-define, 26, 27
  - activating help mode, 61

## I

- INS KEY
  - IN TEXT EDIT, 15
- INS key
  - in text edit, 13

## K

- key fields
  - of selection menu, 6
- keys

help access  
  defining, 27  
help definition  
  defining, 26, 50

## L

LEFT command  
  description of, 68  
line  
  copying, 16  
  deletion, 15  
  editing, 13, 15  
  insertion, 15  
  insertion:, 15  
  moving, 16

## M

menu  
  response to, 7  
move line command  
  in text edit, 16  
multiple windows  
  description of, 73  
  example of, 73  
  exiting, 75

## N

nested help  
  description of, 75  
  example of, 75  
  exiting, 77  
  retrieval of, 75  
NEXT command  
  with chained help displays,  
  73

## O

OFF command  
  description of, 81  
ON command  
  description of, 62

## P

panning  
  elimination of, 67  
  in help mode, 67  
panning  
  left, 67  
  right, 67  
PASTE command  
  **description of, 80**

  use of, 67  
PF keys  
  changing the window, 37  
  directional options  
    help mode, 67  
  used as help access key, 63  
  used as help definition  
    key, 27  
PREVIOUS command  
  with multiple windows, 75

## R

RIGHT command  
  description of, 68

## S

screen  
  help mode activation, 63  
  help window, 66  
  status display, 64  
screen displays  
  activation of help access,  
  28  
  AUTODEF window  
    change, 36, 47  
  completed help display, 34  
  document record, 8  
  HDMO screen, 29  
  help key status, 27, 28  
  help-window, 71  
  main AUTODEF display,  
  31, 35  
  multiple windows, 74  
  nested help, 76  
  pre-keyed text, 18  
  section record, 10  
  status display, 81  
  subject record, 12  
  system record, 7  
  text record, 13  
screen level help  
  description of, 29  
screendisplays  
  auto-define activation, 26  
scrolling  
  in help mode, 68, 69  
SEARCH command  
  backward search, 69  
  continuing, 69  
  description of, 69  
  format of, 69  
  in help mode, 69  
section record  
  addition of, 10

STAT command  
  description of, 81  
STOPFLD statement  
  use of, 18  
STRTFLD statement  
  use of, 18  
structured help text  
  copying, 24  
  creation of, 12  
  deleting, 24  
  editing, 15  
  format of, 13  
  identifying for help  
    display, 18  
  list mode display, 22  
  using in help display, 26  
structured hierarchy. (see  
  hierarchy)  
subject record  
  addition of, 11  
system record  
  addition of, 7  
  fields of, 8

## T

text  
  fast-path  
    advantage of, 50  
    description of, 3, 50  
    disadvantage of, 50  
    keying of, 56  
    using in help display, 50  
  structured  
    copying, 24  
    creation of, 12  
    deleting, 24  
    editing, 15  
    format of, 13  
    identifying for help  
      display, 18  
    list mode display, 22  
    using in help display, 26  
  types of, 2  
text maintenance transaction.  
  (see HWIN)  
text record  
  addition of, 12, 13  
  locating, 15  
  update of, 13, 15  
transaction code  
  dummy, 78  
  exit processor, 61  
transaction key record  
  chaining, 73  
transaction level help

creating, 44  
defining, 46  
description of, 29  
reasons for use, 44

## U

UP command  
description of, 69

## W

word-wrap  
description of, 67