

About This Manual

Purpose

This manual provides the information you need to install CANDE and use its commands and utilities.

Scope

This manual contains all the information you need to use the CANDE product, including

- Syntax and examples for CANDE commands and View File utility commands
- Complete information for using utilities that let you view copy, print, and merge files
- Error messages and corrective action

This manual does not provide information about specific compilers or programming languages or MCP operations.

Audience

This manual can be used by any level of programmer.

Prerequisites

Before using CANDE, you should be familiar with the compilers and programming languages you intend to use with the CANDE editor.

How to Use This Manual

Before using CANDE, you should first read Section 1 of this manual, which provides general information about the CANDE product, including

- A description of the software
- The CANDE file-naming conventions
- Instructions for starting and terminating a session
- Information on file recovery and program dumps

Sections 2 and 3 describe all the commands you can use with CANDE and the View File utility.

The remaining sections explain how to modify the CANFMT file and use the CANDE utilities.

Error messages for CANDE and the View File utility are provided in Appendix A.

Instructions for installing CANDE are provided in Appendix B.

Organization

This manual consists of seven sections, three appendixes, and an index. The content of the sections and appendixes is described as follows:

Section 1. Introduction

This section describes CANDE, its major functional features, and the files that constitute the CANDE software. This section also discusses CANDE file-naming conventions, explains how to invoke and terminate a CANDE session, and describes file recovery.

Section 2. CANDE Commands

This section presents all of the CANDE commands in alphabetical order and provides the syntax, explanation, and examples for each command.

Section 3. Printing and Viewing Files

This section provides the syntax, explanation, and examples for the commands you can use with the View File utility.

Section 4. Creating Batch Files

This section explains how to create a file of commands, called a *batch file*, that can be executed by issuing a single command from your keyboard. This section also includes information on creating special batch files that get executed automatically at the beginning of a CANDE session.

Section 5. Using the CANUTL Utility

This section explains how to use the CANUTL utility which performs a variety of file conversion, comparison, copy, print, rebuild, and reorganize operations.

Section 6. Understanding and Modifying the CANDE Format File

This section describes the CANFMT file, which contains the text for error messages and help responses, and formats for printer backup files. This section also explains how you can customize the text in the CANFMT file and add workstation tab settings.

Section 7. Merging Editor and System Files

This section explains how to use the patch merge utility to merge multiple files into one file.

Appendix A. Error Messages

This appendix lists and explains the error messages you might encounter when using CANDE or one of its utilities. The action you should take in response to a message is provided where appropriate.

Appendix B. Installing CANDE

This appendix provides step-by-step instructions for installing CANDE.

Appendix C. Understanding Railroad Diagrams

This appendix explains how to read the railroad diagram syntax used with all of the commands in this manual.

Related Product Information

The following manuals contain additional material on topics discussed in this manual.

V Series Systems Operations Guide, Volume 1: Installation (4127 4804)

This volume of the *Systems Operations Guide* contains introductory material about the MCP/VS operating system and details the procedures and options available for installing and initializing MCP/VS on V Series systems.

V Series Systems Operations Guide, Volume 2: System Commands (4127 4994)

This volume of the *Systems Operations Guide* provides detailed information about the use and syntax for each system command. It also includes some basic information for the first-time user and reference material for the experienced user.

V Series Systems Operations Guide, Volume 3: System Utilities (4127 0000)

This volume of the *Systems Operations Guide* describes the MCP/VS utilities, which act as extensions of the Master Control Program (MCP) for such functions as converting files from one medium to another, initializing diskpack units, or creating pseudo card decks.

V Series Systems Operations Guide, Volume 4: System Messages and Recovery (4127 0018)

This volume of the *Systems Operations Guide* is a reference to the system messages that appear on the Operator Display Terminal (ODT) to either inform you of a process taking place or of an error condition. Recovery procedures are also included.

***V Series System Security Installation and Operations Reference Manual
(5031511)***

This guide explains how to install and use the system security facilities for the Unisys V Series systems to control access to the system and to files.

***B 2000/B 3000/B 4000/V Series GEMCOS Implementation and Programming
Reference Manual (5026313)***

This manual provides the features, the methods of application, and the utilities of the Generalized Message Control System (GEMCOS).

***V Series VCS Implementation Reference Manual, Volume 1: Entities and
Screens (4127 0059)***

This manual describes all the entities that constitute the VCS network and explains how to configure each entity.

***V Series VCS Implementation Reference Manual Volume 2: System
Transactions (4127 0067)***

This manual provides the syntax and explanation of all the commands you can use with VCS.

Contents

About This Manual	v
Section 1. Introduction	
What Is CANDE?	1-1
CANDE Software	1-2
CANDE File-Naming Conventions	1-3
Invoking CANDE	1-4
CANDE Help Facility	1-4
Terminating a CANDE Session	1-5
Editor File Recovery	1-5
Program Dumps from Internal Errors	1-5
Section 2. CANDE Commands	
Common Syntax Elements for CANDE Commands	2-1
File Title	2-1
Sequence Range	2-3
ABORT Command	2-4
ADM (Auto Display Mode) Command	2-5
APPEND Command	2-6
BACK Command	2-7
BLOCK Command	2-8
BOUND Command	2-9
BYE Command	2-10
CANUTL Command	2-11
CATALOG Command	2-12
CHANGE Command	2-15
COMMENT Command	2-16
COMPILE Command	2-16
COPY Command	2-17
CUT Command	2-19
DEBUG Command	2-19
DELETE Command	2-20
DETANK Command	2-20
DO Command	2-21
DUPLICATE Command	2-22
EMODE Command	2-23
END Command	2-24
FAMILY Command	2-24
FILES Command	2-26

Contents

FIND Command	2-27
FIX Command	2-29
FPW (File Password) Command	2-31
GET Command	2-32
HARDCOPY Command	2-33
HELP Command	2-34
JMGR (Job Manager) Command	2-35
LAST Command	2-36
LESN (Last Edited Sequence Number) Command	2-36
LIST Command	2-37
MAKE Command	2-39
MARK Command	2-42
MERGE Command	2-43
MOVE Command	2-44
NEXT Command	2-45
PAGE Command	2-46
PASTE Command	2-48
PATCHID Command	2-49
PCR (Pseudo Card Reader) Command	2-50
PLOG (Print Log) Command	2-51
PREVIOUS Command	2-51
PSCREEN (Print Screen) Command	2-53
PURGE Command	2-53
QUIT Command	2-54
REBUILD Command	2-55
RECALL Command	2-55
REDO Command	2-56
RELEASE Command	2-56
REMOVE Command	2-57
REORGANIZE Command	2-58
REPLACE Command	2-59
RESEQUENCE Command	2-61
RLOG (run log) Command	2-62
RMERGE Command	2-65
RUN Command	2-65
SAME Command	2-66
SAVE Command	2-67
SEARCH Command	2-67
SECURITY Command	2-69
SEQUENCE Command	2-70
SESSION Command	2-72
SLOG (ODT Log) Command	2-76
START Command	2-79
TABS Command	2-83
TANK Command	2-85
TYPE Command	2-85
VIEW Command	2-86
VMODE Command	2-87
VOID Command	2-88
WHATS Command	2-88
Workstation Command	2-89
-<nn> Command	2-95

+<nn> Command	2-95
---------------------	------

Section 3. Viewing and Printing Files

Starting the VF Utility	3-1
AT Command	3-2
BYE Command	3-3
CLOSE Command	3-3
FORM Command	3-3
FORMAT Command	3-4
HELP Command	3-9
LIST Command	3-9
NEXT Command	3-13
OUTPUT Command	3-14
REDO Command	3-17
REMOVE Command	3-18
SAVE Command	3-18
SESSION Command	3-18
VIEW Command	3-19
WHAT Command	3-20
WORKSTATION Command	3-21
?BRK Command	3-22
-<nn> Command	3-22
+<nn> Command	3-23

Section 4. Creating Batch Files

Batch File Contents	4-1
Special CANDE Batch File Statements	4-2
//ACCEPT Statement	4-2
//AT Statement	4-2
//BEGIN Statement	4-3
//BEGINJOB Statement	4-3
//DISPLAY Statement	4-3
//END Statement	4-4
//ELSE Statement	4-4
//ENDJOB Statement	4-4
//IF Statement	4-4
Using Labels to Execute Portions of a Batch File ..	4-6
Using Variables in Batch Files	4-6
Including V Series System Commands in a Batch File	4-7
Using Multifile Identifiers in a Batch File	4-8
Referencing Printer Files in a Batch File	4-9
Including Comments in a Batch File	4-9
Including Pseudo Card Reader Commands in a Batch File	4-10
Embedding Miscellaneous Text in a Batch File	4-10
Executing a Batch File	4-11
Executing a Batch File with the DO Command	4-11

Contents

Automatic Execution When CANDE Is Initiated	4-13
Batch File Sample	4-14
Section 5. Using the CANUTL Utility	
Executing CANUTL from CANDE	5-1
Executing CANUTL from VCS or GEMCOS	5-2
Section 6. Understanding and Modifying the CANDE Format File	
Structure of the CANFMT File	6-1
Including Custom Formats and Tab Settings in the CANFMT File	6-2
Modifying Error Messages and Help Responses	6-2
Specifying an Alternate Name for the CANFMT File	6-3
How the System Locates the CANFMT File	6-3
Section 7. Merging Editor and System Files	
Patch Merge Utility Guidelines	7-1
Patch Merge Utility Features	7-1
Files Used and Generated by the Patch Merge Utility	7-2
Providing Instructions in the SCRIPT File	7-2
Including Options in the SCRIPT File	7-2
Specifying File Titles in the SCRIPT File	7-4
Including Comments in Input Files	7-5
CANDE Menu Interface	7-5
Error and Warning Messages	7-6
Appendix A. Error Messages	
Appendix B. Installing CANDE	
Customizing a CANDE Session	B-2
Appendix C. Understanding Railroad Diagrams	
What Are Railroad Diagrams?	C-1
Following the Paths of a Railroad Diagram	C-2
Bibliography	
Index	

Figures

5-1.	CANUTL Utility Menu	5-1
7-1.	PATMRG Utility Menu	7-5



Tables

1-1.	CANDE Files and Programs	1-2
2-1.	CANDE File Kinds	2-41
2-2.	File Kinds and Comment Indicators for Source Files Used in CANDE File-Equated Execution	2-81
7-1.	Files Processed by the Patch Merge Utility	7-2

Section 1

Introduction

What Is CANDE?

Command and Edit (CANDE) is a host-based text editor that lets you generate and maintain source files in any of the supported V Series compiler file kinds, including data and sequential file kinds of up to 476 bytes per record. Because CANDE executes as a single-user application, one copy of CANDE appears in the mix for each user. CANDE is supported by the GEMCOS and VCS message control systems.

The major functional features of CANDE include the following:

- The ability to save a copy of the active editor file as a *recall* file and have the recall file updated automatically after a specified number of changes

The *active* editor file is the editor file you have most recently opened with the GET command.

A recall file is a backup copy of the active editor file that provides backup protection if the active editor file gets lost or corrupted. You can use the recall file whenever you want to revert to an earlier version of the editor file.

CANDE creates a recall file whenever you specify the RECALL option with either the GET or SESSION command. The recall file gets updated when you close a file by using the RELEASE command with the SAVE option. You can also direct the system to update the recall file automatically after a specified number of changes by using the AUTOSAVE option of the SESSION command. For details on these commands, see Section 2.

- The insertion of a patch ID in lines of the editor file that have been added or modified

You can use the PATCHID command to specify the identifier with which added or modified lines will be marked. COBOL68, COBOL74, and RPG support a patch ID field in the source text. For all other file kinds, you define the column location and field length for the patch ID by using the PATCHID command. For details about this command, see Section 2.

- System-generated sequence numbers for input source lines

Once you have established an initial sequence number, the system will generate further sequence numbers. The system determines the new sequence number by starting with the last edited sequence number (LESN) and incrementing by the value you have defined for the SEQINC option of the SESSION command. When you input new source text lines, be sure to leave blank the entire space where the sequence would normally appear. For details about the SESSION command, see Section 2.

- Page-oriented display and automatic purge after three unrecognized commands

Although the CANDE editor is page-oriented, each input line is examined and processed individually so that a single screen can contain multiple commands. When CANDE encounters three consecutive, unrecognized commands, CANDE ignores the remaining input to prevent a lengthy delay caused by the display of numerous error messages that result from erroneous input.

- The ability to create a file of CANDE commands that is automatically invoked when the user initiates a CANDE session

You can include any of the CANDE commands in a batch file. When you name the batch file *UVF* for "user variable file" or *SVF* for "system variable file," CANDE automatically executes the commands in this file when you execute CANDE. The variable file is a convenient way to set up session and workstation attributes (see the *SESSION* and *WORKSTATION* commands in Section 2). For details on user-variable and system-variable files, see Section 4.

CANDE Software

The CANDE product consists of the programs and files described in Table 1-1.

Table 1-1. CANDE Files and Programs

File Name	Type of File	Size	Description
CE1DOC	PBD		Release letter
CANDE	Code	165KD	CANDE text editor
CANFMT	Data		CANDE format, help screens, and error messages file. The use of this file is optional. If you do not use this file, the system uses the internally coded English language version of this file. You can customize this file for your language. In addition, you can include custom printer backup file formats and workstation tab settings in this file. For details on the CANFMT file, see Section 6.
CANUTL	Code	66KD	CANDE file copy utility. This utility provides numerous options for copying files. For details on using this utility, see Section 5.
PATMRG	Code	68KD	Patch file merge utility. This utility merges multiple CANDE editor or system format files. For details on using this utility, see Section 7.

continued

Table 1-1. CANDE Files and Programs (cont.)

File Name	Type of File	Size	Description
VF	Code	60KD	View File utility. This utility lets you view or print any filekind . You can use a format from the CANFMT file with this utility to alter the display and define compiler error and warning messages. You can invoke the View File utility from CANDE or from VCS or GEMCOS. For details on this utility and its commands, refer to Section 3.

CANDE File-Naming Conventions

A nonusercoded CANDE file name can consist of 1 to 6 characters and must be preceded by an asterisk (*). A usercoded file name must have exactly 4 characters. File names can consist of alphabetic characters and numbers, but the first character must be alphabetic. An example of a usercoded file and a nonusercoded file follow:

A123

*SYSFIL

During CANDE sessions, CANDE automatically appends your user ID to all usercoded file names (those not preceded by an asterisk (*). The user ID is a 2-character attribute assigned with VCS or with the USERHO utility of GEMCOS. The first character of the user ID must be a lowercase letter from *a* through *z*. The second character can be any character.

If you get or make a file using the RECALL option, CANDE creates the recall file name by placing your user ID in the fourth and sixth positions of the original file name. For example, if the file name is ABCDaa, the recall file name would be ABCaDa.

In VCS, a user is assigned a user ID through the USER-ADD or USER-MOD transaction. The screen displayed by these transactions has a field labeled EDITOR FILE USER CHARS, which defines the CANDE user ID. For details on using these transactions and this screen, refer to the *V Series VCS Implementation Reference Manual, Volume 1: Entities and Screens* and *Volume 2: System Transactions*.

In GEMCOS, a user is assigned a user ID through the <accesscode spec> parameter in the ACCESSCODES section of the input file used by the USERHO utility. For details on the USERHO utility, refer to the *V Series System Software Security Installation and Operations Guide*.

Invoking CANDE

Note: *The 3.2 release of CANDE supports either 4 numeric characters or 10 alphanumeric characters for the usercode in the <accesscode spec> variable of the USERHO utility. Earlier releases of CANDE require the usercode to be 4 numeric characters.*

Whenever you specify a CANDE usercoded file in operating system commands or in any non-CANDE syntax, you must use the complete 6-character name: the 4-character CANDE file name followed by your 2-character user ID.

Examples

A123jd

This example is a usercoded file. The system has appended the 2-character user ID, *jd*, to the end of the file name.

*SYSFL

This example is a nonusercoded file.

Invoking CANDE

Once you log on to VCS or GEMCOS, you can use CANDE if your accesscode is authorized to do so. You do not have to log on to CANDE. In addition, you do not need a separate accesscode. The accesscode that gives you access to the MCS is the accesscode that identifies you as a user to CANDE.

To invoke the CANDE program, enter the VCS transaction or GEMCOS command that has been defined to initiate the CANDE program.

For example, if you defined a VCS transaction labeled *CANDE* to start the program, you would initiate CANDE by entering

/CANDE

Using GEMCOS, you would typically use the ASSIGN or CONNECT command to invoke CANDE, for example:

*CONNECT CANDE

After initiation, CANDE displays a welcome screen, and you can begin entering commands.

CANDE Help Facility

Online help in the form of a syntax diagram and an explanation of the syntax is available for any CANDE command. To display help information for a specific command, enter *HELP* followed by the command. For example, entering the following command displays information about the GET command:

HELP GET

If you enter *HELP* without a command, a list of valid CANDE commands is displayed on the screen with instructions on how to proceed.

Terminating a CANDE Session

You can use the *BYE* or *END* command to terminate a CANDE session. CANDE waits for all active tasks to be completed before actually terminating the session. If a recall file is associated with the active editor file, CANDE saves or removes the recall file according to the value of the *AUTOREMOVE*, *AUTOSAVE*, and *RECALL* options of the *SESSION* command. See Section 2 for details on this command.

If you are using VCS version 3.1 or later, entering the VCS log-off transaction automatically terminates all copies of CANDE executing for your workstation.

Editor File Recovery

If your active editor file gets corrupted, and your session is not abnormally terminated, you can recover the file by using the *REORGANIZE* command (see Section 2).

If your CANDE session is abnormally terminated, and the active editor file gets corrupted, CANDE will attempt to reorganize the file the next time you log on to the MCS. To perform the reorganization, CANDE invokes the *CANUTL* utility.

If the reorganization is not successful, CANDE gives you the option to rebuild the file. Be aware that rebuilding a file could result in the reintroduction of logically deleted blocks into the editor file. Deleted blocks can be reintroduced because all CANDE editor deletions are logical. If an entire block of records is deleted, it is really marked for reuse.

If duplicate sequence numbers are encountered, the first duplicate is included in the rebuilt file. Subsequent duplicates are not included in the rebuilt file and instead are displayed in a printer backup file.

You might have to edit the final, rebuilt file to ensure that it contains the text and sequence numbers that you want to use.

Note: An abnormal termination of the session while you are using the *VF* utility should not adversely affect the file you are viewing.

Program Dumps from Internal Errors

If an internal CANDE error is detected, a dump is taken. A maximum of three dumps are taken during a 5-minute period. When the number of dumps exceeds three, the CANDE session is terminated. This action prevents the possibility of looping when an internal error occurs.



Section 2

CANDE Commands

This section presents all of the CANDE commands in alphabetical order and provides the syntax, explanation, and examples for each command. The minimum portion of a command or parameter that you must enter appears underlined. Where no underlining appears, you must enter the entire command or parameter. In the following command descriptions, the *active* editor file refers to the editor file most recently opened with the GET command.

Commands for the VF utility are described in Section 3.

Common Syntax Elements for CANDE Commands

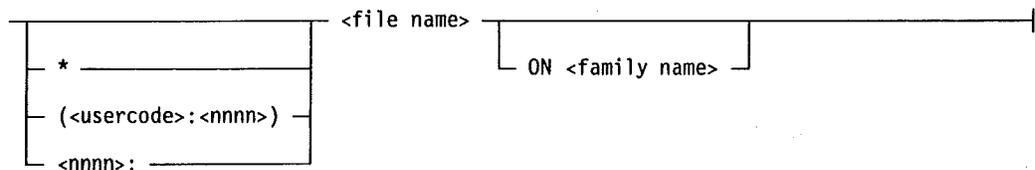
The diagrams that follow provide the syntax and explanations of elements that occur frequently in the railroad syntax diagrams in this manual. For a general discussion of how to read railroad syntax diagrams, refer to Appendix C of this manual.

To enable case-sensitivity for any element of a CANDE command, enclose the element within parentheses or double quotation marks ("). Text marked in this way is case-sensitive regardless of the value of the MAPINPUT option of the WORKSTATION command.

File Title

This syntax is used to select a file for an operation.

Syntax



Explanation

* An asterisk in front of a file name indicates that the file is nonusercoded. CANDE does not append the 2-character user ID to the file name. A nonusercoded

Common Syntax Elements for CANDE Commands

	file name can have from 1 to 6 alphanumeric characters. The first character must be alphabetic.
<usercode>	This is a 1- to 10- character label that identifies the ownership of files.
:<nnnn>	This specifies the number of the VCS security installation in which CANDE will search for the usercode. This specification is necessary because the same usercode can reside in several VCS installations.
<nnnn>:	This is a numerical user ID used in previous versions of CANDE.
<file name>	This specifies the name of a file. The file name must begin with an alphabetic character. Nonusercoded file names can have from 1 to 6 characters and must be preceded by an asterisk (*). Usercoded file names must consist of 4 characters. CANDE automatically appends the 2-character user ID to the end of a usercoded file name, making a 6-character file name. (The 2-character user ID is assigned on the VCS Users screen or with the USERHO utility.) By default, the file name is assumed to be a usercoded file.
ON <family name>	This is the name of the disk or disk pack family (group of associated disks) on which the file is stored.

Examples

(PAPABEAR)ABCD

This example specifies a file named ABCD under the usercode of PAPABEAR. If PAPABEAR's user ID was assigned as pb, CANDE would search for a file named ABCDpb.

(JONES:2)ABCD

This example specifies a file named ABCD under the usercode of JONES in VCS installation number 2.

*PROGFL

This example specifies a nonusercoded file.

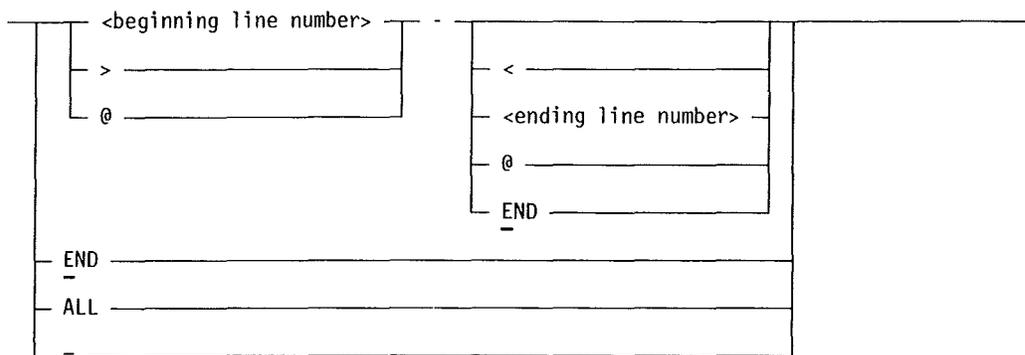
8298:ABCD

This example specifies a file named ABCD with a former CANDE numerical usercode of 8298 (EBCDIC value of bp). Former CANDE usercodes were restricted to 8181 (aa) through 9999 (rr).

Sequence Range

The sequence range indicates which lines in a file will be affected by a command. The default range is every line in the file—line zero through the end of the file (line 99999999).

Syntax



Explanation

- | | |
|-------------------------|--|
| <beginning line number> | The line number of the first line you want to include in the sequence range. By default, the beginning of the file (line zero) is assumed. |
| > | The greater-than sign (>) specifies the first line of a previously defined block as the beginning line number. See "BLOCK Command" for information on defining a block. |
| @ | The "at" sign (@) used as all or part of a sequence range specifies the last edited sequence number (LESN) as the sequence number. |
| - | The hyphen (-) separates the beginning and ending sequence numbers. When no ending sequence number or valid symbol is provided after the hyphen, the hyphen specifies the end of the file. |
| < | The less-than sign (<) specifies the last line of a previously defined block as being the ending line |

Explanation

QUIT	QUIT is a synonym for ABORT. Both commands share identical syntax and function.
DO	This command terminates the processing of an executing batch file. All currently executing tasks, including those initiated by the batch file, continue to run.
COMPILE	This command terminates only those tasks that are awaiting initiation due to a locked file. The COMPILE option aborts the auto-retry of a locked file. For further information on auto-retry, refer to "COMPILE Command."

Examples

```
DO *DOFILE
```

```
ABORT DO
```

The first entry of this example executes a batch file with the DO command. The second entry terminates the task processing of the batch file and all active tasks.

```
DO CMP1
```

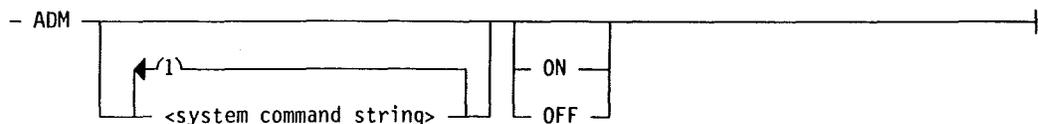
```
ABORT COMPILE
```

The first entry of this example executes a batch file with the DO command. The second entry does not terminate processing of the batch file, but instead terminates any COMPILE command attempting to open a locked file for processing. This situation can be the result of invoking a DO or COMPILE command.

ADM (Auto Display Mode) Command

Use the ADM (auto display mode) command to enable your workstation to receive responses from up to two specified system command strings. A 15-second delay will occur between each response.

Syntax



APPEND Command

Explanation

<system command string>	This is the name of a Master Control Program (MCP) command. Any V Series system command that is listed in the <i>V Series Systems Operations Guide, Volume 2: System Commands</i> can be used as a keyboard command string. You can include up to two command strings with the ADM command. Each command string must be enclosed in double quotation marks (").
ON	This enables automatic display mode.
OFF	This disables automatic display mode. OFF is the default value for automatic display mode at the start of a CANDE session.

Details

When using CANDE with VCS, responses to the system command strings vary depending on whether you used the privileged level CANDE transaction to invoke the ADM command. The transaction with a DCS trancode of 2 is the privileged transaction. The transaction with a DCS trancode of 1 is the nonprivileged transaction.

When using CANDE with GEMCOS, all system command strings are associated with the nonprivileged transaction.

Example

```
ADM "WT;WS;OLPCR" "DQ" ON
```

This example defines two command strings for use with the ADM command and enables the automatic display mode.

APPEND Command

Use this command to append all or a portion of a specified editor file to the end of the current editor file, beginning with the highest line number in the specified file plus the increment.

Syntax

```
- APPEND <file title> [ <sequence range> ] [ + <sequence incr> ] [ NOPATCHID ]
```


BLOCK Command

Explanation

- <sequence range> This is the beginning and ending sequence numbers backward in the editor file that you want to display.
- + <nn> This defines the number of pages from 1 through 99 to skip forward before starting the display.
- <nn> This defines the number of pages from 1 through 99 to skip backward before starting the display.

Details

CANDE stays in backward paging mode until you enter a LIST command, get a new file, or terminate the session.

If you transmit the BACK command with no parameters, the next sequential page back is displayed. If you transmit the BACK command with the cursor positioned after one or more lines of text, the display begins with the last line you transmitted (the last edited record).

Examples

BACK

This example displays one page back from the current page.

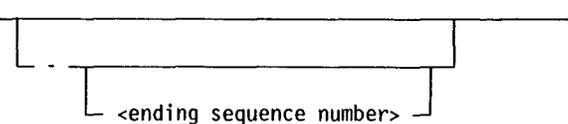
B 5

This example displays five pages back from the current page.

BLOCK Command

Use the BLOCK command in place of the MARK and BOUND commands or the greater-than (>) and less-than (<) symbols to establish a range of lines to be used as the current block. Entering the BLOCK command without parameters displays the current sequence range of the block.

Syntax

- BLOCK - <beginning sequence number> 

Explanation

- <beginning sequence number> This is the sequence number that will be the first line of the block. If no other element follows this sequence number, this sequence number constitutes the block.
- The hyphen separates the beginning and ending sequence numbers. If no ending sequence number follows the hyphen (-), the end of the file is assumed.
- <ending sequence number> This is the sequence number that will be the last line of the block.

Details

You can use the MARK and BOUND commands or the greater-than (>) and less-than (<) symbols to change the beginning and ending sequence numbers of the block after the current block has been defined.

The beginning line number must always be smaller than the ending line number.

When you make or get a file, the current block is defined as line number 00000000 through 99999999.

You can specify the current block as the sequence range in commands by using the equal sign (=).

Example

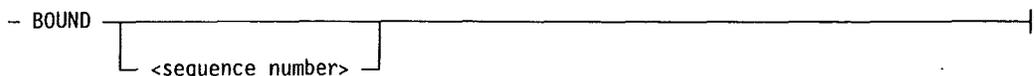
```
BLOCK 17000-17500
```

This command defines a 500-line block consisting of the lines numbered 17000 through 17500.

BOUND Command

Use the BOUND command to update the ending line number of the current block. Entering the BOUND command without parameters displays the sequence number that defines the end of the block.

Syntax



BYE Command

Explanation

<sequence number> This is the sequence number that marks the end of the block. It cannot be lower than the sequence number of the beginning line of the block. If you do not specify a sequence number, the default will be the last edited sequence number (LESN), if valid. For further information on the last edited sequence number, see "LESN Command" earlier in this section.

Details

When you make or get a file, the value for the BOUND command is initialized to 99999999.

Once the ending sequence number of the block is established, you can use the less-than symbol (<) to represent this sequence number in commands.

Examples

```
BOUND 10000
```

This example specifies line 10000 as the ending line number.

```
00000100?COMPILE TESTST WITH BPL LIBRARY  
00000110&?FILE PRINT BUJUNK  
<
```

This example substitutes the less-than sign (<) for the BOUND command. Because no value is specified with the less-than sign, the bound value defaults to the last edited sequence number, 00000110.

BYE Command

Use this command to terminate your CANDE session and release all editor files to the system.

Syntax

```
BYE _____  
END
```

Explanation

END END is a synonym for BYE. Both commands share identical syntax and function.

Details

This command always checks for active subordinate tasks before terminating the session. If active tasks are found, a continuation request is made to ensure the desired action.

The effect of this command to the current editor file is the same as if you had executed a release on the file. If the file retention type for your file is PERMANENT, the file is kept in your library; otherwise, the file will be purged. If there is a recall file associated with the editor file, the recall file will be processed according to the recall disposition options (see "SESSION Command"). For a description of a recall file, see "GET Command" and "RECALL Command."

CANUTL Command

Use this command to execute the CANUTL utility. The system responds with a menu from which you can invoke a specific CANUTL function.

Syntax

- CANUTL _____|

Details

The CANUTL utility is automatically initiated when you execute a COPY, REBUILD, or REORGANIZE command.

When you enter the CANUTL command, the following menu is displayed:

CATALOG Command

```

                                CANUTL MENU
Input file:                               on
Output file:                              on
Precede non usercoded file name(s) with * (Default output = input filename)

Options - Select only one numeric choice by placing non blank char in field(s)
1. Reorganize/compress CANDE editor file
   Retain current records per area (RPA)
2. Rebuild corrupted CANDE editor file as a CANDE editor file
3. Rebuild corrupted CANDE editor file as an 80 byte system file
   Option 2 or 3: Include deleted records
4. Create a CANDE editor file from an 80/90 byte system file
   Editor output filekind:
5. Create a CANDE editor file from an EDIT-1 editor file
6. Create an 80 byte system file from an EDIT-1 editor file
   Option 5 or 6: Use columns 1-72 as text
7. Convert editor to output file of: (PRINT,PUNCH,TAPE,DISK,PACK,PCR)
   Output page size or blocking factor
8. Compare input to output editor files and generate difference report
9. Print first 50 lines of input 80 byte system and/or editor file(s)
   Options 8 or 9: Printer page size (default = 56)

Options 4-7: Sequence range      .00000000.   THRU   .99999999.
Options 4-5: Resequence (base)  .00000000.   (+inc) .00000000.   XMIT here->
```

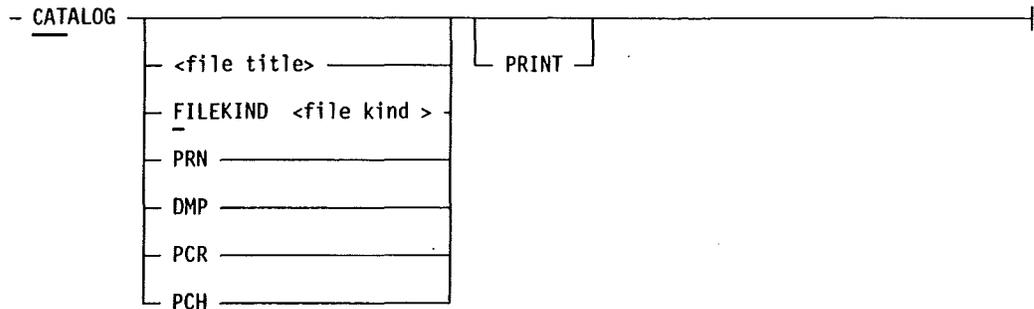
Enter the desired input and output file titles on the menu, and assign nonblank characters to options 1 through 9. Transmit from the last position on the menu.

CATALOG Command

Use this command to display a detailed description about

- A specific file
- All files of a specific file kind
- All printer backup files
- All dump files
- All pseudo card reader files
- All punch files

Syntax



Explanation

- <file title> This is the name and optional family name of the file about which you want information.
- FILEKIND <file kind > This option directs the system to display information for any CANDE editor file of the specified file kind.
- PRN This option directs the system to display information for all printer backup files.
- DMP This option directs the system to display information for all dump files.
- PCR This option directs the system to display information for all pseudo card reader files.
- PCH This option directs the system to display information for all punch files.
- PRINT This option directs output to the CANDE job log printer file. The PLOG command releases the file for printing.

Details

Entering the CATALOG command by itself displays information about all the files on the current default family.

If the disk file security (DFSC) option of the MCP SECURITY record is enabled, all files that have a matching session usercode and duplicate session user ID in the fifth and sixth positions are displayed. If DFSC is not enabled, only files matching the specified usercoded file title with the session user ID in the fifth and sixth positions are displayed.

The information displayed for files varies depending upon the file kind. The minimum information will always include the file name and the file kind.

For editor files, the displayed information includes

- The date and time that the file was last accessed
- The total number of records in the file
- The security classification of the file

For object code files, the displayed information includes

- The date and time that the file was compiled
- The compiler name
- The amount of memory required
- The number of segments

For printer backup, punch, and dump files, the displayed information includes the creation date and time.

If information extends beyond one screen, you can display the next screen by transmitting a space.

Examples

```
CAT FILEKIND COBOL
```

This example displays information about COBOL editor files owned by the session usercode on the default family.

```
CAT ABCD
```

This example displays information about all files titled ABCD<id> on the current default family. The <id> construct represents the 2-character session user ID.

```
CAT = ON BACKUP
```

This example displays information about all files under the session user ID on the BACKUP family.

```
CAT *A= ON PACK
```

This example displays information about all the nonusercoded files beginning with the letter A on resource pack.

```
CAT PRN
```

This example displays information about all the printer backup files on the primary and secondary MCP families.

CAT

This example displays information about all the files under the session user ID on the current default family.

CHANGE Command

Use this command to change the name of a file in your library.

Syntax

```

  [ CHANGE ] <file title> [ TO ] <file title>
  [ RENAME ]

```

Explanation

RENAME

RENAME is a synonym for CHANGE. Both commands share identical syntax and function.

<file title>

Specify the name of the file to be changed as the first file title, and specify the new name as the second file title. For the complete syntax of file title, see the beginning of this section.

Details

If you try to change a file name to a file name that already exists in your library, the CHANGE command will be rejected. The CHANGE command will not affect the file kind or the data contained in the file.

Examples

```
CHANGE FIL1 TO FIL2
```

This example changes the name of file FIL1<id> to FIL2<id>, where <id> is your 2-character user ID which CANDE automatically appends to a usercoded file.

```
CHANGE *MYFILE ON TESTPK TO *TSTFIL
```

This example changes the name of the nonusercoded file MYFILE to TSTFIL. Both files reside on the TESTPK family.

COMMENT Command

Use this command to create a comment that is stored in the current editor file. The comment text can be viewed as part of the response to a WHATS command.

Syntax

- COMMENT <comment text> _____|

Details

The comment text can be up to 80 characters in length. If the comment contains special characters or embedded spaces, it must be enclosed within double quotation marks.

COMPILE Command

Use this command to initiate and monitor the compilation of either the current editor file or another source file.

Syntax

[COMPILE] [<file title>] [<sequence range>] _____|
[START]

Explanation

START

START is a synonym for COMPILE. Both commands share identical syntax and function.

<file title>

This is the file you want to compile. If you do not include a file title, the current editor file is assumed. For the syntax of the file title, refer to the beginning of this section.

<sequence range>

This is a specific range of lines from the editor file that you want to compile. For the syntax of the sequence range, see the beginning of this section.

Details

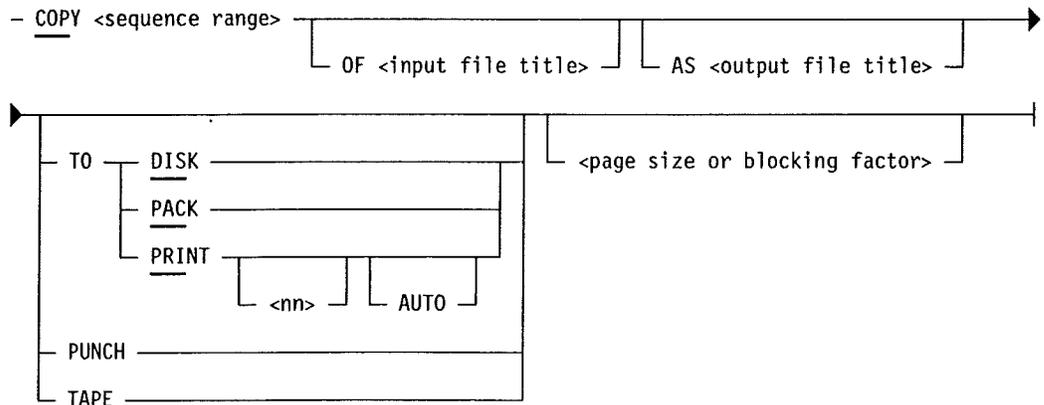
The COMPILE command is used to submit either the active editor file or any other editor or system file to the operating system for compiler or task initiation. If you compile the active editor file, the file is automatically released with no recall file update.

If the compiler or task initiation fails, and the active editor file is in COMPTS format, the editor file is automatically reopened. (See the START command for a discussion of the COMPTS format.) With all other file or control card formats, you must use the GET command to reopen the file when a failure occurs.

COPY Command

Use this command to copy an editor file as a system file to the hardware medium you specify. You can also use this command to convert an editor file to a specified file kind. The COPY command invokes the CANUTL utility, which performs the copy operation.

Syntax



Explanation

- <sequence range> This is a specific group of lines that you want to copy. For the syntax of the sequence range, see the beginning of this section.
- OF <input file title> This lets you specify a file other than the active editor file.
- AS <output file title> This option specifies a file title for the copy of the file. If you do not specify an output file title, the new file title consists of the first 4 characters of the input file name.

COPY Command

TO ...	This option followed by DISK, PACK, PRINT, PUNCH, or TAPE lets you select the output hardware medium. If you omit the TO clause, the current family (as determined by the FAMILY command) is used as the output hardware medium. If the output medium is a printer, and you have specified the COLINDICATOR option of the SESSION command, a heading will be added to each page.
<nn>	If the output hardware is a printer, you can specify a value from 00 through 99 to determine the number of lines to print before inserting a page break. A value of 00 inserts no page break. The default value is 56.
AUTO	The AUTO option causes the printer backup file to be automatically initiated for printing.
<page size or blocking factor>	The page size is applicable when the medium is a printer. You can specify a page size of 00 through 99. The blocking factor is applicable for DISK, PACK, or TAPE files. You can specify a blocking factor of 1 through 99. If you omit the value for the blocking factor, the blocking factor defaults to 1 for TAPE, 5 for DISK, and 9 for PACK.

Details

While the active editor file is being copied, your CANDE station will not respond to further CANDE commands. The destination of PUNCH and PRINTER files is determined by the MCP options that have been defined.

PACK files are created on disk pack with the specified family name. The output record length is 80 characters for all hardware media except PRINTER, which is 132 characters.

To copy an editor file to a pseudo card reader (PCR) deck, use the COMPILE command.

To copy an editor file without a CANDE terminal or to create an editor file from a system file, use the CANUTL utility described in Section 5.

Examples

```
COPY ALL TO PRINTER
```

This example copies all lines of the active editor file to a printer.

```
COP 100000-E AS MYFILE TO DISK
```

This example copies lines 100000 through the end from the active editor file to a system file titled MYFILE.

COP ALL OF MINE AS A123 TO PACK

This example copies all of the lines in the file titled MINE to a system file titled A123 located on the session default family.

COPY 10000-10500 OF 8188:HIZN TO PRINTER

This example copies lines 10000 through 10500 of an editor file labeled 8188:HIZN to a printer backup file.

CUT Command

Use this command to select a range of text to use with the PASTE command. The text is removed from its original location in the editor file and saved in a paste buffer. You can reinsert the text in another file or in another location in the same file by using the PASTE command.

Syntax

- CUT - <sequence range>

Explanation

<sequence range>	This is the range of lines in the editor file that you want to remove from the current location and place in the paste buffer. For the syntax of the sequence range, see the beginning of this section.
------------------	---

DEBUG Command

Use this command as an internal CANDE debugging aid to obtain information on the operation of CANDE code. This command produces a printer backup file that contains information on the progress of various CANDE functions.

Syntax

- DEBUG TRACE
OFF

Explanation

TRACE	This option initiates the machine-level debug of CANDE functions.
-------	---

DELETE Command

OFF This option terminates debug tracing.

DELETE Command

Use this command to delete a specified sequence range of source lines within the active editor file.

Syntax

- DELETE - <sequence range> _____|

Explanation

<sequence range> This is the range of lines in the editor file that you want to delete. Refer to the beginning of this section for further information on sequence ranges.

Details

CANDE record deletions are logical. Instead of destroying deleted records, CANDE removes them from the editor file and marks them for reuse. This allows the records to be reinserted by the REBUILD command as needed when performing a reconstruction of a corrupted file.

If more lines in the file are to be deleted than specified by the MAXDELETE option of the SESSION command, CANDE requests confirmation with the following prompt:

```
OK TO DELETE <n> LINES? (ENTER NO TO STOP)
```

If the CONFIRM option of the SESSION command is TRUE, you must answer YES or NO to this message, or the message is redisplayed when you try to perform any other action.

If the CONFIRM option of the SESSION command is FALSE, you can answer NO or supply any other answer to continue. If you answer NO, the command is terminated. If you enter YES or any other response, the deletion will continue. Any response you make other than YES is sent to the editor as the next command to execute after completing the DELETE command.

After the DELETE operation is completed, the editor displays the message *DELETED n* where *n* is the number of lines deleted.

DETANK Command

Use the DETANK command to perform the following functions:

- Display messages being held in the message tank (MSGTANK) to your station.
- Cancel the TANK command and direct CANDE to resume sending messages to your station. Any new messages follow the messages returned from the message tank. If the HARDCOPY option is enabled, messages are also written to the CANDE job log file.
- Reset the count of messages sent to your station. When this count again exceeds the workstation page size, the TANK command is automatically invoked (except when the page size is zero).

Syntax

```
- DETANK _____|
```

Details

If the workstation page size is greater than zero, and the number of messages in the message tank is greater than the workstation page size, then CANDE will send you a number of messages that equal the workstation page size, and then automatically invoke the TANK command. Refer to "TANK Command" for further information.

DO Command

The DO command initiates a series of commands from a file you have prepared in advance. This file is referred to as a CANDE batch file.

Syntax

```
- DO <file title> _____|
  |_____|
  | <sequence range> _____|
  | @ <label> _____|
  | < <variable name> > <variable value> _____|
```

Explanation

<file title>

The file title indicates the name and family location of the batch file. For information on naming batch files, see Section 4. For the syntax of file title, see the beginning of this section.

<sequence range>

This is the range of lines in the editor file that you want this command to affect. For the syntax of the sequence range, see the beginning of this section.

DUPLICATE Command

@<label> This element executes a particular part of the batch file that is identified by a label. Only the part of the batch file that is between the LABELBEGIN and LABELEND statements is executed as a result of the DO command. For details about including labels within a batch file, see Section 4.

< <variable name> > This parameter lets you create a variable in the batch file. You must enclose the variable name in angle brackets to differentiate the variable name from the variable value. The name of the variable can be up to 17 characters long.

<variable value> This parameter lets you assign a value to a variable that you have specified in the batch file. CANDE inserts this value at the location of the variable name when the batch file is executed. Thus, you can use the same batch file in different ways by supplying different values for the variables.

If you use a variable name in the DO command but do not supply a value for the variable, blanks are inserted in the batch file at the location of the variable when the batch file is executed.

If there is a blank or a special character in the variable value, you must enclose the value in quotation marks when you enter the DO command.

Details

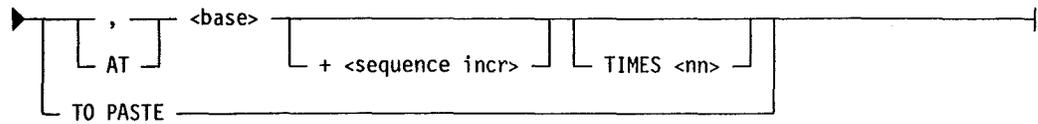
The DO command executes a batch file that can contain CANDE commands, V Series system commands, and online application transactions. Without a batch file, you would have to enter all the commands individually at your terminal. For instructions on creating a batch file, see Section 4.

DUPLICATE Command

Use this command to duplicate lines from one place to another in the editor file. This command does not remove the original lines.

Syntax

- DUPLICATE <sequence range> 

**Explanation**

<sequence range>	This is the range of lines in the editor file that you want to duplicate. For the syntax of the sequence range, see the beginning of this section.
AT <base>	This is the line number at which you wish the duplicated lines to be inserted. If this line number already exists, duplicated lines will be inserted at the base value plus the sequence increment.
+ <sequence incr> r	This is the resequencing increment between line numbers. The increment is an 8-digit value. If you omit the sequence increment, the increment assigned by the SEQINC option of the SESSION command is used. If the lines to be duplicated will not fit into the space available without overlapping the preexisting lines, the duplication is not performed. When the duplication is not performed the following message is displayed: DUPLICATE NOT DONE, MAXIMUM INCREMENT = nnnnnnnn The increment displayed is the maximum that can be used without causing an overlap. In such a case it is advisable to reduce the increment between lines, or resequence the file with a larger increment, and then perform the DUPLICATE command.
TO PASTE	This indicates that the duplicated lines are to become the new PASTE file instead of being inserted into the active editor file.
TIMES<nn>	This is the number of times the sequence range will be duplicated at the location you selected. The range is 1 through 99. The default is 1.

EMODE Command

Use this command to change from the VF utility mode to the CANDE editor mode. All commands not preceded by the mode override character will be processed by CANDE. In this mode, you can send a command to VF by prefacing the command with the mode override character, a period (.) by default. For example, *.PAGE* would be sent to VF, and *PAGE* would be sent to CANDE. You can change the

END Command

mode override character by using the MOC option of the SESSION command. To change from CANDE mode to VF mode, use the VMODE command.

See Section 3 for more information on the VF utility.

Syntax

- EMODE _____|

END Command

Use this command to terminate the CANDE session and release all editor files to the system.

Syntax

[END
BYE] _____|

Explanation

BYE BYE is a synonym for END. Both commands share identical syntax and function.

Details

This command always checks for active subordinate tasks before terminating the session. If active tasks are found, a continuation request is made to ensure the desired action.

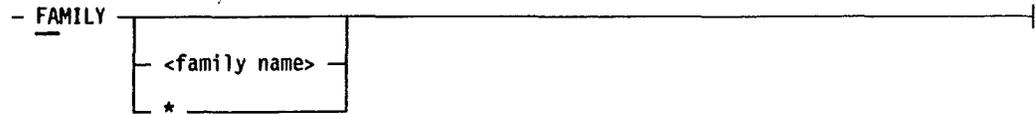
The effect of this command to the active editor file is the same as if you had executed a release on the file. If the file retention type for your file is PERMANENT, the file is kept in your library; otherwise, the file will be purged. If there is a recall file associated with the editor file, the recall file will be processed according to the recall disposition options (see "SESSION Command"). For a description of a recall file, see "GET Command" and "RECALL Command."

FAMILY Command

Use this command to specify the disk or disk pack family that you want to be the default family specification for all commands. Enter this command without parameters to display the current family.

Note: *You cannot change the family name with this command if your accesscode specifies a FAMILYONLY attribute.*

Syntax



Explanation

- <family name> This parameter lets you specify any family as the file storage family.
- * This option reverts to the family attribute defined for your accesscode.

Details

CANDE commands use the current family unless you specify an override family in command syntax. At the beginning of each CANDE session, the current family is set to your accesscode default, unless you have a FAMILY statement in your user variable file. In this case, the current family is set to the family defined in the user variable file.

You can change the current family at any time during the session by using the FAMILY command. If you change to a different family while you have an active editor file, you must enter the appropriate family name with subsequent commands.

To specify a file on a family other than the current family, specify the family name as part of the file title in the command syntax. For the syntax of file title, see the beginning of this section.

Because CANDE retains the location of the original editor file, any commands or lines entered apply to that file, on its original family.

Examples

```
FAMILY = PACK
```

This example makes the system resource pack the current family. PACK will be the default family used in all commands in which you do not specify a family name.

FILES Command

FAMILY = TESTPK

This example makes TESTPK the current family. TESTPK will be the default family used in all commands in which you do not specify a family name.

FAMILY *

This example makes the default family revert to the default family associated with your accesscode.

FILES Command

Use this command to generate an ascending sorted list of the names of the files in your library.

Syntax

```
- FILES [ <file title> ] [ NOSORT ]
```

Explanation

<file title> This is the name and optional family name of the file whose title you want to list. Usercoded file names of less than 4 characters assume trailing masking characters. For example, if *AB* is the file name, the system displays all file names beginning with *AB* regardless of the last two characters. Nonusercoded file names are interpreted literally, so you must include the desired masking characters. If you omit the file title, the names of all usercoded files on your default family are listed.

NOSORT This option displays the file names as they appear on the family media, rather than displaying them sorted in ascending order.

Examples

FIL

This example lists all the usercoded files in your library.

FILES ==1=

This example lists all the files that have the numeral 1 in the third position of their name. The equal sign (=) is used as a masking character, which

causes CANDE to ignore the character in that position. Masking characters can be used in any position of the file group specifier.

FILES ABCD

This example searches for the usercoded file named, ABCD, and displays the file name if the file is in your library.

FILES Q

This example lists all usercoded files beginning with a Q.

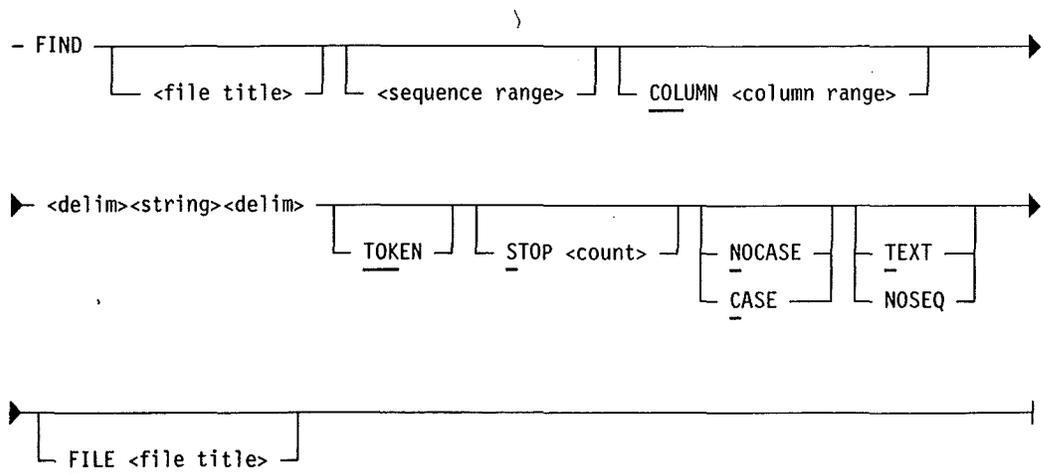
FILES *A1====

This example lists all usercoded and nonusercoded file names that begin with A1 on the default family.

FIND Command

Use this command to search for a string in a range of lines in your editor file and to list the line numbers of the lines that contain the string. Optionally, you can have the system list the lines of text that contain the string you are looking for.

Syntax



Explanation

<file title>

The file title option allows you to select a file other than the active editor file. If you do not enter a file title, the active editor file remains selected. For the syntax of the file title, see the beginning of this section.

FIND Command

<u><sequence range></u>	This is the range of editor file lines you want to be searched. If you do not specify a value, CANDE searches all the lines of the file. Refer to the beginning of this section for further information on sequence ranges.
<u>COLUMN</u> <column range>	This narrows the scope of the search to the indicated column range. If no column range is selected, the defaults of the first and last text locations for the file kind will be used.
<u><delim></u>	This element is used to delimit the ends of the string to be searched for and must not appear in the string itself. You can use any special character as a delimiter, except an ETX.
<u><string></u>	This is the group of characters you want to find. The maximum string length is 80 characters.
<u>TOKEN</u>	This indicates that the string must be delimited by spaces in the file to be identified as the string.
<u>STOP</u> <count>	This allows you to select how many times the string may be found in your editor file before the FIND command terminates. The absence of this parameter will cause all occurrences of the string to be displayed. The count must be a 2-digit integer in the range of 01 through 99.
<u>CASE/NOCASE</u>	You can use these parameters to determine whether your FIND command execution will be case sensitive. The default is controlled by the CASE option of the SESSION command.
<u>TEXT</u>	This lists the entire record, including the line number, in which the string is found. The record and the line number are also listed if the DISPTEXT option of the SESSION command is set to TRUE. Otherwise, only the line numbers where matches are found are listed. Refer to "SESSION Command" for more information on the DISPTEXT option.
<u>NOSEQ</u>	This parameter suppresses the sequence number list that is usually displayed when no text output is to be generated. This option is useful if the matches are to be sent to an output file with the FILE option of this command. NOSEQ is valid only if the DISPTEXT option of the SESSION command is FALSE.
<u>FILE</u> <file title>	This allows you to select a file to which the occurrences of the string will be copied. If the specified file already exists and is in a correct output file format, the new

records are appended at the end. If the file format is not correct, the global AUTOREMOVE session attribute is checked. If AUTOREMOVE is set to TRUE, a new output file is opened to overwrite the existing one. If AUTOREMOVE is set to FALSE, a duplicate file error message is displayed.

Details

The total number of matches displayed at the end of the command reflects the total number of sequence lines that contained a match. Only one match per line is reflected in this total.

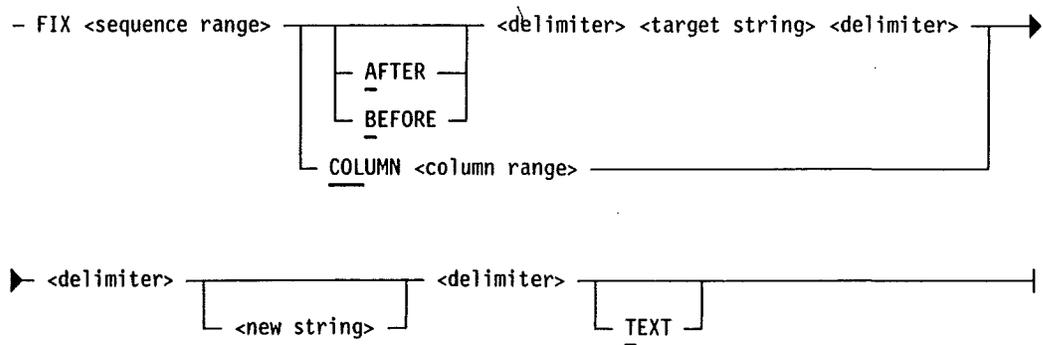
You can interrupt the FIND command by entering the following command:

?BRK

FIX Command

Use this command to insert a character string or to justify text in one or more lines of the editor file.

Syntax



Explanation

`<sequence range>`

This is the range of editor file lines you want to be examined. Refer to the beginning of this section for further information on sequence ranges.

`AFTER/BEFORE`

These options are used to insert the new string BEFORE or AFTER the target string. For example, the following syntax inserts the string `OR VALUE = 2` after all occurrences of the string `VALUE = 1`.

`FIX 1000-3000 AFTER /VALUE = 1/ /OR VALUE = 2/`

COLUMN <column
range>

This parameter indicates that a new string is to be inserted into a certain column range in all lines indicated by the sequence range. If the new string is shorter than the column range, the extra characters in the column range are deleted. The COLUMN option facilitates left or right justification of blocks of text.

You cannot use the COLUMN option to indicate that a target string is to be searched for in a particular column range. For this function, use the REPLACE command.

<delimiter>

This element is used to delimit the ends of the string to be searched for and must not appear in the string itself. You can use any special character except an ETX as a delimiter.

<target string>

This is the group of characters to be searched for in the file. The maximum string length is 80 characters.

If you omit the target string, the COLUMN option must be included. The length of the new string in this case must also be less than or equal to the length indicated by the column range.

The following example moves all text lines in sequence range 1000 through 2000 four columns to the right. It does this by inserting blanks in columns 12 through 16.

```
FIX 1000-2000 COL 12-16 / /
```

<new string>

This is the group of characters that will replace the target string. The maximum string length is 80 characters. If the new string is null (only the delimiters are specified), all occurrences of the target string are deleted.

The new string you enter can contain any characters except the character chosen as the delimiter. The new string can be longer or shorter than the target string, and the characters to the right of the target string in the affected lines will be shifted accordingly.

TEXT

This parameter lists the text as well as the sequence number of a line in which a change has been made. If you do not specify this option, the value of the DISPTEXT option of the SESSION command is used. (The default value for the DISPTEXT option is TRUE.) The TEXT option overrides the DISPTEXT option. If more than one change is made on a line, the line is listed only once.

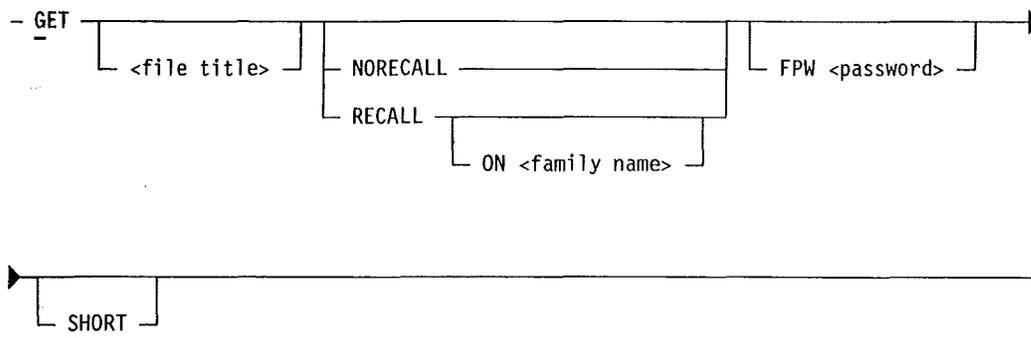
GET Command

access this file no longer have to supply a password when they open the file with the GET command.

GET Command

Use this command to open a previously created file as the active editor file. Entering this command without parameters loads the file previously accessed with the GET command. This command optionally creates a backup file called a recall file.

Syntax



Explanation

<file title>

This specifies the file you want to open as the active editor file. All changes you make to the editor file after you open it are permanent. If you do not include the file title, the last file accessed by this command during the session is used, if available. For the syntax of the file title, refer to the beginning of this section.

RECALL/NORECALL

The RECALL and NORECALL parameters let you specify whether the editor file will have an associated recall file. These parameters override the value set for the RECALL option of the SESSION command. If you omit this parameter, the default is determined by the RECALL option of the SESSION command.

The recall file is useful when you desire to have backup protection for the active editor file. The recall file gets updated when you enter the SAVE command or if you have assigned a value to the AUTOSAVE option of the SESSION command. You can use the recall file at any time to restore the active editor file to the last saved version. Use the RECALL command to replace the active editor file with its associated recall file.

ON <family name>	This option lets you specify an alternate family on which to store the recall file. This option overrides the RECFAM option of the SESSION command.
FPW <password>	If a CANDE file password has been assigned to the file you are attempting to access, you must supply the password as part of the GET command. Refer to "FPW Command" earlier in this section for more information on file passwords.
<u>SHORT</u>	You can use this option to obtain a one-line summary of the information screen that is usually displayed when a file is accessed. The one-line summary includes the file title, the file kind, and the generation status of the recall file.

Details

Entering a GET command while an editor file is present releases the active editor file. For details on releasing files, see "RELEASE Command."

Examples

GET UVF:

This example loads your user variable file as the active editor file.

GET *AFILE ON TESTPK SHORT

This example loads the nonusercoded file AFILE located on the TESTPK family and displays a one-line summary of file information.

HARDCOPY Command

Use the HARDCOPY command to start or stop the writing of messages to the CANDE job log printer file.

Syntax

HARDCOPY [ALL] [ON <nn>] [OFF]

HELP Command

Explanation

- ALL** This option causes both user input and responses sent to the station to be logged in the CANDE job log file. By default, only messages that come from the MSGTANK are logged.
- ON <nn>** This option starts the recording of messages to the CANDE job log file. The <nn> value is a 2-digit number that represents printer page size. CANDE will put the specified number of lines on each page of the printer file, following each with a page feed. Once established, this value is retained throughout the session or until you change it. The default value is 56. If you use a value of 0, no page breaks occur in the job log file.
- OFF** This option stops the recording of messages to the CANDE job log. You can start and stop recording without printing or otherwise disturbing the contents of the CANDE job log file. OFF is the default setting for the HARDCOPY command.

Details

If the HARDCOPY option is enabled, a new CANDE job log file is opened immediately after the old log is closed and released by the PLOG command.

The default setting of the HARDCOPY command is OFF.

If you enter the HARDCOPY command by itself, the system displays the current setting of the HARDCOPY command.

HELP Command

Use the HELP command to display a list of CANDE editor commands or to display information about a specific editor command.

Syntax

```
| HELP |-----|
| TEACH | | <command> |
```

Explanation

TEACH TEACH is a synonym for HELP. Both commands share identical syntax and function.

<command> This is the command about which you want information.

Details

If you enter *HELP* without a command, a list of valid CANDE commands is displayed on the screen with instructions on how to proceed.

If you enter *HELP* with a command, the syntax for the command and an explanation of the syntax are displayed on your screen.

JMGR (Job Manager) Command

Use this command to enable your workstation to adopt a task that is not owned by another job manager task, or to abandon a task that is currently running at your workstation.

Syntax

```
- JMGR [ ADOPT | ABANDON ] <task number> _____
```

Explanation

ADOPT

This option adopts a running task that was initiated outside your CANDE session. The adopted task continues to run as if it had been initiated at your workstation. All subsequent messages pertaining to the adopted task are displayed at your workstation.

ABANDON

This option abandons a task that is running as part of your CANDE session. The abandoned task continues to run, but messages pertaining to the abandoned task are not displayed at your workstation.

<task number>

This is the mix number of the task that you want to adopt or abandon.

LAST Command

Use this command to display the last page of the active editor file.

Syntax

- LAST _____|

LESN (Last Edited Sequence Number) Command

Use this command to display a page that begins with the last edited sequence number processed as input to the active editor file.

Syntax

- LESN _____|
 + <nn>
 - -
 - LINE -
 - PAGE -

Explanation

- + This indicates that the display direction is forward through the editor file. A space must precede the plus (+) sign.
- This indicates that the display direction is backward through the editor file. A space must precede the minus (-) sign.
- <nn> This defines the number of lines or pages to begin the display before or after the current LESN. The default is 0 and the maximum value is 99. A space must precede the plus (+) or minus (-) sign.
- LINE This specifies lines as the unit display for text. This is the default value.
- PAGE This specifies pages as the unit display for text.

Details

The DISPLAY option of the WORKSTATION command has no effect on this command.

You can substitute the “at” sign (@) for the last edited sequence number in commands. The following example uses this substitution:

```
MOVE = @ +10
```

The syntax in the preceding example causes the current block definition, indicated by the equal sign (=), to be moved to the sequence number beginning with the last edited sequence number and incremented by 10. Thus, if the last edited sequence number is 100, the block is moved to line number 100 and is incremented in units of 10.

The last edited sequence number appears in the information bar at the bottom of your screen if you specified the INFOBAR option with the WORKSTATION command.

Example

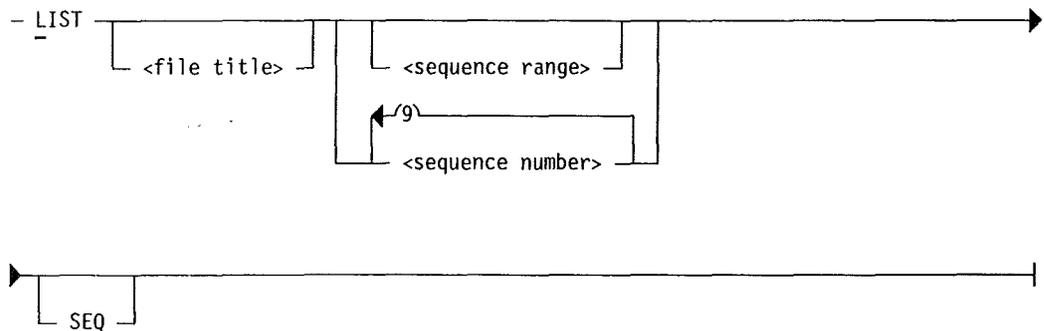
```
LESN+22L
```

This example displays a page that begins 22 lines forward from the last edited sequence number.

LIST Command

Use this command to list all or a specified portion of an editor file. You can list the editor file with which you are currently working, another editor file in your library, or an editor file in another user’s library if you have access permission.

Syntax



Explanation

<file title>

This option lets you select a file other than the active editor file to list. If you do not enter a file title, the active editor file is assumed. For the syntax of the file title, see the beginning of this section.

LIST Command

You can use the LIST command to view only CANDE editor files. To view files other than editor files, use the VIEW command.

<sequence range> This is the range of editor file lines you want to list. For the syntax of the sequence range, see the beginning of this section. The default is ALL.

<sequence number> This option lets you specify up to 10 individual lines to list.

SEQ All files except data files contain sequence numbers as part of the record. Thus, the sequence numbers are displayed when you list the file. Data files have pseudo sequence numbers that are not considered part of the record. Thus, when you list a data file, only the text is displayed.

This option specifies that pseudo sequence numbers will be displayed for data files listed with the LIST command. This option overrides the global setting defined by the DATA SEQ NUM option of the SESSION command.

If you specify the SEQ option, the 8-digit line number of each record appears in columns 1 through 8 of your terminal, followed by text. For data or sequential files with text lengths greater than 72 characters, each succeeding line begins with an ampersand (&) in column 1, followed by additional text.

If you do not specify the SEQ option, the first character of the first data file record appears in position 1 on the first line of your terminal. Additional lines of the terminal are then used, as needed, to hold the complete record. Each continuation line begins with an ampersand (&) in column 1 of the succeeding terminal line.

Details

You can use the LIST command to view only CANDE editor files. To view files other than editor files, use the VIEW command.

When you are listing a file other than the active editor file, you can use the NEXT command or a space to page through the file you are listing. All other commands are interpreted as commands for the active editor file.

If you have enabled column indicators by using the COLINDICATOR option of the SESSION command, the column indicators appear as the first line of the screen

display listed with the LIST command. Following is an example of how column indicators appear:

```

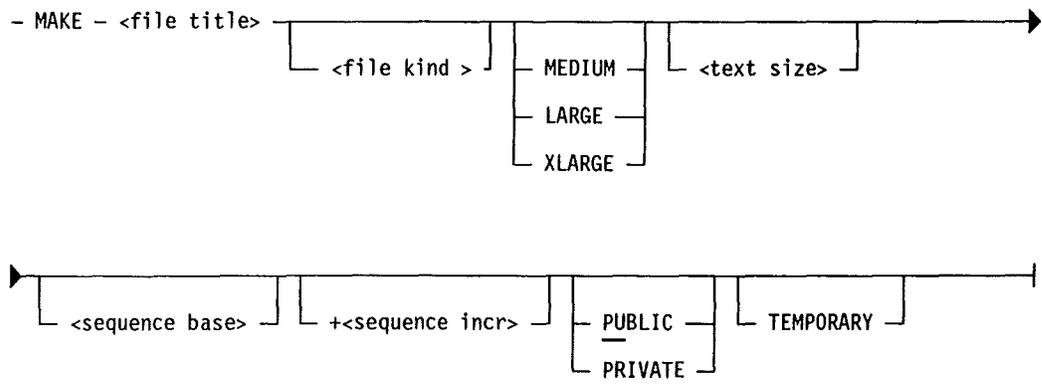
-----1-----2-----3-----4-----5-----6-----7-
00010010The first line is an example of column indicators as
00010020they appear when you list a file with the COLINDICATOR
00010030option of the SESSION command enabled.
    
```

Note: The LIST command differs from the PAGE command in that if you specify a single range value with these commands, LIST displays the matching record, whereas PAGE displays a screen of records beginning with the specified record.

MAKE Command

Use this command to create a new editor file of the type you specify.

Syntax



Explanation

- `<file title>` This is the file name and optional family name of your editor file. The file name must be four alphanumeric characters. The first character must be alphabetic. For the syntax of the file title, see the beginning of this section.
- `<file kind >` The available file kinds with their corresponding compilers, their maximum usable text length per line, and the position of text and line numbers within the 80-character line are shown in Table 2-1.
- MEDIUM These parameters let you specify an area size for your file. The area size for an editor file usually starts at 10 records per area and is increased by CANDE only when
- LARGE
- XLARGE

the contents dictate such a need. To avoid possible delays in the future when CANDE expands the area size, you can set the initial area size by using one of these options, which create the following file area sizes:

MEDIUM: 50 records per area
LARGE: 200 records per area
XLARGE: 1000 records per area

- <text size> This option applies only to data and sequential files and refers to the text length of the records in these files. The default text length is 80 characters for data files and 72 characters for sequential files, but can extend to 476 characters.
- <sequence base> This is the beginning sequence number for your editor file. Using this parameter invokes the SEQUENCE command, which automatically inserts sequential line numbers into the file.
- If you do not specify a base value, the value of the SEQBASE option of the SESSION command is used. If the SEQBASE option was not declared, the default value of 100 is used.
- + <sequence incr> This is the sequence increment between line numbers. The increment is an 8-digit value. A space must precede the plus sign (+). Using this parameter invokes the SEQUENCE command, which automatically inserts sequential line numbers into the file.
- If you omit the sequence increment, the increment assigned by the SEQINC option of the SESSION command is used. If no increment was assigned with the SESSION command, the CANDE default value of 10 is used.
- PUBLIC This option allows anyone to access the file when the system is running under security. PUBLIC is the default value.
- PRIVATE This option restricts access to the file to the person who created the file and to users who have been assigned MCP privileged status when the system is running with file security.
- TEMPORARY This option causes the editor file to be purged when you enter a subsequent RELEASE, GET, or MAKE command or when you terminate CANDE by using the BYE command. If you do not select this option, the file will be permanent.

Table 2-1. CANDE File Kinds

File Kind	Text Length	Text Position	Sequence Number Length	Sequence Number Position
ADSINP	72	1-72	8	73-80
BINDER	72	1-72	8	73-80
BPL	72	1-72	8	73-80
COBOL	74	7-80	6	1-6
DASDL	72	1-72	8	73-80
DATA	1-476	1-170	8	
FORTRAN	72	1-72	8	73-80
IPS	74	5-80	4	1-4
PASCAL	72	1-72	8	73-80
RPG	75	6-80	5	1-5
SEQ	1-476	9-170	8	1-8
WFL	72	1-72	8	73-80

Note: Any file kind can be abbreviated to the minimum unique characters of the file kind.

Format of Data and Sequential Files

An 8-digit line number is associated with each record in a data file, but is not part of the record. When a data file is listed, the line number is displayed only if the SEQ option is specified. The line number is never part of the system file created by a COPY command that copies a data editor file.

In a sequential file, the eight-digit line number physically occupies the first eight positions of each record.

Data files usually contain 80 text characters, but can hold up to 476 characters. Sequential files usually contain 72 text characters but can hold up to 476 characters. You can specify text sizes when you use the MAKE or TYPE command.

Format of RPG Files

RPG program records always contain 75 characters of text and a 5-digit line number, for a total of 80 characters. Compile-time vector data records can be included in an RPG program.

Vector data records can contain up to 80 characters of text, but in an editor file, they also require a 5-digit line number and an asterisk immediately after the line

MARK Command

number (the asterisk denotes the record as a vector data record) for a possible total of 86 characters.

Editing Records Longer than 80 Characters

Data, sequential, and RPG vector data records can require more than 80 characters each. Since the width of most terminals does not exceed 80 characters, the following method is used to display and enter records that exceed the terminal width capability.

In records longer than 80 characters, the line number and text appear on the first line. Subsequent lines for the same record begin with an ampersand (&) in column 1, followed by the additional text. The following lines provide an example of this format:

```
00010100This is the first line of the first record and
&here is the second line
```

```
00010200This is the first line of the second record and
&here is the second line
```

You can omit the leading zeros in the line numbers that follow the first line number. For example,

```
00010100Record one, line one
&record one, line two
```

```
10200Record two, line one
&record two, line two
```

```
10300Record three, line one
&record three, line two
```

MARK Command

Use this command to update the beginning sequence number of the currently defined block.

Syntax

```
- MARK _____
   |_____|
   |<sequence number>|
```

Explanation

<sequence number> This is the sequence number that marks the beginning of the block. It cannot be greater than the sequence number of the ending line of the block. If you do not

specify a sequence number, the default will be the last edited sequence number (LESN), if valid. For further information on the last edited sequence number, see "LESN Command" earlier in this section.

Details

The MARK sequence number is initialized to zero when you make or get a file.

Once the beginning sequence number of the block is established, you can use the greater-than symbol (>) to represent this sequence number in commands.

You define the beginning and ending sequence numbers of a block by using the BLOCK command.

Examples

MARK 20000

This example updates the beginning sequence number of the current block to 20000.

```
00000100?COMPILE TESTST WITH BPL LIBRARY
00000110&?FILE PRINT BUJUNK
>
```

This example substitutes the greater-than sign (>) for the MARK command. Because no value is specified with the greater-than sign, the mark value defaults to the last edited sequence number, 00000110.

MERGE Command

Use this command to merge another editor file with the active editor file.

Syntax

```
- MERGE - <file title> _____
                        [ <sequence range> ] [ NOPATCHID ]
```

Explanation

<file title>

This is the file name and, if needed, the location of the file you wish to merge into your active editor file. For the syntax of the file title, see the beginning of this section.

MOVE Command

<sequence range> This allows you to select a range of editor file lines to be merged into your active editor file. Refer to the beginning of this section for further information on sequence ranges.

NOPATCHID The NOPATCHID option prevents merged lines from being flagged regardless of the setting of the PATCHID parameter. Only merged records that have blank patch ID locations or a previous patch ID will be updated with any patch ID from this command. Refer to "PATCHID Command" for further information.

Details

When lines in the merge file and the active editor file have identical line numbers, the active editor file lines are retained, and the merge file lines are disregarded. An error message and the duplicate line numbers are then displayed. A line counter prevents this data from being scrolled off the screen. If the screen fills, the display stops until you enter further input. Any input then continues the MERGE operation.

You can use the RMERGE command to replace editor file lines with merge file lines of the same line numbers.

Note: If the active editor file contains no records and you issue a MERGE command, the internal password for the active editor file is reset to blanks.

MOVE Command

Use this command to move lines from one place in your active editor file to another place in the editor file.

Syntax

- MOVE <sequence range> TO <sequence base> + <sequence incr>

Explanation

<sequence range> This is the range of editor file lines you want to move. For the syntax of the sequence range, see the beginning of this section.

TO This parameter is optional.

<sequence base> This is the line number at which you want to insert the moved lines. If you specify an existing line number, the moved lines are inserted at the base plus one.

+ <sequence incr> This option lets you specify a resequencing increment for moved lines. A space must precede the plus sign (+). If you omit the increment, CANDE uses the existing increment between the lines being moved. If the lines you want to move do not fit into the available space without overlapping, CANDE does not perform the move and instead displays the following message:

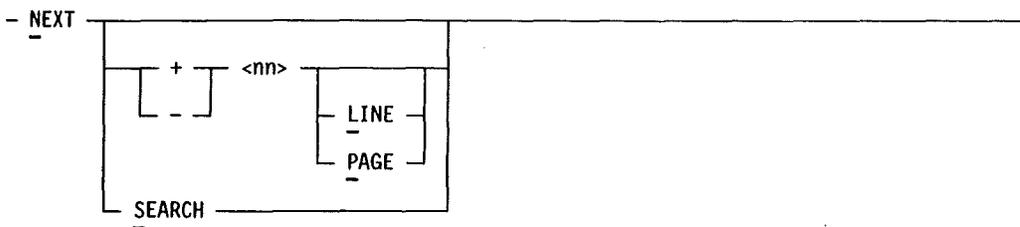
MOVE NOT DONE, MAXIMUM INCREMENT = nnnnnnn

The increment displayed in the preceding message is the maximum increment that can be used without causing overlap. If you get this message, you should either reduce the increment between lines or resequence the file with a larger increment, and then perform the MOVE command again.

NEXT Command

Use this command to move forward or backward through an editor file. Once you have defined a page direction by entering *N+* or *N-*, you can display the next page by entering an ETX or a space, or by pressing the SPCFY key.

Syntax



Explanation

- +** This defines a forward page direction. A space must precede the plus sign (+).
- This defines a backward page direction. A space must precede the minus sign (-).
- <nn>** This is a value from 0 through 99 that indicates the number of pages or lines to display forward or backward from the previous page. (The default unit is

PAGE Command

lines. To display pages, include the PAGE option of this command.) If you omit the value, the default value of 0 (zero) is used.

LINE

This specifies that lines is the unit by which text is displayed. Because lines is the default unit, this parameter is optional.

PAGE

This specifies that pages is the unit by which text is displayed. Lines is the default unit.

SEARCH

This option causes CANDE to search for the next match and is valid only if you previously invoked the SEARCH command.

Details

Once you have established page mode by using the PAGE command, subsequent displays automatically include the NEXT command followed by a plus (+) or a minus (-). The cursor is positioned after this command so that you can easily transmit it to display the next page.

You can reverse the page direction by changing the + or - sign to the desired direction before transmitting the NEXT command.

Examples

NEXT

This example displays the next screen of text as defined by the PAGE command.

N +20 P

This example skips forward 20 pages of text.

N -15

This example skips backward 15 lines of text.

PAGE Command

Use this command to page backward and forward through the active editor file, displaying full screens of the file or portions defined by a sequence range.

Syntax

```

- PAGE [ <file title> ] [ <sequence range> ] [ SEQUENCE ]

```

Explanation

<file title>

This option lets you page through a file other than the active editor file. If you do not enter a file title, the active editor file is assumed. For the syntax of the file title, see the beginning of this section.

The PAGE command is valid only for CANDE editor files. To view files other than editor files, use the VIEW command.

<sequence range>

This is the portion of the active editor file that you want to display. If you do not specify a sequence range, the display begins with the first record of the editor file and displays each page in sequence to the end. For the syntax of the sequence range, see the beginning of this section.

SEQUENCE

This applies only to data files. Data files have pseudo sequence numbers that are not part of the data record. This option causes the pseudo sequence numbers to appear as the first 8 characters of the display. If you do not specify the SEQUENCE option, the pseudo sequence numbers are controlled by the data sequence attribute setting at your workstation.

Examples

P

This example returns the first page and each page in succession through the end of the file.

P 10300

This example returns a page beginning with sequence number 10300 offset by any value specified for the OFFSET option of the WORKSTATION command.

P 1000-2000

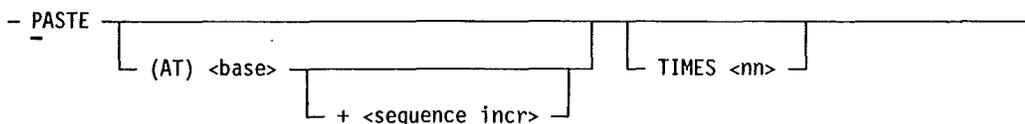
This example returns sequence numbers 1000 through 2000.

Note: The PAGE command differs from the LIST command in that if you specify a single range value with these commands, PAGE displays a screen of records beginning with the specified record, whereas LIST displays only the matching record.

PASTE Command

Use this command to insert text saved by the CUT command. CANDE checks to insure that all of the text will fit in the new location before attempting the insertion. If you enter the PASTE command by itself, the editor displays the first and last lines of the text saved with the CUT command.

Syntax



Explanation

(AT)<base>

This is the sequence number at which you want to begin the insertion of text. The keyword, AT, is optional.

+ <sequence incr>

This is the resequencing increment between line numbers. The increment is an 8-digit value. A space must precede the plus sign (+). If you omit the sequence increment, the increment assigned by the SEQINC option of the SESSION command is used. If no increment was assigned with the SESSION command, the CANDE default value of 10 is used.

If the lines you want to paste do not fit into the available space without overlapping, CANDE does not perform the paste and instead displays the following message:

```
PASTE NOT DONE, MAXIMUM INCREMENT = nnnnnnn
```

The increment displayed in the preceding message is the maximum increment that can be used without causing overlap. If you get this message, you should either reduce the increment between lines or resequence the file with a larger increment, and then perform the PASTE command again.

PCR (Pseudo Card Reader) Command

<column position> This is the position where the patch ID field begins. You need not specify a column position for COBOLV, COBOL74, or RPG files because these languages already contain source text fields for the patch ID field. All other file kinds require you to specify the starting column position.

Details

The patch ID field for COBOLV and COBOL74 is located in columns 73–80. The patch ID field for RPG is located in columns 75–80. All other file kinds require you to define the starting column for the patch ID.

CANDE automatically inserts the appropriate comment character as the first character of the patch ID (and the last character for Pascal).

Example

```
PATCHID LS<DATE>
```

This example indicates that added or modified lines should be flagged with the initials LS and the current date; for example, LS112090.

PCR (Pseudo Card Reader) Command

Use this command to build a pseudo card reader file from literal text that you supply. After you create the file, CANDE submits it to the operating system for execution.

Syntax

```
- PCR - <delimiter> - <text> _____|
```

Explanation

<delimiter> This element specifies the character that you will use to separate records in the text input. It can be any single character other than a blank, colon (:), slash (/), or hyphen (-). Use of this delimiter allows you to build pseudo card reader files that are longer than 24 records.

<text> This element represents the contents of the PCR record. The first nonblank character following the first delimiter is assumed to be record 1, column 1. All other blanks are preserved intact. Two consecutive delimiter characters result in a blank record.

Details

At least one blank character must follow the PCR command. This character can be placed anywhere in the input.

Examples

```
PCR #?COMPILE MYPROG WITH COBOL
LIBRARY#?FILE PRINT MYPROG/LIST AUTO#?FILE
SOURCE MYPACK/MYFILE DPK#
?DATA CARD# $SET LISTDOLLAR MERGE XREF LIST#?END
```

In this example, the pound sign (#) is the delimiter character.

```
PCR #?COPY A AS B FROM MYPACK(PACK) ,#
      B AS C FROM DISK
#           TO MYTAPE (TAPE) #
```

In this example, the pound sign (#) is the delimiter character. The leading blanks are preserved in all records other than the first record.

PLOG (Print Log) Command

The PLOG (print log) command closes and releases the CANDE job log file so that it can be printed. You can then print the log file with the PRINT system command.

Syntax

- PLOG _____|

Details

The PLOG command is automatically invoked at the end of your session if the HARDCOPY command is enabled.

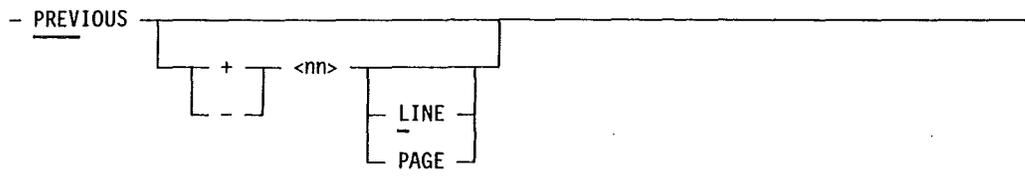
In addition, if the HARDCOPY option is enabled, a new CANDE job log file is opened immediately after the old log is closed and released.

PREVIOUS Command

Use this command to redisplay the previous page of the active editor file.

PREVIOUS Command

Syntax



Explanation

+	This defines a forward page direction. A space must precede the plus sign (+).
-	This defines a backward page direction. A space must precede the minus sign (-).
<nn>	This is a value from 0 through 99 that indicates the number of pages or lines to display forward or backward from the previous page. (The default unit is lines. To display pages, include the PAGE option with the PREVIOUS command.) If you omit the value, the default value of 0 (zero) is used.
<u>LINE</u>	This defines the unit for display as lines. LINE is the default value.
<u>PAGE</u>	This defines the unit for display as pages. The default value is LINE.

Details

The editor remains in the current direction after the completion of this command. You can change the direction by changing the + or - sign.

The DISPLAY SCROLL DOWN option of the WORKSTATION command does not affect this command.

Examples

PREV

This example redisplay the previous display page of the editor file.

PREV +5

This example redisplay the editor file beginning five lines forward from the previous display page.

PREV -2P

This example redisplay the editor file beginning two pages before the previous display page.

PAGE 1000
PAGE 10000
PREV

This example shows three command entries. The result of entering the PREVIOUS command redisplay the page beginning with sequence number 1000.

PSCREEN (Print Screen) Command

Use this command to copy your screen display (from home to the cursor position) into the CANDE job log file. After the display is copied, you can print the job log by using the CANDE *PLOG* command and the PRINT system command.

Syntax

- PSCREEN - <text to be printed> _____|

Details

The PSCREEN command must be the first command on the screen.

Also, if you want a page break between screen displays written to the CANDE job log file, you can use the following command:

PSCREEN PAGEBREAK

Examples

<initiator char> CANDE PSCREEN <text to be printed>

In this example, the station uses transaction-based routing.

PSCREEN <text to be printed>

In this example, the station either is permanently linked to CANDE or is in continuator mode (VCS only).

PURGE Command

Use this command to discard messages that have not yet been sent to the station. The PURGE command discards the following types of messages:

QUIT Command

- Any MSGTANK messages that have not yet been sent to the station.
- Any CANDE job log printer messages that have been collected for printing, but not yet released by the PLOG command.

Syntax



Explanation

PRINT The PRINT option closes the CANDE job log file. Messages purged from the MSGTANK are not included in the job log file. The job log printer backup file remains on disk or disk pack; you can print its contents by using the PRINT system command.

Details

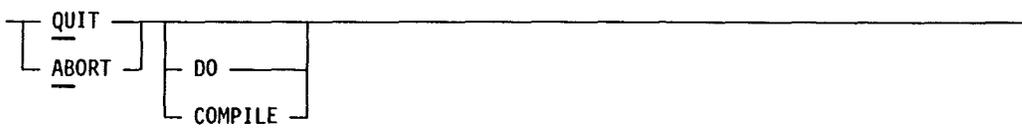
The purged messages will not appear in the CANDE job log file even if the HARDCOPY option has been enabled by the HARDCOPY command.

Each use of the PURGE command is recorded in the CANDE job log file.

QUIT Command

Use this command to terminate all tasks that were initiated during the current session. This includes any batch file that is being processed and any task initiated from the batch file. The QUIT command does not end your session. Your copy of CANDE continues to execute in the mix and awaits subsequent commands.

Syntax



Explanation

ABORT QUIT is a synonym for ABORT. Both commands share identical syntax and function.

DO	The DO option terminates the processing of the executing batch file. All currently executing tasks, including those initiated by the batch file, continue to execute.
<u>COMPILE</u>	This option terminates only those tasks that are awaiting initiation due to a locked file. The COMPILE option aborts the auto-retry mechanism for a locked file. For further information on auto-retry, refer to "COMPILE Command."

REBUILD Command

Use this command to rebuild an editor file that is corrupted beyond the recognition of the editor or to save the modifications made since the file was last backed up or saved.

Syntax

```
-- REBUILD [ <file title> ] [ AS <file title> ]
```

Explanation

<file title>	This is the name and optional family name of the editor file you want to rebuild. The default file title is that of the active editor file. For the syntax of the file title, refer to the beginning of this section.
AS <file title>	This specifies an alternate file name for the rebuilt file, so you can keep the original editor file. If you do not use this option, CANDE rebuilds the editor file under the same file name and purges the input editor file. For the syntax of the file title, refer to the beginning of this section.

Details

The REBUILD command invokes the CANUTL utility. CANUTL reads the file sequentially and requires only that the header record be valid. CANUTL ignores all pointer links.

RECALL Command

Use this command to replace the active editor file with its associated recall file.

REDO Command

Syntax

RECALL —————
 └─┬──────────┘
 <file title>

Explanation

<file title> This parameter is used when there is not an active editor file. For the syntax of file title, see the beginning of this section.

Details

CANDE makes a recall file for an editor file if you specified the RECALL option with the GET command. If the file name is ABCD and your user ID is aa, CANDE names the editor file ABCDaa. CANDE names the associated recall file AaaBCD.

REDO Command

Use this command to display the last command you entered.

Syntax

REDO —————

Details

The REDO command displays the last command you entered. You can then edit the command and retransmit it. The REDO command can save you time if you are entering similar commands repeatedly, or if you made a mistake in entering a command.

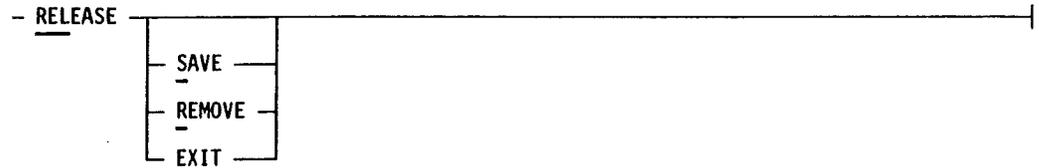
The REDO command displays up to 500 characters of the last command.

The REDO command will not display the AJ, WJ, WY, DQ, or MX commands because these are inquiry commands. If the last command was one of these, REDO shows the previous command.

RELEASE Command

Use this command to close the active editor file (if open).

Syntax



Explanation

<u>SAVE</u>	This updates the recall file when the active editor file is closed.
<u>REMOVE</u>	This removes the recall file when the active editor file is closed.
<u>EXIT</u>	This leaves the recall file present but does not update it.

Details

The effect of the **RELEASE** command on the active editor file is as follows:

- If the file retention type is **PERMANENT**, the editor file is kept in your library with any new changes.
- If the file retention type is **TEMPORARY**, the editor file is purged.
- If a recall file is associated with the editor file, its disposition is determined by the parameter specified with the **RELEASE** command.
- If you do not specify a parameter with the **RELEASE** command, the action on the recall file is as follows:
 - If the **AUTOREMOVE** option of the **SESSION** command is **TRUE**, or the **AUTOSAVE** option of the **SESSION** command is 0 (zero), then the recall file is removed.
 - If the **AUTOSAVE** option of the **SESSION** command is 1, the recall file is exited.
 - If you specified the **RECALL** option with the **GET** command, and the **AUTOSAVE** option of the **SESSION** command is greater than 1, the recall file is updated.

REMOVE Command

Use this command without a file title to remove the active editor file and its associated recall file. Use this command with a file title to remove an editor file other than the active one.

REORGANIZE Command

Syntax

REMOVE <file title>

Explanation

<file title> This removes an editor file other than the active editor file. The associated recall file is not removed. You can include masking characters in the file name. For the syntax of file title, see the beginning of this section.

REORGANIZE Command

Use this command to reorganize records in your editor file to conserve storage space. This command also eliminates deleted blocks from the editor file which can increase the operating speed of the editor. You should use this command if you have performed numerous deletes, merges, or moves.

Syntax

REORGANIZE <file title> AS <file title>

Explanation

<file title> This is the file title of the file you want to reorganize. If you do not include a file title, the active editor file is assumed. For the syntax of the file title, refer to the beginning of this section.

AS <file title> This is an alternate name under which you want to reorganize the editor file. Using this option retains the original editor file. For the syntax of the file title, refer to the beginning of this section.

Details

To rebuild a corrupted editor file, you should use the REORGANIZE command first, and then use the REBUILD command.

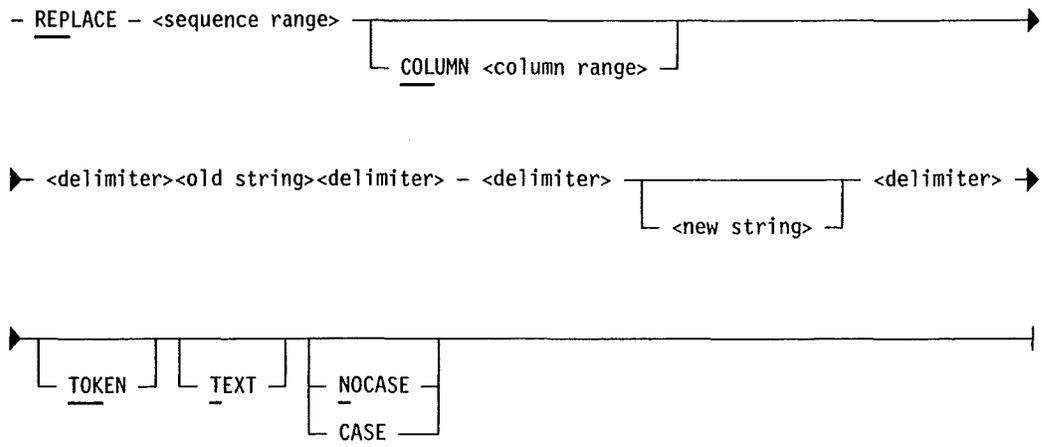
The REORGANIZE command invokes the CANUTL utility. CANUTL reorganizes the file and overwrites the input file, unless you specify an alternate file name with the AS *<file title>* syntax.

If the CANUTL utility detects a record count mismatch between the original input file and the reorganized output file, and the original file is usercoded, the reorganization automatically restarts with switch 1 equal to 2. If you do not specify an output file title, CANUTL creates a file name by taking the original file name and placing the 2-character user ID in the second and third positions of the file name. For example, if the file name is ABCDaa, CANUTL creates an output file name of AaaBCD. This prevents the input file from being overwritten. The file AaaBCD is the result of executing CANDE with switch 1 equal to 2.

REPLACE Command

Use this command to insert, change, or delete characters in the text lines by replacing a specified text string or column range with a new string.

Syntax



Explanation

- `<sequence range>` This parameter specifies the range of records in which you want the replace to occur. For the syntax of the sequence range, see the beginning of this section.
- `COLUMN <column range>` This option allows you to insert a new string starting at a specified column (specify a start column) or to replace certain columns with a new string (specify both a start and an end column).
- `<delimiter>` This character marks the beginning and the end of each string and cannot appear within the strings. You can use any special character as the delimiter except an ETX. Use the same delimiter character to mark the beginning and the end of each string.

REPLACE Command

<old string>	This is the string of text you want to replace.
<new string>	This is the string of text you want to use in place of the old string. The new string can be longer or shorter than the old string. The trailing characters are shifted to the right or left accordingly.
<u>TOKEN</u>	This causes only matching token strings to be replaced. The absence of this option replaces all matches, even if contained within another string.
<u>TEXT</u>	<p>This option lists the lines modified by the REPLACE command. If more lines are changed than will fit on the screen, the system pauses after displaying each full page. To continue with the display, transmit a blank character.</p> <p>You can interrupt this option without halting the replace operation by transmitting <i>NOLIST</i> after the first page of changes has been displayed. The replace operation continues until all changes have been made, then the following message is displayed to verify that the command has been completed: <i>nn REPLACED</i>, where <i>nn</i> is the number of times the string was replaced.</p> <p><i>Note: If you specify TEXT for a multipage replace operation, transmit only a blank character to cause the next page of replacements to occur. The transmitting of any other character is interpreted as a new command.</i></p>
<u>NOCASE</u>	The text in the <old string> is not case sensitive.
<u>CASE</u>	The text in the <old string> is case sensitive.

Details

A line cannot exceed the text length for the editor file kind you are using. Table 2-1 lists the text lengths for the file kinds valid with CANDE.

If you attempt to exceed the permissible length of a line by replacing a string with a longer one, or by inserting a string, an error message will be displayed, and the replacement will not be made in that line. The valid column ranges for valid CANDE file kinds are listed in Table 2-1.

Note: The comment field of COBOL files (columns 73-80) is not shifted or affected by string or column replacements, unless the specified column range overlaps the comment field. In this case, the comments are affected.

Example

REPLACE ALL "G-VALU-01" "G-VALUE-01"

This example replaces all occurrences of G-VALU-01 with G-VALUE-01.

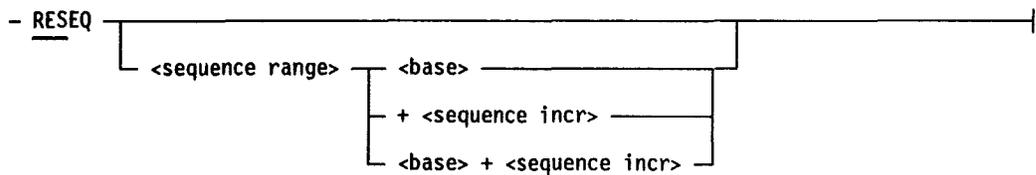
REPLACE 1000-2000 COL 8-10 "01" "02"

This example replaces all occurrences of 01 with 02 in columns 8 through 10 of lines 1000 through 2000.

RESEQUENCE Command

Use this command to resequence all or part of the lines in the editor file.

Syntax



Explanation

- <sequence range> This specifies the lines that you want to resequence. For the syntax of the sequence range, refer to the beginning of this section.
- <base> This is the beginning line number for the lines to be resequenced. If you do not assign a value to the base, the base value assigned by the SEQBASE option of the SESSION command is used. If no base value was assigned with the SESSION command, the CANDE default value of 100 is used.
- + <sequence incr> This is the resequencing increment between line numbers. The increment is an 8-digit value. A space must precede the plus sign (+). If you omit the sequence increment, the increment assigned by the SEQINC option of the SESSION command is used. If no increment was assigned with the SESSION command, the CANDE default value of 10 is used.

RLOG (run log) Command

Details

The RESEQUENCE command does not change the relative position of the lines in the editor file. The resequencing is done only if the new range of line numbers does not overlap any preceding or trailing line numbers. If the new range does overlap, the following message is displayed:

```
RESEQUENCING NOT DONE, MAXIMUM INCREMENT = nnnnnn
```

The increment displayed is the maximum that can be used without causing overlap.

Example

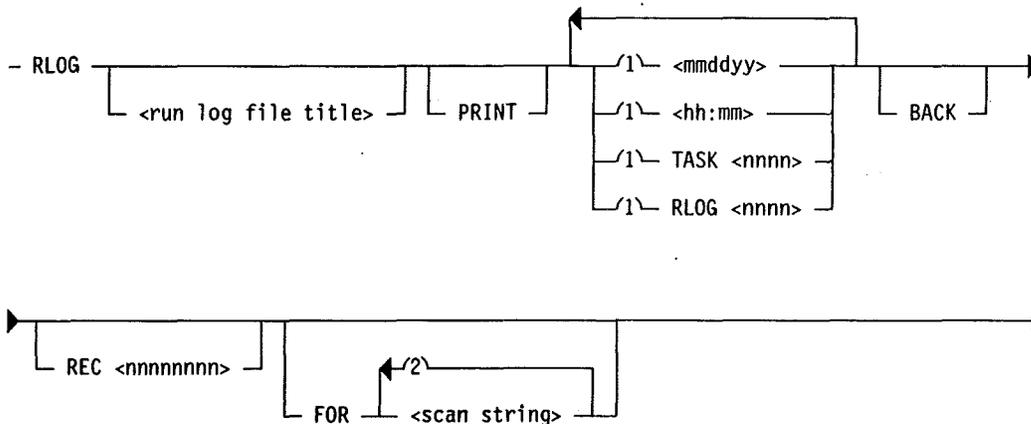
```
RES ALL,1000 +100
```

This example resequences all the lines in the active editor file, assigning the beginning line number as 1000 and incrementing each line by 100.

RLOG (run log) Command

Use this command to display or print selected portions of the run log file maintained by the operating system. The run log file contains information on system operation.

Syntax



Explanation

`<run log file title>`

This specifies the name of the run log you want to display or print. The CANDE usercode override syntax, in which a colon is required to follow the 4-digit usercode, cannot be used because it conflicts with the

hh:mm time syntax. The default log file name is the current log file (RLOGp0 where *p* is the processor number). The default family is DISK.

- PRINT** This option sends the output of the RLOG command to the CANDE job log file. To print this file, use the CANDE PLOG command and then the PRINT system command. The PRINT option requires at least one of the following options: <mmddy>, <hh:mm>, TASK, or RLOG.
- <mmddy> This option selects a beginning date. The display or print of the run log begins with records created on or after this date (if the run log contains records created over many days).
- <hh:mm> This option selects a beginning time. The display or print of the run log begins with records created on or after this time. The colon (:) is required.
- TASK <nnnn>** This option selects records pertaining to a particular task number. The <nnnn> value is the 4-digit task number.
- RLOG <nnnn>** This option selects records pertaining to a particular run log number. The <nnnn> value is the run log number. The difference between a task number and a run log number is that run log numbers are not reused within a run log, whereas task numbers are recycled as they become available.
- BACK** This option causes the search or display of log records to begin with the last record in the run log file and proceed backward toward the beginning of the file.
- REC <nnnnnnnn>** This option allows a specific record selection as the starting location for file examination. The <nnnnnnnn> value is the record number.
- FOR <scan string>** This option specifies up to 3 strings, of up to 30 characters each, that are to be used for selecting matching records. Records that contain the specified string are selected for display or printing. If you specify more than one string, any record that contains at least one of the strings is selected. Masking characters are not permitted.

If you want to include blanks or special characters in a string, enclose the entire string in quotation marks (").

RLOG (run log) Command

Details

Once a search is initiated, it can be terminated by entering a new command or by entering the ?BRK command. CANDE checks every 500 records for a new command. If PRINT was specified, the early termination is recorded in the CANDE job log.

The RLOG command displays only the following run log record types:

Number	Record Type
0/1	File close remove
0/2	File removed
2/0	End of job
2/1	Pseudo end of job
5/0	Log comment
5/1	PATCH
6/0	Beginning of job

run log record types and subtypes are displayed in columns 78 through 80. The run log entry types are discussed in the *V Series System Software Logging Operations Reference Manual*.

The RLOG command can display the entire run log, but it can print only selected portions. To print the entire run log, use the RLGOUT utility of the master control program (MCP).

Transferred run log files are given the names RLOGp1 through RLOGp9, where *p* is the processor number. The operating system creates these files with the security attribute PRIVATE (DFSC=1). Therefore, CANDE must be executing under a PRIVILEGED accesscode to view these files.

Example

```
RLOG FOR MYPROG "OBJECT/MYPROG" "OBJECT MYPROG"  
      string      string      string  
      1           2           3
```

This example causes the display of those run log records that contain any of the following strings:

MYPROG

OBJECT/MYPROG

OBJECT MYPROG

RMERGE Command

Use this command to merge an editor file with the active editor file. Lines in the active editor file are replaced with those in the merge file only when the line numbers are identical. The file kind of both files must be the same.

Syntax

```
- RMERGE <file title> [ <sequence range> ] [ NOPATCHID ]
```

Explanation

- | | |
|------------------|--|
| <file title> | This is the name and optional family name of the editor file that you want to merge with the active editor file. For the syntax of the file title, refer to the beginning of this section. |
| <sequence range> | This optionally specifies a range of lines that you want to merge. For the syntax of the sequence range, refer to the beginning of this section. |
| <u>NOPATCHID</u> | This prevents merged lines from being marked, even if you have specified a patch ID with the PATCHID command. |

Details

If you do not specify the NOPATCHID parameter, the marking of merged lines is under the control of the PATCHID command. Only merged records containing all spaces or a previous patch ID are updated with the current patch ID.

RUN Command

Use this command to execute a program in the normal task mix and cause run time messages to appear only at your workstation. A suggested use of this command is as a debugging tool when developing programs.

Syntax

```
- RUN <program name> [ <execution parameters> ]
```

Explanation

- | | |
|----------------|--|
| <program name> | This specifies the name of the program to be executed. |
|----------------|--|

SAME Command

[<execution parameters>] This specifies optional parameters used in the execution of the program, such as file equations.

Details

The RUN command opens the CANDE job log (if not already open) and writes responses to the job log so that you can review the responses if you desire. (Once the messages are removed from the workstation display, they cannot be retrieved from the ODT log.)

Example

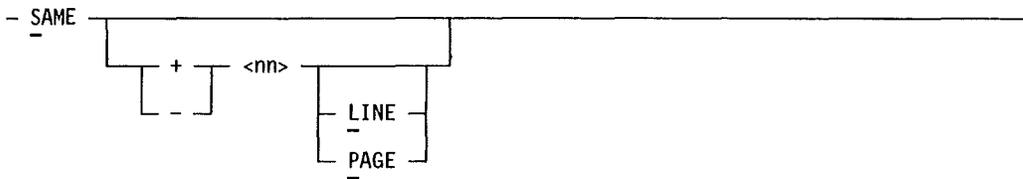
```
RUN VCSCVB FILE BINARY MYPROG
```

This example invokes the VCSCVB program and displays resulting messages only at the user's workstation.

SAME Command

Use this command to redisplay the current page.

Syntax



Explanation

- +** This defines a forward page direction. A space must precede the plus sign (+).
- This defines a backward page direction. A space must precede the minus sign (-).
- <nn>** This is a value from 0 through 99 that indicates the number of pages or lines to display forward or backward from the current page. (The default unit is lines. To display pages, use the PAGE option with this command.) If you omit the value, the default value of zero is used.
- LINE** This displays the editor file in lines. LINE is the default unit.

PAGE This displays the editor file in pages. The default unit is lines.

Details

The DISPLAY SCROLL DOWN option of the WORKSTATION command has no effect on this command.

SAVE Command

Use this command by itself to save your active editor file to its associated recall file. (The associated recall file is overwritten without warning.) Use this command with a file title to save your active editor file under an alternate file title or to save nonusercoded files.

Syntax

```
- SAVE [ AS <file title> ]
```

Explanation

<file title> The file name and optional family under which you want to save the active editor file. If you do not specify a family name, the default family is assumed. For the syntax of the file title, see the beginning of this section.

If a file with the same name exists, and the AUTOREMOVE option of the SESSION command is TRUE, the file is overwritten. Otherwise, CANDE aborts the command and displays an error message.

Details

When this command is successfully completed, the following message will be displayed:

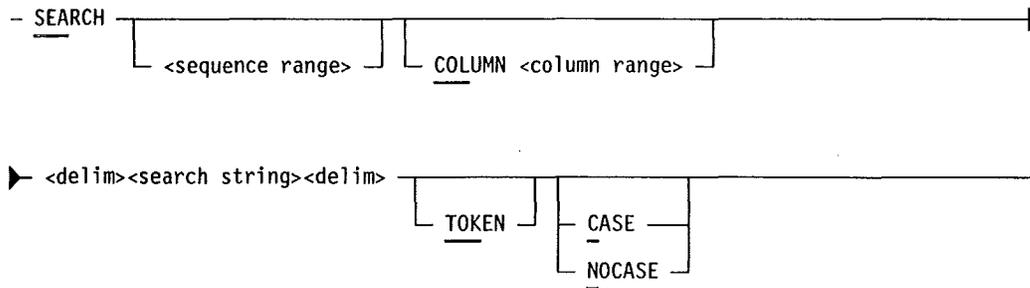
Active editfile saved as <filetitle>

SEARCH Command

Use this command to search the editor file for a specified text string.

SEARCH Command

Syntax



Explanation

- <sequence range>** This is the range of editor file lines you want to be searched. If you do not specify a value, CANDE searches all lines in the file. Refer to the beginning of this section for further information on sequence ranges.
- COLUMN <column range>** This narrows the scope of the search to the indicated column range. If no column range is selected, the defaults of the first and last text locations for the file kind are used.
- <delim>** This element is used to delimit the ends of the string to be searched for and must not appear in the string itself. You can use any special character as a delimiter, except an ETX. If case sensitivity is required, use double quotation marks (") as delimiters.
- <search string>** This is the group of characters you want to search for. The maximum string length is 80 characters.
- TOKEN** This indicates that the string must be delimited by spaces in the file to be identified as the string.
- CASE/NOCASE** You can use this parameter to determine whether your SEARCH command execution will be case sensitive. The default is controlled by the CASE option of the SESSION command.

Details

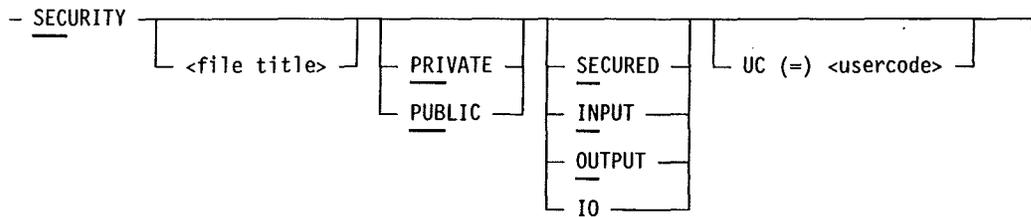
The SEARCH command differs from the FIND command in that the FIND command returns only the matching records, and the SEARCH command returns the page of text that contains the match.

Once a match has been found, you can search for additional matches by using the *NEXT SEARCH* command. You can enter this command in shortened form as *N SEA*.

SECURITY Command

Use this command to change the security attributes of a file.

Syntax



Explanation

<file title>	This allows you to specify the editor file for which you want to set file security. For the syntax of file title, see the beginning of this section.
<u>PUBLIC</u>	This option allows any user to access the file.
<u>PRIVATE</u>	This option allows only the person who created the file to access it.
<u>SECURED</u>	This option indicates that the file is a program and cannot be opened as a data file.
<u>INPUT</u>	This option permits read-only access to the file.
<u>OUTPUT</u>	This option permits write-only access to the file.
IO	This option permits read and write access to the file.
UC (=) <usercode>	This option lets you give ownership of the file to another usercode. If the file is private, and MCP file security is enabled, the file will be accessible only by a CANDE session executing under a matching usercode. The equal sign (=) is optional.

SEQUENCE Command

Details

The SECURITY command allows you to change the security attributes of any file created under your own usercode, regardless of the current security attributes of the file. You cannot change the security attributes of private files not associated with your usercode.

If you do not declare either PUBLIC or PRIVATE when you make an editor file and if the system is running SECURITY, the mode specified for you in the USERHO file will be used as the default. If the system is not running SECURITY, the default is PUBLIC.

The access security level of a file cannot be changed when the file is in use.

If you do not specify the security use (SECURED, OUT, IN, IO), the current security use of the file remains unchanged.

Note: The editor usually opens files with IO permissions. Thus, you should not use a security use other than IO for editor files that you intend to open as active editor files with the GET command.

SEQUENCE Command

Use this command to direct CANDE to automatically insert sequential line numbers into the editor file.

Syntax

```
- SEQUENCE _____  
  |_____| |_____|  
  <base>  + <sequence incr>
```

Explanation

- | | |
|-------------------|---|
| <base> | This indicates the starting line number. If you do not specify a value for base, the default value of 100 is used. |
| + <sequence incr> | This is the resequencing increment between line numbers. The increment is an 8-digit value. A space must precede the plus sign (+). If you omit the sequence increment, the increment assigned by the SEQINC option of the SESSION command is used. If no increment was assigned with the SESSION command, the CANDE default value of 10 is used. |

Details

If the base and increment values cause new line numbers to be the same as old line numbers, the text associated with the old line numbers is displayed so that you can review the text and avoid accidentally overwriting data.

The range of allowable line numbers is 1 through the highest number that will fit in the sequence field of the file kind; see Table 2-1 under "MAKE command."

Blanks are not required between line numbers and text.

When you transmit any valid command or a blank, the SEQUENCE mode will end. Transmitting from the end of any other line will transmit the text and cause new line numbers to be displayed. The top line of the display will always be the last line that you transmitted.

Example

Assume that your editor file currently looks like the following:

```
00100200This example shows how the sequence command  
00100300automatically generates line numbers.
```

You enter the SEQUENCE command as follows:

```
S100200 +10
```

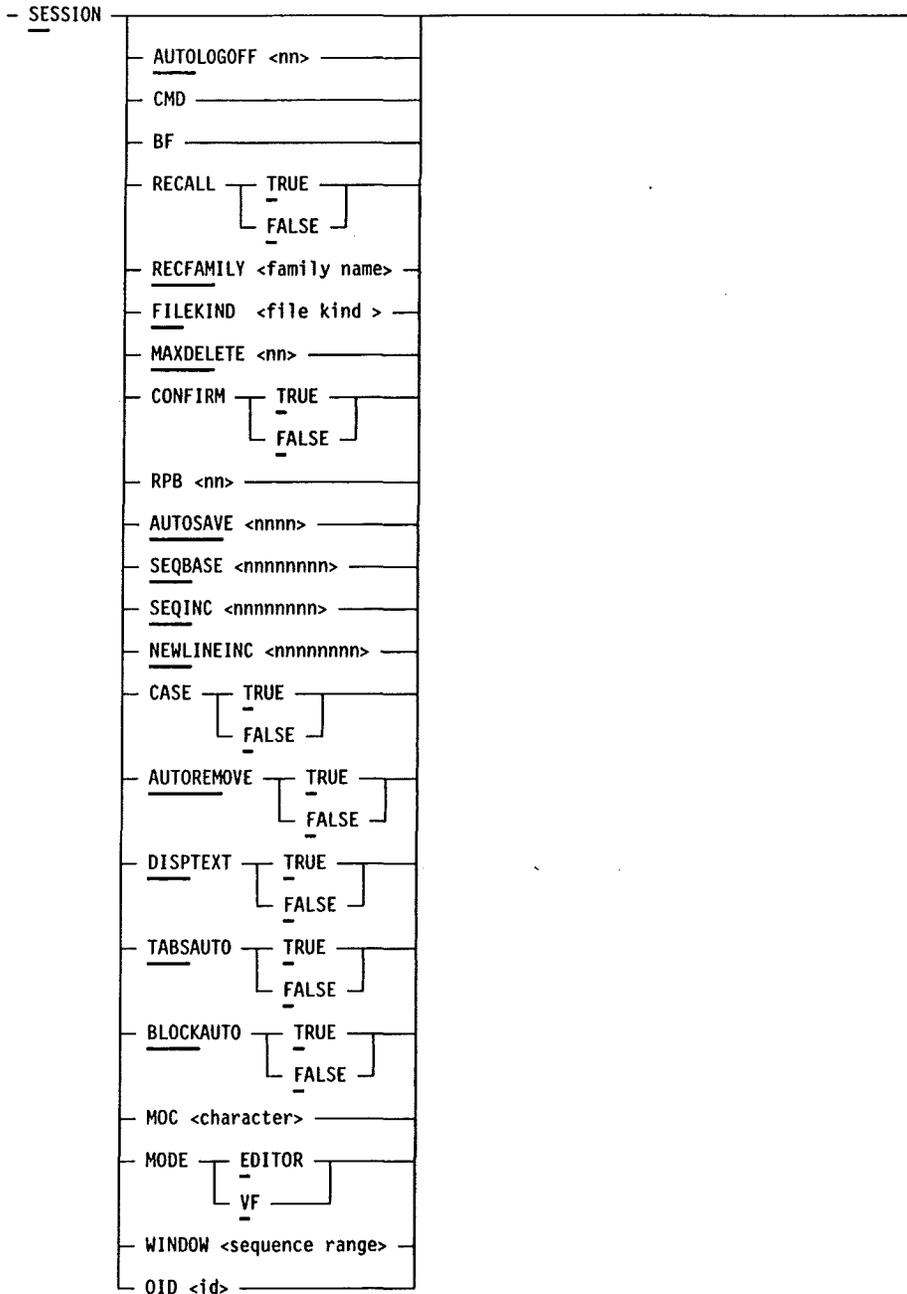
The system responds with the following screen:

```
00100200This example shows how the sequence command  
00100210  
00100220  
00100230  
00100240  
00100250  
00100260  
00100270  
00100280  
00100290  
00100300automatically generates line numbers.
```

SESSION Command

Use this command to establish operating parameters and default parameter values for the duration of your CANDE session.

Syntax



Explanation

AUTOLOGOFF <nn> This option lets you modify the automatic log-off time already assigned to your session. The value of <nn> is 1 through 99 minutes.

CMD This option displays the last 10 commands you entered, showing a maximum of 80 characters for each command. To redisplay only the last command you entered, use the REDO command.

BF This option displays the titles of the last 15 printer, punch, and dump files generated by tasks initiated during this session. This option also tracks file name changes made with the CHANGE system command.

The format of the display is as follows:

```

SESSION USERID = c&
SESSION USERCODE = CANDE (PRIVILEGED)
SESSION CHARGE # = 000000
USERNAME = CANDE2
INSTALLATION # = 0001
HOSTNAME = PASHOSTV
AUTOLOGOFF = 099
TANKING MESSAGES = OFF
HARDCOPY = OFF
SESSION FAMILY = DATCOM
AUTO DISPLAY MODE = OFF
SEQ BASE = 100
SEQ INC = 10
NEW LINE INC = 10
MODEOVERRIDECHAR = .
OVERRIDE USERID =
CURRENTLY PROCESSING BATCH FILE...
ATTEMPTING TO INITIATE COMMAND...

RECALL = TRUE
RECALL FAMILY = VCS2
FILEKIND = BPL
MAXDELETE = 06
CONFIRM = FALSE
REC PER BLOCK = 00
AUTOSAVE = 0050
AUTOREMOVE = FALSE
CASE = FALSE
DISPLAY TEXT = TRUE
TABSAUTO = TRUE
WINDOW = 0-99999999
    
```

The CURRENTLY PROCESSING BATCH FILE line appears only if a batch file is active.

The ATTEMPTING TO INITIATE COMMAND line appears only when the system is attempting to initiate a locked file.

- RECALL** When TRUE, this option causes CANDE to make a recall file when you enter commands that use the RECALL and NORECALL parameters, like MAKE and GET. When FALSE, CANDE does not make a recall file. The value for this option is used only if you do not specify RECALL or NORECALL for applicable commands. The default setting for this option is TRUE.
- RECFAMILY <family name>** This option lets you specify a default family for recall files. The specified family is always used, unless you override the family name with the GET command syntax. If you do not specify a default family for recall files, the files are stored on the current session default family.
- FILEKIND <file kind>** This option lets you specify a default file kind for the MAKE command.
- MAXDELETE <nn>** This option lets you specify how many lines the editor will delete before requesting confirmation from you. The value range is 1 through 99 lines. The default value is 5 lines.
- CONFIRM** When TRUE, this option requires you to confirm a DELETE, FIX, or REPLACE command by entering Y or N. For DELETE commands only, when this option is FALSE, any response other than N is an implied Y. The default value is TRUE.
- RPB <nn>** This lets you specify a records-per-block value to be used with the COPY command. The value range is 1 through 99.
- AUTOSAVE <nnnn>** This specifies the number of input lines after which CANDE automatically updates the recall file. The RECALL option must be TRUE in order to use this option. The value range is 1 to 9999. If the active editor file is larger than 1000 records, CANDE displays the following message:
- Please wait while recall file <file title> is created/updated.
- SEQBASE <nnnnnnnn>** This option lets you specify a base sequence number value that is used in all commands where a base sequence number is applicable but not specified. The value range is 0 to 99999999.

- SEQINC <nnnnnnnn> This option lets you specify a sequence number increment that is used in all commands where a sequence number increment is applicable but not specified. This value is also used as the increment for generating new sequence numbers for source text entered without sequence numbers after the last sequence number in the file. The value range is 0 through 99999999.
- NEWLINEINC
<nnnnnnnn> This option specifies the sequence number increment to use when adding source text with no sequence numbers. If the new source line number is greater than the last established sequence number in the file, the SEQINC value is used.
- If the value of NEWLINEINC is too large, a dynamic calculation is made using the available sequence range divided by the total number of blank sequence number input lines. If the resulting value is less than one, an error message is displayed along with all blank sequence lines in question. The default value of NEWLINEINC is 10.
- CASE When this option is FALSE, text is not case sensitive for the FIND, FIX, and REPLACE commands. All text is translated to uppercase characters. When TRUE, the text is case sensitive. The default value is FALSE. The value for this option is used only if you do not specify the CASE or NOCASE option with each of the applicable commands.
- AUTOREMOVE When TRUE, a duplicate file is removed when encountered during a command. When FALSE, CANDE aborts the action and displays an error message when a duplicate file is encountered. The default value is FALSE.
- DISPTEXT When TRUE, this option displays the sequence number and the matching text line for the FIND, MERGE, REPLACE, and RMERGE commands. When FALSE, this option displays only the sequence number. The default value is TRUE. The value for this option is used only if you do not specify the TEXT or NOTEXT option with each of the applicable commands.
- TABSAUTO When TRUE, this option causes CANDE to examine an editor file opened with a GET command for valid tab settings and load those settings to the user's workstation. When FALSE, CANDE takes no action on the tab settings. The default value is FALSE.

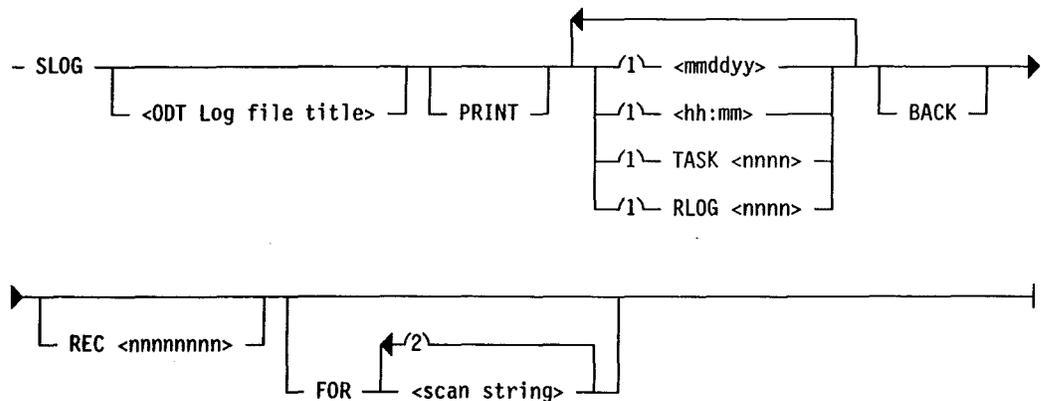
SLOG (ODT Log) Command

- BLOCKAUTO** When TRUE, this option causes the sequence range defined with the BLOCK command to be updated with the sequence range specified in the FIND, FIX, LIST, PAGE, or REPLACE commands. When FALSE, the block sequence range is not updated. The default value is FALSE.
- MOC <character>** This option lets you specify a mode override character to indicate that a particular command is directed to the View File (VF) utility when in CANDE mode; or to CANDE when in VF mode. (You change modes by using the EMODE and VMODE commands.) The default mode override character is a period (.). For example, if you are in CANDE, entering *.FORM* puts VF in forms mode.
- MODE** This option indicates the mode, editor or VF, of your current session. Editor mode indicates that all commands and input are to be processed by the editor. VF mode indicates that all commands are to be passed to the VF utility. To switch to the opposite mode, you can use the mode override character on each screen or command. The default mode is editor.
- WINDOW <sequence range>** This option lets you define a sequence range to be used with commands that perform addition, modification, and deletion operations. If a record is outside this sequence range, CANDE displays an error message and ignores all further input of the same transmission. The default sequence range is 0 to 99999999.
- OID <id>** This option lets you specify a 2-character user ID to be used as the session user ID for all nonusercoded file names. Assigning an override user ID for the session prevents you from having to constantly provide a usercode override statement as part of command syntax or from having to restart the session under a different accesscode. All file name default references to the user ID will be overridden with the value of <id>.

SLOG (ODT Log) Command

Use this command to display or print selected portions of the Operator Display Terminal (ODT) log file. (The ODT log file contains the messages that have been displayed on the ODT.)

Syntax



Explanation

- <ODT log file title>** This element specifies the name and optional family of the ODT log file you want displayed or printed. The CANDE usercode override syntax, in which a colon is required to follow the 4-digit usercode, cannot be used because it conflicts with the hh:mm time syntax. The default log file name is the current log file (SLOGp0, where *p* is the processor number); the default location is DISK.
- PRINT** This option sends the output of the SLOG command to the CANDE job log file. (After the SLOG command finishes, you can print the job log file by using the PLOG command and then the PRINT system command.) The PRINT option requires at least one of the following options: <mmddy>, <hh:mm>, TASK, or RLOG.
- <mmddy>** This option selects a beginning date. The display or print of the ODT log file begins with records created on or after this date (if the ODT log file contains records created on more than one day).
- <hh:mm>** This option selects a beginning time. The display or print of the ODT log file begins with records created on or after this time. The colon (:) is required.
- TASK <nnnn>** This option selects records that pertain to a particular task number. The <nnnn> value is the 4-digit task number.
- RLOG <nnnn>** This option selects records pertaining to a particular run log number. The <nnnn> value is the run log number. The difference between a task number and a run log number is that run log numbers are not reused

SLOG (ODT Log) Command

within a run log, whereas task numbers are recycled as they become available.

BACK <nnnnnnnn> This option causes the search or display of log records to begin with the last record in the ODT Log file and proceed backward, toward the beginning of the file. The <nnnnnnnn> value is the sequence number of the record.

REC <nnnnnnnn> This option lets you specify a particular record as the starting location for a file examination. The <nnnnnnnn> value is the sequence number of the record.

FOR <scan string> This option specifies up to 3 strings, of up to 30 characters each, that are to be used for selecting matching records. Records that contain the specified string are selected for display or printing. If you specify more than one string, any record that contains at least one of the strings is selected. Masking characters are not permitted.

If you want to include blanks or special characters in a string, enclose the entire string in quotation marks.

Once a search is initiated, it can be terminated by entering a new command or by entering the **BREAK** command. **CANDE** checks every 500 records for a new command. If **PRINT** was specified, termination is recorded in the **CANDE** log.

Details

The **SLOG** command can display the entire ODT log, but it can print only selected portions. To print the entire ODT log, use the **SLGOUT** utility of the **MCP/VS** operating system.

Example

```
SLOG FOR MYPROG "OBJECT/MYPROG" "OBJECT MYPROG"  
      string      string      string  
      1           2           3
```

The preceding example displays only those ODT log records that contain any of the following strings:

MYPROG

OBJECT/MYPROG

OBJECT MYPROG

START Command

Use this command to initiate and monitor the compilation of either the active editor file or another source file.

Syntax

```

  [ START ] [ COMPILE ] [ <file title> ] [ <sequence range> ]

```

Explanation

<u>COMPILE</u>	COMPILE is a synonym for START. Both commands share identical syntax and function.
<file title>	This specifies a file other than the active editor file that contains the commands to be executed. For the syntax of the file title, see the beginning of this section.
<sequence range>	This option lets you specify a sequence range within the CANDE editor file that will be affected by the START command.

Details

The START command allows you to control functions from your workstation that would otherwise be performed with the pseudo card reader facility or with the file-equated execution function of CANDE. The START command initiates a task by extracting standard control commands from the beginning of an input file. The input file can be either an editor file (500 bytes per record) or a system file (80 bytes per record, 1-10 records per block).

The compiler control information or job control information for each file can be in either of two formats:

- CANDE file-equated execution format
- Pseudo card reader format

The CANDE file-equated execution format is described later in this discussion. Pseudo card reader format is described in the *V Series MCP/VS System Software Operations Guide, Volume 2: System Commands*. (If you are running release level 3.0 or later of the MCP, refer to the *V Series Systems Operations Guide, Volume 2: System Commands*.)

When you issue a START command, CANDE verifies the presence of the specified file and the file kind, and then submits the file for compilation using the

START Command

appropriate file equation or pseudo card reader (PCR) file generation. CANDE determines the compiler to be used based on the file kind of the input file.

If the input file size exceeds 500 records, processing is passed to the CANDE utility program CANUTL for background execution. This step enables you to continue working at your station without interruption. If the CANUTL utility program is unavailable, CANDE processes the file.

If the file is locked, CANDE displays the following message:

```
FILE IN USE BY TASK <task number> ON PROC# <processor number> WILL  
REPEATEDLY RETRY FILE INITIATION EVERY 5 MINUTES
```

CANDE attempts to recompile the source file every 5 minutes until the file is unlocked or until an ABORT COMPILE command is issued.

If the input file is a batch file, the commands in it will be replayed. In this case, the START command is synonymous with the DO command and must be marked by the LABELBEGIN or BEGINJOB batch file commands (see "DO Command").

Source File Format

The input file must contain all MCP control records, compiler control (dollar) records, and source text needed for the compilation.

File-equated compiler control records (formerly referred to as COMPTS format records) must meet the following criteria for CANDE to recognize them:

- They must be the first records in the file.
- They must begin with the comment indicator followed by a question mark (?). The comment indicator is unique to each compiler, as shown in Table 2-1.
- They, as well as the comment indicator, must correspond to the file kind of the input file.

If the comment indicator is missing or is unrecognizable, or if the file kind does not match the comment indicator, CANDE displays the control records and the following message:

```
INCORRECT FILE TYPE OR MISSING COMMENT INDICATOR  
ON CONTROL CARD,  
COMPILER REQUEST ABORTED
```

The MCP control commands must precede all source text in the file. Dollar records can precede the MCP control instructions as illustrated in the examples that follow.

If the file is a patch file, the first record must be a dollar record that indicates to the compiler the medium on which the source file resides.

The file kinds and required comment indicators are shown in Table 2-2.

Table 2-2. File Kinds and Comment Indicators for Source Files Used in CANDE File-Equated Execution

File Kind	Comment Indicator
ADS	# COMMENT...#
BINDER	%
BPL	&
COBOL, COBOLV	*
DASDL	%
DATA	.
FORTRAN	C or *
PASCAL	(*...*)or {...}
RPG	*
WFL	%

Examples

The following examples show input file control statements used to initiate various compilers with the CANDE file-equated execution function. The options illustrated in these examples might not be required in all cases.

BINDER (file kind must be BINDER)

```
1000%?COMPILE TESTER WITH BINDER LIBRARY
1020%?DATA CARD
.(BINDER control statements)
.
.
```

BPL (file kind must be BPL)

```
1000$ LST1 XREF LN60 MTCH STK4
1020&?COMPILE B29U BPL LIBRARY MEM + 10.
1040&?FILE PRINT = FUL96/B29U PBK FORM.
1060&? DATA CARD
1080BEGIN
.
.
.
```

START Command

Note: BPL ignores certain dollar options (for example, XREF) if they are not specified on a leading dollar record. A leading dollar record must precede all other records in the source program, including comments. Since the compiler control records used by COMPTS remain in place in the source file as comments, a leading dollar record must precede the compiler control records, as shown in the example. Refer to the B 2000/B 3000/B 4000/V Series BPL Compiler Programming Reference Manual for more information.

COBOL ANSI-74 (file kind must be COBOL)

```
100100*?CMP TIOB9U COBOL LIB MEM + 100.  
100200*?FILE LINE = FUL96/TIOB9U FORM.  
100220*?FILE DDF = FTRMST/TIODDF.  
100300*?DATA CARD  
100400$LIST LIST$ MAP NOBOUNDS  
100500 IDENTIFICATION DIVISION.  
.  
.
```

DASDL (file kind must be DASDL)

```
1010%?COMPILE TIO DASDL LIB.  
1020%?FILE DDF FTRMST/TIODDF.  
1025%?FILE DDFOLD FTRMST/TIODDF.  
1030%?FILE LINE FUL96/TIO PBK FORM.  
1035%?FILE PRINT FUL96/TIOBND PBK FORM.  
1040%?DATA CARD.  
2000$ SET UPDATE LIST LIST$ LISTBIND BIND MEDIUM CHECKCOBOL  
2100OPTIONS ( STATISTICS SET, ...  
.  
.
```

FORTRAN-77 (FORT77) (file kind must be FORTRAN)

```
1000C?COMPILE FORT FORT77 LIB  
1010C?DATA CARD  
1020$LIST LISTDOLLAR MAP  
1020PROGRAM TEST1  
.  
.
```

Pascal (file kind must be PASCAL)

```
1000(*?COMPILE PASC1 PASCAL LIB.*)  
1010(*?DATA CARD.*)  
1020$LIST LISTDOLLAR MAP  
1020PROGRAM pascl;  
.  
.
```

Note: The comment indicator for Pascal begins with (* and ends with *). The trailing *) must be included. A period must precede the trailing *).

RPG (file kind must be RPG)

```
10100 *?CMP RPGTST RPG LIB.
10104 *?DATA CARD.
10108H 1
10120FCARDIN IP F 80 80 READER
:
```

Note: The RPG comment character, which is an asterisk (*), must appear in column seven. Column six must be blank.

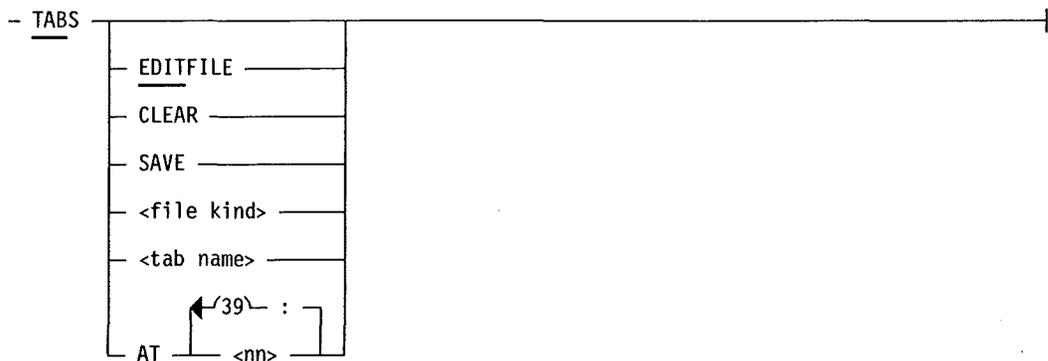
WFL (file kind WFL; compile for LIBRARY or SYNTAX only)

```
1000%?COMPILE WFLJOB WFL ON SYPACK LIB
1010%?FILE LINE FUL96/WFLJOB PBK FORM
1020%?DATA CARD
1030$ LIST LIST$ PAGESIZE = 50
1040BEGIN JOB WFLTST;
:
```

TABS Command

Use this command to establish tab settings for your workstation.

Syntax



Explanation

EDITFILE

This option uses the settings previously established and saved within the active editor file.

TABS Command

CLEAR	This option clears all current tab settings, including any in the active editor file header.
SAVE	This option saves the current tab settings (if any) to the active editor file. The tab settings can then be reloaded with the EDITFILE option of the TABS command. (A newly created CANDE editor file contains no tab settings.)
<file kind>	This option requests that the workstation tab settings be initialized to those appropriate to the file kind of the active editor file. The CANFMT file supplied with CANDE contains default tab settings for all supported file kinds.
<tab name>	This option lets you specify a 1- to 17-character tab name for a set of customized tab settings. CANDE will search for this tab name in the user format file or the CANFMT file.
AT <nn>	This option lets you specify custom tab settings for your workstation. You can specify a total of 40 tab settings. You must use a colon (:) to separate each value, for example, 10:20:35. When specifying tab settings in a batch file, you must put the entire tab setting sequence on one or more contiguous lines. The word TABS must be the first token on each line.

Details

To declare tab settings in the CANFMT or in another format file, use the following syntax:

```
DEFINE TAB <tab name> AT <nn>:<nn>:<nn>:.....<nn>
```

You can specify up to 40 tab settings, indicated by <nn> in the preceding syntax. Your tab definitions must begin at sequence number 10000 or higher in the CANFMT or user format file.

For example, the DEFINE TAB command might look like the following:

```
DEFINE TAB COBOLX AT 1:7:11:15:19:80
```

Note: *When you open an editor file by using the GET command, and the TABSAUTO option of the SESSION command is TRUE, the tab settings last specified for the editor file are automatically loaded.*

TANK Command

Use this command to prevent CANDE from sending messages to your station. Messages for the station are held in a file called the message tank (MSGTANK). The messages are collected and stored until you issue a DETANK command, which returns the stored messages to your station.

Syntax

- TANK _____

Details

While the TANK command is enabled, messages are not entered in the CANDE job log even if the logging option has been enabled with the HARDCOPY command.

The TANK command is automatically invoked if the number of messages sent to the station since the last DETANK command exceeds the workstation page size value.

You set the workstation page size with the PAGE option of the WORKSTATION command. The default page size is 24 lines per page. When you use the DETANK command you are sent a number of messages equal to the workstation page size, and then the TANK command is automatically reinvoked.

The TANK command is not automatically invoked if you have set the workstation page size to zero.

When you receive messages with the DETANK command, the message count is reset to zero.

If you issue the TANK command while the ECHO option of the WORKSTATION command is FALSE, CANDE displays only messages labeled //DISPLAY when processing batch files. You will not see intermediate messages until you issue a DETANK command.

If you issue the TANK command while the ECHO option of the WORKSTATION command is TRUE, CANDE suppresses any further reading of the job manager MSGTANK file, which is needed for batch file processing.

TYPE Command

Use this command to change the file kind of your active editor file or another specified file.

VIEW Command

Syntax

```
- TYPE [ TO ] <file kind> [ <file title> ] [ <text length> ]
```

Explanation

- <file kind>** This parameter lets you change the record format of a file to that of a different file kind. If changing to a new file kind results in a shorter text length, confirmation is required. Once you confirm the change, CANDE invokes the CANUTL utility, which rewrites the entire file to the correct record format.
- <file title>** This parameter specifies the name of the file you want to use with the TYPE command. If you do not specify a file name, the active editor file is assumed.
- <text length>** This parameter specifies the length of text when data or sequential file kinds are specified. The default text length depends upon the kind of file you are using. See Table 2-1 for a listing of the various file kinds and their text lengths.

Details

You can change the file kind of files accessible with your usercode only. If you do not specify a file name, the active editor file is assumed.

The file kind you enter is the type you want to assign to the CANDE file; the file kind can be any of the types listed in Table 2-1 under the MAKE command.

VIEW Command

Use this command to initiate the View File (VF) utility program, which lets you print and view files.

Syntax

```
- VIEW [ <file title> ] [ <printer backup number> ]
```

Explanation

- <file title> This parameter specifies the name of the file you want to use with the VF utility.
- <printer backup number> This parameter specifies the printer backup number of the file you want to use with the VF utility.

Details

You can use the VIEW command to initiate VF from a CANDE session, or you can initiate VF from the MCS (GEMCOS or VCS).

When you initiate the VF utility from CANDE, the values assigned to the MOC, MODE, and OID options of the CANDE *SESSION* command are passed to VF. The values assigned to the PAGE, WIDTH, and OFFSET options of the CANDE *WORKSTATION* command are also passed to VF. You can later change the page size and width values with VF commands if you desire.

Also when you initiate VF from CANDE, screen displays will be in nonforms mode if the file requested for viewing is a printer backup file with a compiler file kind that matches the file kind of the current CANDE editor file. Nonforms mode lets you view a compiler listing and make corrections that will be saved in the editor file. In all other situations, initiating VF from CANDE causes screens to be displayed in forms mode.

If you initiate VF directly from the MCS, screen displays will be in forms mode, so you cannot edit them.

Refer to Section 3 for details on the VF utility and its commands.

VMODE Command

Use this command to change from the CANDE editor mode to the VF utility mode. All subsequent commands will be processed by VF.

Syntax

- VMODE _____|

Details

From VF, you can send a command to CANDE by prefacing the command with the mode override character, which is a period (.) by default. For example, if you enter *.RELEASE* in VF mode, the RELEASE command is processed by CANDE. To change the mode override character, use the MOC command. To change from VF mode to CANDE mode, use the EMODE command.

VOID Command

See Section 3 for more details on the VF utility and its commands.

VOID Command

Use this command to delete individual lines in the editor file.

Syntax

- VOID <sequence number> _____|

Explanation

<sequence number> This is the sequence number of the line you want to delete. You do not need to type the whole sequence number. Instead, you can use only as many characters as necessary to correctly identify the line number.

Details

To easily delete numerous lines, you can place a V at the beginning of each line that you want to delete and transmit the entire screen.

WHATS Command

Use this command to return information about the status of a file.

Syntax

- WHATS _____|
 |
 | <file title> |
 | SHORT _____|

Explanation

<file title> This parameter lets you specify a file to use with the WHATS command. If you do not include a file name, the active editor file is assumed.

SHORT This parameter displays only the file title of the active editor file in the following format:

ACTIVE EDITFILE <filetitle> <records> <file kind >

Details

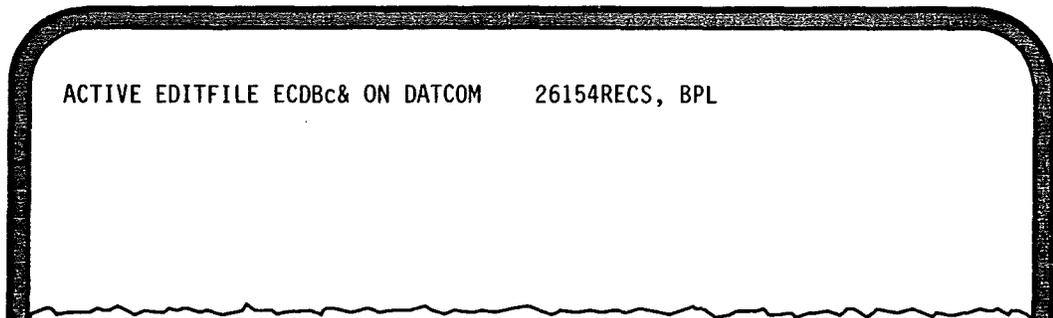
The WHATS command displays the following information about an editor file:

- Any comments that you put in the editor file header
- The time and date that the file was created
- The first and last sequence numbers in the file
- The current record count
- The sequence range defined with the BLOCK command for the current block
- The beginning sequence number of the current page
- The beginning sequence number of the last page
- The beginning sequence number of the previous page
- The current mark ID (if the language file kind is supported)
- The current patch ID (if the language file kind is supported)

Examples

WHATS SHORT

This example displays information about the active editor file that looks similar to the following screen:



W TEST

This example requests information about a file named TEST. The system response to this command looks similar to the following screen:

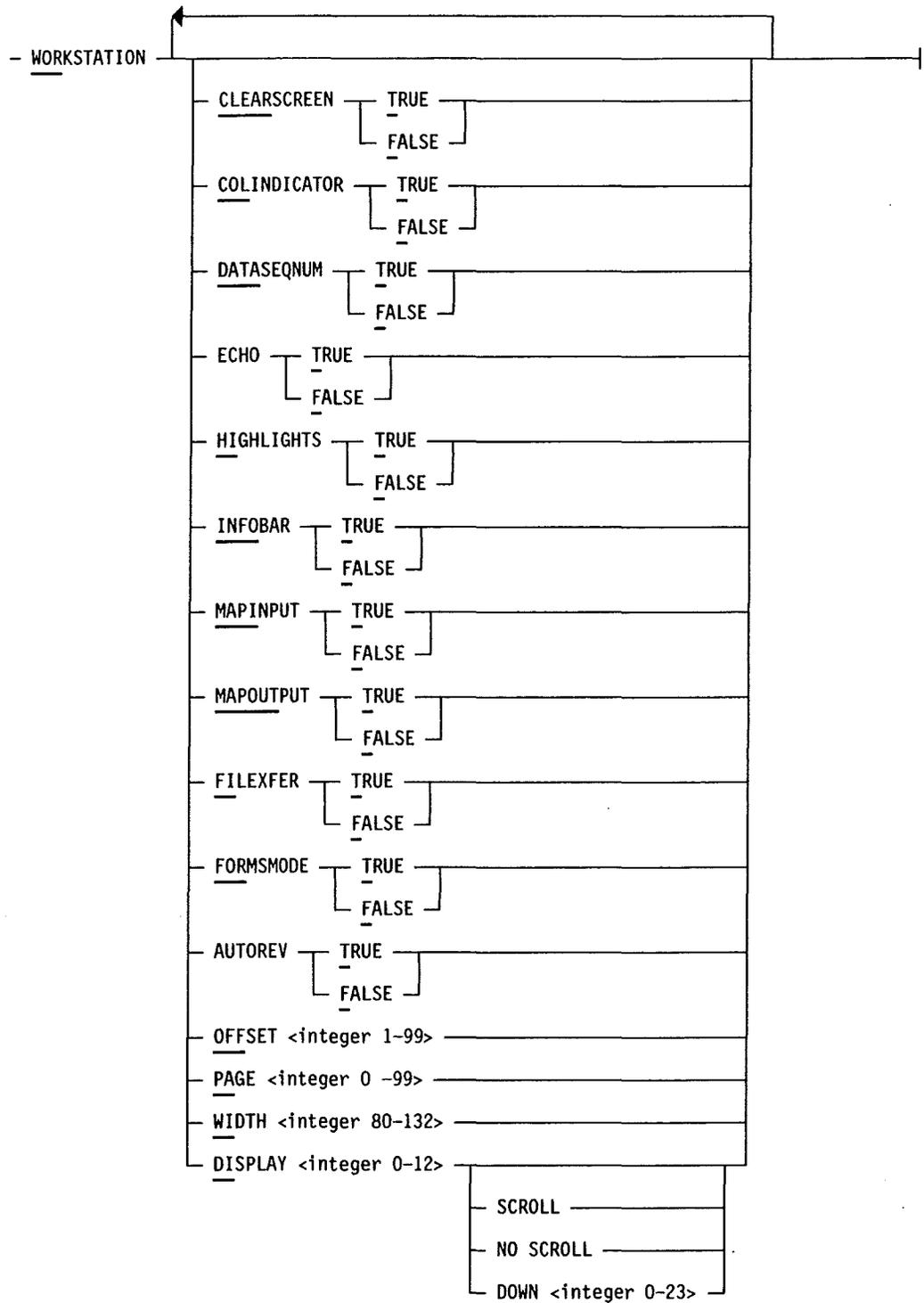
WORKSTATION Command

```
FILETITLE . . . . . TEST ON DATCOM
filekind . . . . . BPL
RECORDS . . . . . 26154 (RPA= 200)
CREATED . . . . .
LAST ACCESSED . . . . 10/22/91 AT 10:48:42
  BY USERCODE. . . . . CANDE
SECURITY TYPE/USE . . PUBLIC /IO
STATUS . . . . . PERMANENT
RECORDSIZE. . . . . 80
SEQNOSIZE . . . . . 8
TEXTSIZE. . . . . 72
SEQRANGE. . . . . 1000-99999999
CURRBLOCK . . . . . 40000000-99999999
SAME SEQNO . . . . . 85442000
LAST EDITED SEQ NBR . 85450000
PREV SEQNO . . . . . 85201110
MARKID . . . . . N/A
PATCHID . . . . . &<DATE>
TABS . . . . . EDITOR FILE HDR
FILE OWNER USERCODE . CANDE
CANDE VERSION 2 SOURCE
```

WORKSTATION Command

Use this command to define the attributes of your terminal. If you enter the WORKSTATION command without options, the current attributes of your terminal are displayed.

Syntax



WORKSTATION Command

Explanation

CLEARSCREEN

This option determines whether the workstation screen is cleared before the next output display. The default value is FALSE.

COLINDICATOR

This option determines whether column indicators are displayed at the top of the screen. The default value is TRUE. Column indicators appear below the first text line and look like the following:

```
-----+-----1-----+-----2-----+-----3-----+---  
00100This is an example  
00200of what column indicators look like.
```

DATASEQNUM

This option indicates whether sequence numbers are displayed for data files when the LIST and PAGE commands are used. The default value is TRUE. The SEQUENCE option of the LIST and PAGE commands overrides the value of this option.

ECHO

This option determines whether batch file commands are displayed on your workstation. The default setting is TRUE. This option also interacts with the TANK command.

If you issue the TANK command while the ECHO option is FALSE, CANDE displays only those batch file messages labeled //DISPLAY. You will not see intermediate messages until you issue a DETANK command.

If the ECHO option is TRUE, issuing the TANK command suppresses any further reading of the job manager MSGTANK file, which is needed for batch file processing.

HIGHLIGHTS

This option turns on and off the video highlighting for special lines. If HIGHLIGHTS is TRUE, special lines are highlighted. The default setting is TRUE.

INFOBAR

This parameter determines whether an information bar is displayed at the bottom of your screen. The default value is TRUE. The information bar provides the following information:

- The name, file kind, and current record count of the active editor file
- The last edited sequence number (LESN)
- The current date and time

MAPINPUT

This option, when TRUE, causes the first 80 characters you enter to be automatically translated to uppercase letters. When FALSE, the characters pass untranslated. This is helpful in working with programs that expect all input in uppercase text. Using a value of FALSE also automatically sets the CASE option of the SESSION command to TRUE. The default setting for the MAPINPUT option is FALSE.

You can transmit lowercase letters when the MAPINPUT option is TRUE by surrounding the letters with parentheses. For example, to send the string ABCdE, you would enter (ABCdE).

Note: The accesscode entered with the ACCESSCODE command is never translated to uppercase letters, even when the MAPINPUT option is TRUE.

MAPOUTPUT

This parameter determines whether all control characters are translated to null values before being displayed on your screen. Translating prevents display distortion on certain types of display terminals. The default setting is TRUE.

FILEXFER

This parameter, when TRUE, suppresses the display of the information bar and column indicators so that only text records are displayed. This parameter is intended to facilitate the use of special file transfer software, that use the CANDE editor functions to download source files to a workstation. The only nontext record that is displayed is #END. The default value for this parameter is FALSE.

FORMSMODE

This parameter causes an alternate editor file to be displayed in forms mode so that no changes can be made to the file. The default value for this parameter is TRUE.

AUTOREV

This parameter causes the automatic reversal of page display direction, once the end or beginning of the source file has been reached. The default value is TRUE.

OFFSET <integer
1-99>

This parameter defines the number of lines prior to the beginning sequence number that are to be included in each screen display when the PAGE command is used. The default value is two lines.

PAGE <integer 0-99>

This option does the following:

WORKSTATION Command

- Sets the number of lines per page. Multiline displays such as the log commands use this value.
- Sets the maximum number of messages to send before automatically invoking the TANK command.

If the number of messages sent to your station since the last DETANK command exceeds this value, the TANK command is automatically invoked and messages go to the message tank and not to your station. To receive the queued messages, issue the DETANK command.

Setting the WORKSTATION PAGE value to zero causes CANDE to never stop sending messages to your station. Messages are not held in the message tank.

The page range is from 0 through 99 lines. The default setting is 24 lines. The default takes effect for each new CANDE session.

WIDTH <integer
80-132>

This option sets the workstation page width. The accepted range is from 80 through 132 characters. The default setting is 80 characters.

DISPLAY <integer
0-12>

This option can be used with display terminals that provide more than one display page. It sets the page number to which all messages to this station will be sent. The default is the current data communication pointer page.

The accepted range is from 0 through 12. Valid values depend upon the type of terminal you are using. The default setting is 0.

SCROLL

This option can be used with display terminals that support a scrolling mode. In scrolling mode, new text appears at the bottom of the screen and scrolls upward as new text is displayed. The most recent information pushes the oldest information off the page.

NO SCROLL

This option selects nonscrolling mode in which new text appears at the top of the screen, overwriting the next most recent information. The default setting is SCROLL.

If you select the NO SCROLL option and a nonzero DISPLAY page value, the cursor is sent to the home position with each line of each message. Thus, each line of each message is written on top of the previous line.

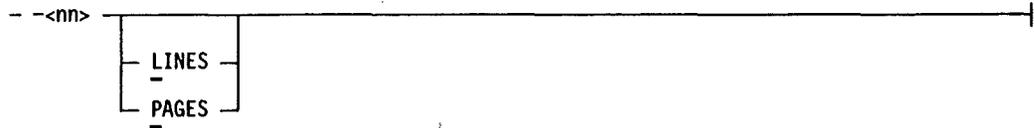
DOWN <integer 0-23>

This option indicates the number of lines to skip down the screen before starting the scroll display. The integer can be a value from 0 to 23. The default value is 2.

- <nn> Command

Use this command to cause the next screen display to start <nn> lines or pages backward in the editor file. You can use an integer in the range 00 to 99 as the value of <nn>.

Syntax



Explanation

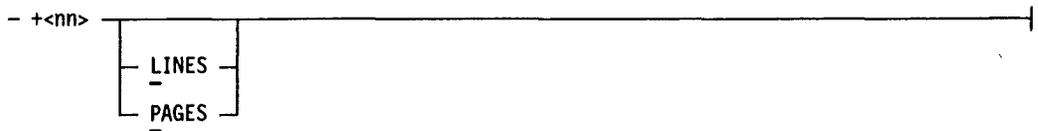
LINES This option sets the unit for <nn> as lines. Lines is the default unit.

PAGES This option sets the unit for <nn> as pages. Lines is the default unit.

+ <nn> Command

Use this command to cause the next screen display to start <nn> lines or pages forward in the editor file. You can use an integer in the range 00 to 99 as the value of <nn>.

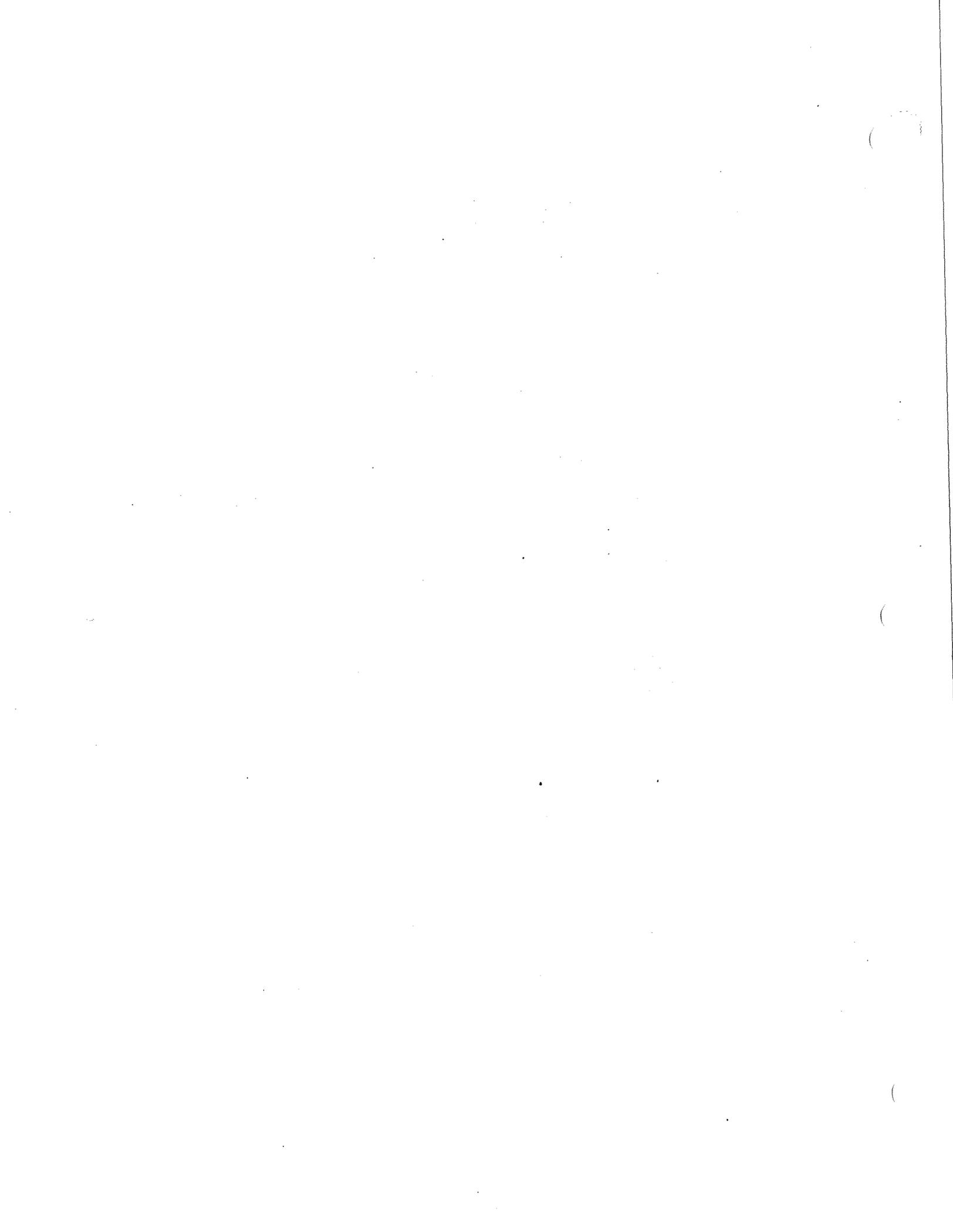
Syntax



Explanation

LINES This option sets the unit for <nn> as lines. Lines is the default unit.

PAGES This option sets the unit for <nn> as pages. Lines is the default unit.



Section 3

Viewing and Printing Files

You can view all editor files on your terminal and print printer backup files by using the View File (VF) utility. This section explains how to initiate this utility and provides the syntax, explanation, and examples for each of its commands. In the syntax diagrams, the minimum portion of a command or parameter that you must enter appears underlined. Where no underlining appears, you must enter the entire command or parameter.

Starting the VF Utility

You can initiate VF from a CANDE editor session, from VCS, or from GEMCOS by using the CANDE *VIEW* command. The *VIEW* command is explained in Section 2.

When you initiate the VF utility from CANDE, the values assigned to the MOC, MODE, OID, PAGE, WIDTH, and OFFSET options of the WORKSTATION command are passed to VF. You can change the page size and page width values with VF commands if you desire. The default family used in CANDE mode is also used in VF mode.

In addition, if you initiate VF from CANDE, screen displays will be in nonforms mode if the file requested for viewing is a printer backup file with a compiler file kind that matches the file kind of the current CANDE editor file. This lets you view a compiler listing and make corrections that will be saved in the editor file. In all other situations, initiating VF from CANDE causes screens to be displayed in forms mode.

If you initiate VF directly from the VCS or GEMCOS, screen displays will be in forms mode, so you cannot edit them.

You can change between forms and nonforms mode by using the VF *FORM* command.

After you enter the *VIEW* command, the system displays either a welcome back message or the last screen display of an open view file.

Note: *Unlike CANDE, the VF utility does not support the input of multiple commands. The entire screen is always considered to be a single command.*

AT Command

Use the AT command to modify the current offset and length for displayed records.

Syntax

- AT - <offset> _____
 └─ :<length> ─┘

Explanation

<offset> This variable represents the offset into the input record in bytes where VF is to begin retrieving data when you use a LIST, PAGE, or OUTPUT command. The default offset is 0. The maximum offset is 20,000 bytes. (The display type is unsigned alpha or hexadecimal.) This option is valid only if a format is not being applied.

:<length> This variable represents the number of bytes that will be retrieved and displayed, starting at the specified offset, when you use a LIST, PAGE, or OUTPUT command. The default value is 80 bytes. The maximum value is the lesser of the record size or the difference between the record size and the offset. The maximum display is 1000 bytes in hex or 1760 bytes in unsigned alphanumeric due to output buffer size limitations. This option is valid only if a format is not being applied.

If the length exceeds the record size or the difference between the record size and the offset, only the remaining portion of the record is printed, to a maximum of 1760 display character in unsigned alpha or 1000 hexadecimal characters.

Example

AT 4:132

This example defines the display offset as 4 bytes and the length as 132 bytes.

BYE Command

Use this command to terminate the VF utility. Any open files are closed.

Syntax

```
[ BYE  
  END ] _____|
```

Explanation

END END is a synonym for BYE. Both commands share identical syntax and function.

Details

When initiated directly from VCS or GEMCOS, the VF utility is automatically terminated if it receives no input for a 30-minute period.

When initiated from a CANDE session, VF is automatically terminated when you terminate CANDE.

CLOSE Command

Use this command to close the current printer backup file. Note that once you open a printer backup file by using the OUTPUT command, each additional OUTPUT command writes data to the same printer backup file until you close it.

Syntax

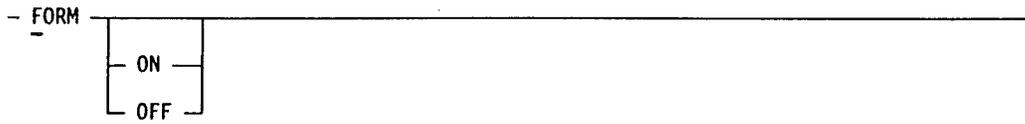
```
- CLOSE _____|
```

FORM Command

Use this command to turn forms mode on and off for screen displays. When forms mode is on, the command line area is the only unprotected data field. When forms mode is off, the entire screen is unprotected.

FORMAT Command

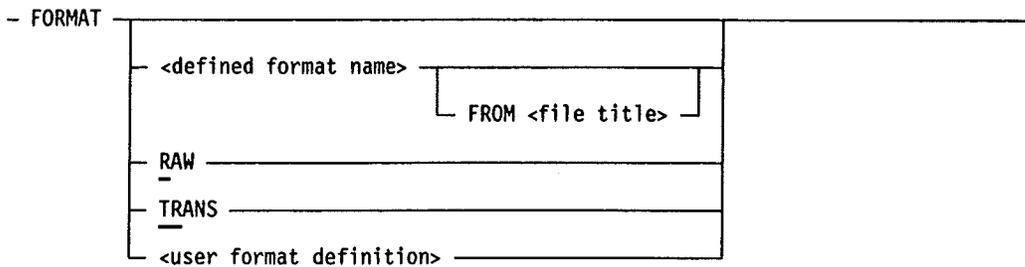
Syntax



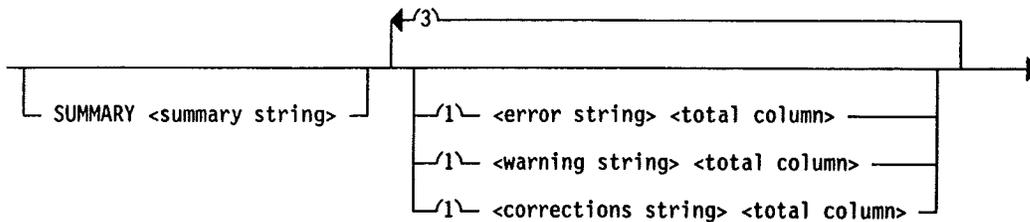
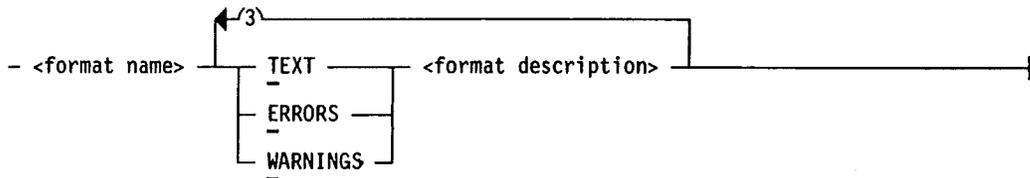
FORMAT Command

Use this command to reformat a listing before displaying it. You can also use the command to create a temporary format definition.

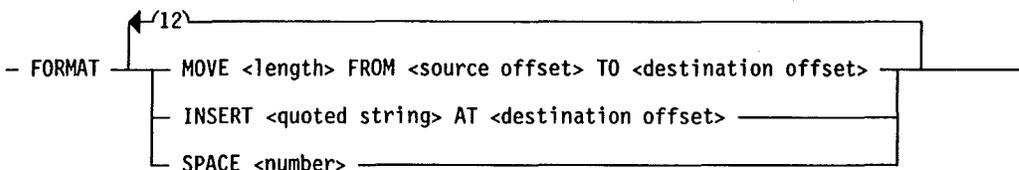
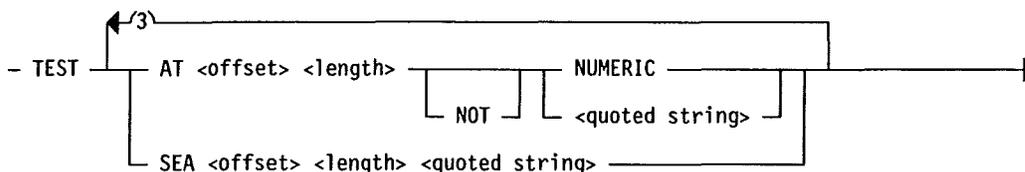
Syntax



<user format definition>



<format description>



Explanation

<defined format name>

This specifies a predefined format that you want to use to reformat the display. The CANFMT file that accompanies the CANDE release contains a variety of predefined formats for your use. You can also specify a format from any other CANDE format file. To display the CANFMT file, enter *WHAT FORMATS*. The maximum length of the format name is six characters.

If you do not need a specific predefined format, you can specify a transparent format by using the TRANS parameter. This parameter translates each embedded data communication control character, that is, any character below EBCDIC 40, to a question mark (?). Thus, the record is displayed in its original context, with only the nongraphic characters being replaced.

FROM <file title>

This parameter specifies the file and optional family in which the defined format is stored.

RAW

This parameter specifies that the display will reflect the original record format of the file with no formatting rules applied.

TRANS

This parameter translates each embedded data communication control character, that is, any character below EBCDIC 40, to a question mark (?). Thus, the record is displayed in its original context, with only the nongraphic characters being replaced.

<user format definition>

This is the name and description of your custom format.

FORMAT Command

<format name>	This is the name of a custom format that you have stored in an editor file. The maximum length of the format name is six alphanumeric characters.
TEXT ERRORS WARNINGS	These keywords identify the format type. You can select only one keyword for the format type; however, a single format definition can be made up of more than one of these format types.
<format description>	This parameter defines the validation checks that are to be performed on each record (TEST parameter) and describes how the input record will be reformatted for display (FORMAT parameter). Both the TEST and the FORMAT parameters are described later in this discussion of the FORMAT command.
SUMMARY <summary string>	This parameter defines the unique string that is searched for at the end of the compile listing to determine the starting record for displaying all summary information. The summary string can contain a maximum of 30 alphanumeric characters.
<error string>	This parameter defines the unique string that is searched for in the summary information to determine the total number of errors. The error string can contain a maximum of 30 alphanumeric characters.
<total column>	This parameter indicates the beginning column where the total number of errors, warnings, or corrections will be displayed.
<warning string>	This option defines the unique string that is searched for in the summary information to determine the total number of warnings. The warning string can contain a maximum of 30 alphanumeric characters.
<corrections string>	This option defines the unique string that is searched for in the summary information to determine the total number of corrections. The corrections string can contain a maximum of 30 alphanumeric characters.
SEQNBR <offset> <length>	This parameter defines the zero-relative position for the beginning sequence number of the file kind for the file being displayed. This value is used when the SEQUENCE option is specified as part of the LIST, PAGE, or OUTPUT command. If you do not specify offset and length values for this parameter, you cannot display the file by sequence number.
FLAGINDREC	This is a boolean indicating that the compiler generated error and/or warning listing includes multiple records.

When a match is found for an error or warning, the previous record is also formatted and displayed.

Format definitions that use the **FLAGINDREC** parameter will not display records that do not match any of the text, error, or warning descriptions. To view these records, specify a record range rather than a sequence range when using the **PAGE** command, for example *PAGE REC 1030*.

TEST This parameter defines the validation checks to be performed to determine if the record is to be displayed. The parameters that follow the word *TEST* indicate the location in the record and the characteristics **CANDE** is to check.

AT This parameter indicates that the test result should be **TRUE**. You can use the **AT** statement up to three times in a single **FORMAT** command. Logical **AND** statements must precede **OR** statements. For example,

```
AT ... AND AT ... AND AT ....
AT ... OR AT ... OR AT ...
AT ... OR AT ...OR SEA ...
SEA ... AND SEA ... AND SEA ...
SEA ... AND SEA ... AND AT ...
SEA ... AND AT ... AND AT ...
SEA ... AND AT ... OR AT ...
```

See the example at the end of this discussion.

<offset> This parameter defines the offset where the test is to begin. You must specify a value.

<length> This parameter defines the length of the record to be tested. The maximum value is 30 characters.

NUMERIC This parameter directs **CANDE** to test for all numeric characters. Including the word *NOT* before **NUMERIC** directs **CANDE** to test for all alphabetic characters.

<quoted string> This parameter directs **CANDE** to search for an exact match of the specified string. While the string can contain a maximum of 30 characters, the length that will be used in the comparison will be the value specified as <length>. Including the word *NOT* before the quoted string directs **CANDE** to search for the absence of the quoted string.

SEA This parameter limits the search to a quoted string, beginning with the offset and continuing through the end of the current display record length. You can use

FORMAT Command

the SEA statement up to three times in a single FORMAT command. Logical AND statements must precede OR statements. For example,

```
AT ... AND AT ... AND AT ....
AT ... OR AT ... OR AT ...
AT ... OR AT ...OR SEA ...
SEA ... AND SEA ... AND SEA ...
SEA ... AND SEA ... AND AT ...
SEA ... AND AT ... AND AT ...
SEA ... AND AT ... OR AT ...
```

See the example at the end of this discussion.

FORMAT

This parameter describes how the input record will be formatted for display.

MOVE <length>
 <source offset>
 <destination offset>

This parameter moves the number of characters specified as length from the source offset to the destination offset. The value for length is 1 to 132 characters. The values for the offsets are 0 to 20000.

INSERT <quoted
 string> AT
 <destination offset>

This parameter inserts the quoted string at the destination offset. The maximum string length is 30 characters. The offset is 0 to 20000. You can use the INSERT parameter a maximum of three times within a format description.

SPACE <number>

This parameter indicates the number of line feeds for a printer backup file. The value of the number can be 1 through 99.

Details

A variety of predefined formats is available in the CANFMT file that accompanies this release. You can select one of these formats by name with the FORMAT command. For details on accessing the CANFMT file, see Section 6.

You can define your own format in the CANFMT file or another CANDE format file. For custom format files, you must use DATA as the file kind and specify a record text length of 476 bytes (the maximum length of a format definition). You can type the format definition in free format, with a maximum of 80 contiguous spaces between format definition statements. See the example that follows.

Examples

```
FORMAT COBOLX FROM CANFM2 ON PRODPK
```

This example uses the defined format named COBOLX, located in the CANFM2 file on the PRODPK family.

```

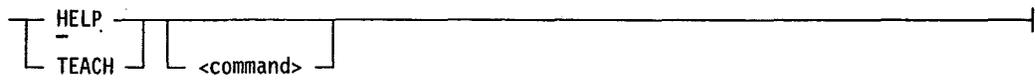
FORMAT COBOLX
  TEXT TEST AT 5 6 NUMERIC AND AT 12 1 " "
  FORMAT MOVE 80 FROM 5 TO 0
  ERRORS TEST AT 4 1 "E"
  FORMAT MOVE 80 5 0 SPACE 2
  WARNINGS TEST AT 4 1 "W" OR AT 4 1 "F"
  FORMAT MOVE 80 FROM 5 0
  SUMMARY "SUMMARY"
    "TOTAL ERRORS" 30
    "TOTAL WARNINGS" 30
    "TOTAL CORRECTIONS" 30
  SEQNBR 5 6
  FLAGINDREC
  
```

This example uses the format named COBOLX and specifies numerous options with the FORMAT command.

HELP Command

Use this command to list the VF commands or to display the syntax diagram and brief explanation of a specific command.

Syntax



Example

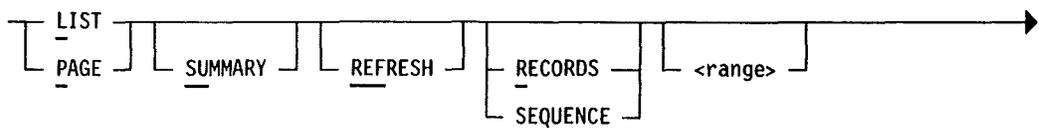
```
H FORMAT
```

This example displays the syntax diagram and explanation of the FORMAT command.

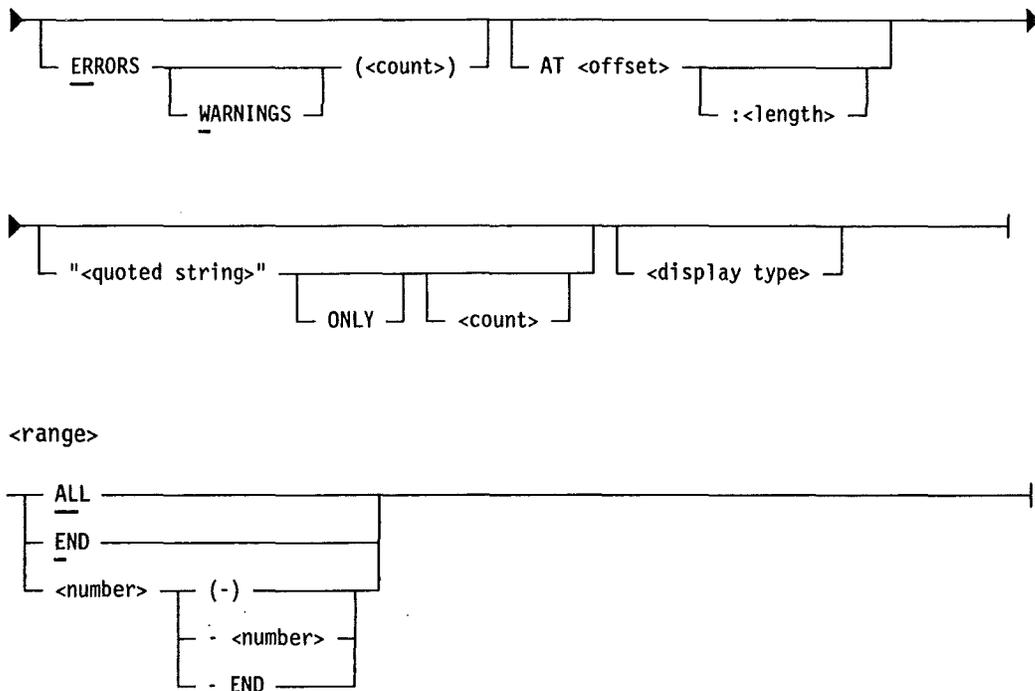
LIST Command

Use this command to display a portion of the current view file on your workstation.

Syntax



LIST Command



Explanation

- PAGE** PAGE is a synonym for LIST. Both commands share identical syntax and function.
- SUMMARY** This option causes the compiler summary to be included in a printer backup listing.
- REFRESH** This option causes the previous display page to be displayed on the workstation.
- RECORDS** This option indicates that the range value refers to records. A record is a relative position (first, second, third, and so on) within the file. RECORDS is the default option for file kinds that do not have their sequence number information defined by the format currently being used.
- SEQUENCE** This option indicates that the range value refers to sequence numbers. Using this option requires that the format description being used for the file has a SEQNBR definition.
- <range>** This option indicates the starting position of the file you are listing. You can specify ALL to select all of the file, END to select the last record or sequence number,

or you can specify a range of records or sequence numbers. ALL is the default value.

Unless otherwise specified, the display begins with the next record following the last screen displayed.

If the format in use has a SEQNBR definition (assigned with the FORMAT command), the range is interpreted as sequence numbers. Otherwise, the range is interpreted as records.

ERRORS

This option displays the compiler syntax errors contained in a printer backup file listing.

WARNINGS

This option displays the nonfatal compiler messages in a printer backup listing.

<count>

This is a 2-digit integer from 01 through 99. When the count value follows the ERRORS and WARNINGS parameters, it indicates the maximum number of errors and warnings that can occur before the command is terminated. When the count value follows the quoted string parameter, it indicates the number of quoted strings that can occur before the command is terminated.

AT <offset>

This variable represents the offset into the input record, in bytes, where VF is to begin retrieving data, and is valid only if a format is not being applied. The default offset is 0. The maximum offset is 20,000 bytes. (The display type is unsigned alpha or hexadecimal.)

:<length>

This variable represents the number of bytes that will be retrieved and displayed, starting at the specified offset, and is valid only if a format is not being applied. The maximum value is the lesser of the record size or the difference between the record size and the offset. The maximum display for each page is 1000 bytes in hex, or 1760 bytes in unsigned alpha, due to output buffer size limitations.

If the length exceeds the record size or the difference between the record size and the offset, the remaining portion of the record is printed or displayed. Display lengths greater than the workstation page size are displayed one page at a time through the end of the record.

If you do not specify a value for the length, the default value is 80 bytes.

LIST Command

<quoted string> This parameter lets you specify a string for which the LIST command will search the file. The maximum length of the string is 30 characters.

The characters in the string are interpreted according to the current display type (unsigned alphanumeric or hexadecimal). You can override the current display type by assigning a display type with the LIST command.

If a printer backup file is being searched, the entire record (132 characters) is searched, regardless of the current length value.

ONLY This parameter directs VF to display only matching records when searching for a quoted string.

<display type> This option indicates whether or not VF needs to translate the file into displayable characters. The values for the display type are UA (unsigned alphanumeric) or HX (hexadecimal). UA is the default value.

If the display type is unsigned alphanumeric, VF performs no translation. VF assumes that the file contains only displayable characters.

If the display type is hexadecimal, VF translates each digit in the file to its corresponding displayable character. VF then displays both the hexadecimal and alphanumeric representations of the record. Nondisplayable characters below EBCDIC 40 are translated to a question mark (?). Nondisplayable characters above EBCDIC 40 are translated to a period (.).

Details

Whether or not you can edit a file that you have listed depends upon how you execute the VF utility. If you initiate VF directly from the MCS, screen displays will be in forms mode, so you cannot edit them.

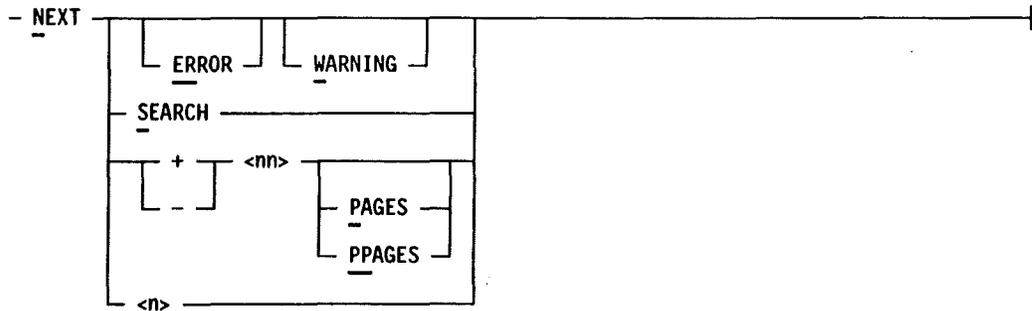
If you initiate VF from CANDE, screen displays will be in nonforms mode if the file requested for viewing is a printer backup file with a compiler file kind that matches the file kind of the active CANDE editor file. This lets you view a compiler listing and make corrections that are immediately entered into the active editor file. In all other situations, initiating VF from CANDE causes screens to be displayed in forms mode.

You can change between forms and nonforms mode by using the FORM command.

NEXT Command

Use this command to display the next page of the file you are viewing. Entering an ETX, a space, or pressing the specify key also displays the next page.

Syntax



Explanation

ERROR

This option causes VF to search for the next error beginning with the record that follows the last error display. You can use this option to resume a search for errors that you interrupted. For example, you could enter the PAGE ERRORS command, which would display the first page of errors. You could use the LIST command to display additional lines of the file. Then you could enter *N ER* to continue displaying pages of errors. This option is valid only if you previously initiated a search for errors by using the PAGE or LIST command with the ERROR option.

WARNING

This option causes VF to search for the next warning beginning with the record that follows the last error display. You can use this option to resume a search for warnings that you interrupted to view other portions of the printer backup listing. This option is valid only if you previously initiated a search for warnings by using the PAGE or LIST command with the WARNING option.

SEARCH

This option specifies that the next display should begin with the quoted search string previously defined in the LIST or PAGE commands.

+

This defines a forward page direction. A space must precede the plus sign (+).

-

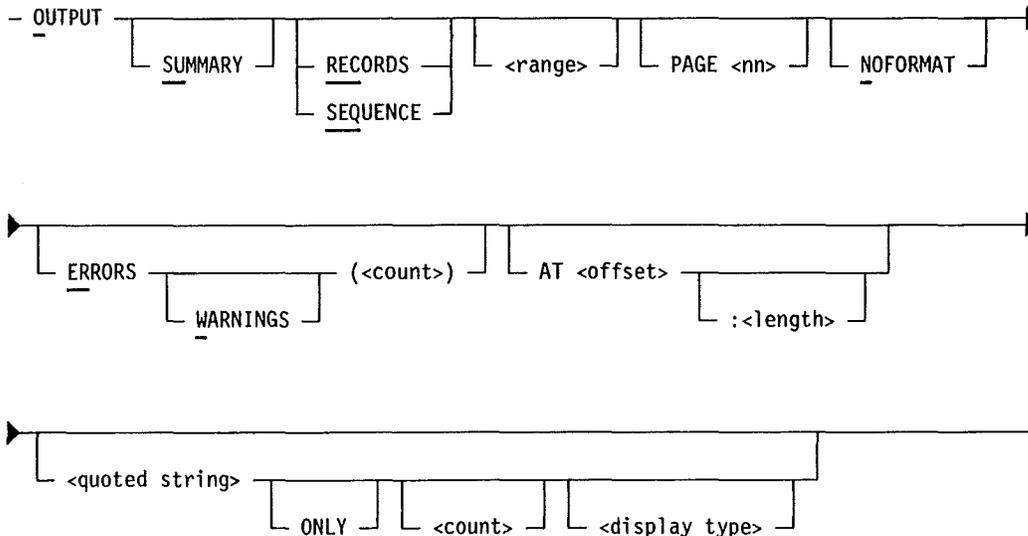
This defines a backward page direction. A space must precede the minus sign (-).

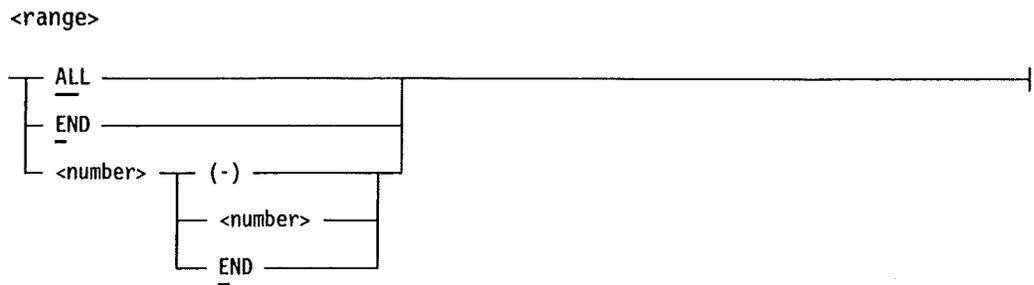
OUTPUT

- <nn>** This is a value from 0 to 99 that indicates the number of pages or records to display forward or backward from the previous page. (The default unit is records. To display pages, include the PAGE option described below.) If you omit the value, the default value of 0 (zero) is used.
- PAGES** This defines the unit for <nn> as workstation pages. The default value is records.
- PPAGES** This option indicates the number of printer pages to skip before beginning the next display. This option is valid only for printer backup files that contain page skips.
- <n>** This parameter specifies an additional number of pages to be displayed consecutively after the display of the next page. The valid range is from 0 to 9 pages. The default value is 0, which means that no additional pages are automatically displayed.

OUTPUT

Use this command to send a portion of a printer backup listing, or the defined or raw format of any other type of file, to a printer backup file.





Explanation

SUMMARY

This option causes the compiler listing to be printed. This option is valid only when viewing printer backup files and requires an associated compiler listing format to be in effect.

RECORDS

This option interprets the unit for the range parameter value as records. The unit for the range can be either records or sequence numbers. When VF is initialized, the range interpretation is in records. Afterward, the setting last used in either the OUTPUT or the LIST command is the setting used when you do not specify either the RECORDS or the SEQUENCE parameter.

Note that the RECORDS parameter indicates a relative position (first, second, and so on) and is the default value for any file kind that does not have sequence number information defined by the format currently being used.

SEQUENCE

This option interprets the unit for the range parameter value as sequence numbers. The unit for the range can be either records or sequence numbers. When VF is initialized, the range interpretation is in records. Afterward, the setting last used in either the OUTPUT or the LIST command is the setting used when you do not specify either the RECORDS or the SEQUENCE parameter.

<range>

This option indicates the portion of the file that is to be selected. You can specify ALL to select all of the file, END to select the last record or sequence number, or a specific range of records or sequence numbers. The default value is ALL.

If the format you are using has a SEQNBR definition assigned with the FORMAT command, the range is interpreted as a sequence number.

OUTPUT

- NOFORMAT** This option indicates that no format is associated with this file.
- PAGE <nn>** This option lets you specify a page size to override the default page size of 56 lines per page. The valid range of values is 00 through 99. Using a value of 00 causes no page breaks to occur in the printer backup listing.
- ERRORS** This option displays the compiler syntax errors contained in a printer backup listing. This option is valid only when viewing printer backup files and requires an associated compiler listing format to be in effect.
- WARNINGS** This option displays the nonfatal compiler messages contained in a printer backup listing. This option is valid only when viewing printer backup files and requires an associated compiler listing format to be in effect.
- <count>** This is a 2-digit integer from 01 to 99. When it follows the **ERRORS** and **WARNINGS** options, it specifies the maximum number of errors and warnings that can occur before the command is terminated. When the count follows the quoted string parameter, it indicates the number of quoted string matches that will occur before the command is terminated.
- AT <offset>** This variable represents the offset into the input record in bytes where VF is to begin retrieving data. This option is valid only if a format is not being applied. The maximum offset is 20,000 bytes. If you do not specify an offset, the default offset is either the offset last specified in an **OUTPUT**, **PAGE**, or **LIST** command or the initialized value of 0.
- :<length>** This variable represents the number of bytes that will be retrieved and displayed, starting at the specified offset. This option is valid only if a format is not being applied. The maximum value is the lesser of 20,000 or the difference between the record size and the offset. If you do not specify a length, the value for length is either the value specified in the last **OUTPUT**, **PAGE**, or **LIST** command or the initialized value of 80.
- If the length exceeds the record size or the difference between the record size and the offset, only the remaining portion of the record is printed, to a maximum of 20,000 display characters.

<quoted string> This parameter lets you specify a string for which the LIST command will search in the file. The maximum length of the string is 30 characters.

The characters in the string are interpreted according to the current display type (unsigned alphanumeric or hexadecimal). You can override the current display type by assigning UA (for unsigned alphanumeric) or HX (hexadecimal) to the DISPLAY TYPE option of the OUTPUT command.

ONLY This parameter directs VF to display only matching records when searching for a quoted string.

<display type> This option indicates whether or not VF needs to translate the file into displayable characters. The values for the display type are UA (unsigned alphanumeric) or HX (hexadecimal). UA is the default value.

If the display type is unsigned alphanumeric, VF performs no translation. VF assumes that the file contains only displayable characters.

If the display type is hexadecimal, VF translates each digit in the file to its corresponding displayable character. VF then displays both the hexadecimal and alphanumeric representations of the record. Nondisplayable characters below EBCDIC 40 are translated to a question mark (?). Nondisplayable characters above EBCDIC 40 are translated to a period (.).

Details

After loading the print record to the buffer, CANDE determines if columns 129 through 132 are blank. If so, CANDE places the file-relative record number in these columns.

REDO Command

Use this command to display the last valid command you entered. This lets you easily resubmit or edit the command.

Syntax

- REDO _____

REMOVE Command

REMOVE Command

Use this command to close and purge the file you are currently viewing. If the file is not a printer backup file, this command closes and saves the file.

- REMOVE _____|

SAVE Command

Use this command to close and save the file you are currently viewing.

- SAVE _____|

SESSION Command

Use this command to display various statistics about your VF session, including information about the session user and the current view file. See the following example for a sample screen.

Syntax

- SESSION _____|

Details

The following is an example of the screen response from the SESSION command:

SESSION

```

Session Usercode = TOPGUNN
Session Charge # = 62147
Username         = PROGRAMMER
Installation #   = 0001
Hostname        = DEVHOST
Autologoff      = 30
FORM            = OFF
Format Name     = BPL , from: CANFMT on DISK
Options:        Offset = 1, Length = 80, Display = SEQ UA
Workstation:    Page  = 14 Width  = 80
Active ViewFile: @01680 on BACKUP Owner: TOPGUNN
                  Current range, REC 2-1350 SEQ 1-99999999
                  Rec size (digits) = 288, Recs per block = 10
                  Current rec = 1, Total recs = 1388

```

VIEW Command

Use this command to close the current file, if one is open, and read the specified file. Before VF closes the current file, it displays a prompt that lets you save or remove the file being closed.

Syntax

```

- VIEW [ <file title> ] [ <printer backup file number> ]
        [ <format name> ] [ FROM <file title> ]

```

WHAT Command

Explanation

<file title>	This is the name and optional family of the file you want to view.
<printer backup file number>	This is the number of the printer backup file you want to view. You must include all five digits of this number if required for processor number identification.
<format name>	This option lets you specify a format that overrides the compiler file kind in the printer backup file label.
FROM <file title>	This option lets you specify a custom format file from which the specified format is derived. If you do not include a file title, the session format file name specified in the user variable file is used. The default format name is CANFMT, the Unisys standard format file. You can change this name by including the CANFMT statement in the user variable file. For details about the CANFMT file, see Section 6.

Details

After you enter the VIEW command, CANDE searches for the file on the following media, in order:

1. Primary system backup family
2. Secondary system backup family
3. DISK
4. Session default family

WHAT Command

Use this command to perform the following functions:

- List all format names found in the CANFMT file currently in use.
- Display the file header information of a specific file.
- Display the sector or segment that contains the first record of the most recent display page.

?BRK Command

Explanation

- PAGE (=) <1-25>** This option determines the maximum number of lines that will be displayed in a single response from a command.
- WIDTH (=) <80-132>** This option determines the maximum number of columns that are displayed per line. If this value exceeds the actual workstation capability, the line is continued on the next line. The default value is 80 columns.
- OFFSET (=) <0-99>** This option determines how many records preceding the actual beginning record will be displayed when you enter a LIST or PAGE command containing a search for a quoted string. This value is applied only when you are using a predefined or custom format. The default value is 11 records.
- PREVIOUS (=) <0-99>** This option determines how many records preceding a warning or error message will be displayed when you are using a compiler format and searching a printer backup listing. The default value is 2.
- TRANSLATE (=) TRUE
(or) FALSE** This option determines whether or not all input commands are translated to uppercase letters. The default setting is TRUE.

?BRK Command

Use this command to terminate the command currently in progress. You can also use this command to abort lengthy string searches or generation of printer output.

Syntax

- ?BRK _____

- <nn> Command

Use this command to cause the next screen display to start <nn> lines or pages backward in the file you are viewing. You can use an integer in the range 00 to 99 as the value of <nn>.

Syntax

```
-- <nn> [ PAGES | PPPAGES ]
```

Explanation

PAGES This option interprets the unit for <nn> as workstation pages. Lines is the default unit.

PPPAGES This option interprets the unit for <nn> as printer page skips and is valid only for printer backup files that contain page skip information.

+ <nn> Command

Use this command to cause the next screen display to start <nn> lines or pages forward in the file you are viewing. You can use an integer in the range 0 to 99 as the value of <nn>.

Syntax

```
- + <nn> [ PAGES | PPPAGES ]
```

Explanation

PAGES This option interprets the unit for <nn> as workstation pages. Lines is the default unit.

PPPAGES This option interprets the unit for <nn> as printer page skips and is valid only for printer backup files that contain page skip information.



Section 4

Creating Batch Files

A CANDE batch file is a file of commands that can be executed by a single CANDE *DO* command or by the system when CANDE is initiated. A CANDE batch file can be either of the following:

- A CANDE editor file (500 bytes per record)
- A system file (80 bytes per record)

This section describes the structure and contents of a batch file, provides guidelines for creating a batch file, and explains how to execute a batch file.

Batch File Contents

The first record of a batch file must contain either the LABELBEGIN keyword or the //BEGINJOB statement. The last record must contain either the LABELEND keyword or the //ENDJOB statement.

You can include the following input in a batch file:

- CANDE commands (except DO)
- Special statements that control the processing of the batch file (described in this section)
- Labels to mark portions of the batch file to be processed
- V Series system commands
- Variables whose value you can specify when you enter the DO command, allowing you to create generic batch files that can be used in many situations
- Multifile identifiers
- References to printers
- Comments
- Pseudo card reader (PCR) commands
- Miscellaneous text, including VCS transactions and GEMCOS commands and input for applications

Instructions for including these various elements in a batch file are provided in the following text.

Special CANDE Batch File Statements

The following are special batch file statements. These statements must always be preceded by two slashes (//).

//ACCEPT Statement

This statement suspends processing of the batch file and waits for your input. You usually precede the //ACCEPT statement with an informative //DISPLAY statement.

Together, the //DISPLAY and //ACCEPT statements allow user interaction for direction of flow within the batch file. The //IF RESPONSE batch file statement can be used to interrogate and branch, based on the responses. For more information on the //DISPLAY and //IF RESPONSE statements, read further in this section.

The maximum //ACCEPT response length is 60 characters.

Example

```
//DISPLAY "ANSWER Y OR N"  
//ACCEPT  
//IF RESPONSE = "Y" THEN  
//BEGIN...  
//END ELSE
```

//AT Statement

This statement suspends execution of the CANDE batch file until all previously initiated tasks have run to completion, whether or not the tasks ran successfully. The tasks must have been initiated by the CANDE batch file.

When the tasks complete, CANDE resets any error flags that may have been set by an unsuccessful completion and executes the next statement in the batch file.

This statement contrasts with the //IF statement, which tests whether or not a task completed successfully.

Text following the //AT statement acts as a comment only. Syntax and spelling checks are not performed for these words; you may use any words.

Example

```
//AT TASKCOMPLETION
```

//BEGIN Statement

The //BEGIN and //END statements form logical groups of statements in a CANDE batch file. There must be an END statement for each BEGIN statement. BEGIN statement and END statement pairs can be nested, as shown in the following example:

Example

```

BEGIN
  .
  BEGIN
    .
    END
  .
  END

```

//BEGINJOB Statement

This marks the beginning of a CANDE batch file, or the beginning of a job within a batch file. The //BEGINJOB statement must be either the first statement, or else the first statement after a LABELBEGIN keyword in the batch file.

Examples

```

//BEGINJOB
COMPILE CMP1 ON OMEGA

LABELBEGIN CANDEXX
//BEGINJOB
RUN CANDEX ON DATCOM
//LABELEND CANDEX
//ENDJOB

```

//DISPLAY Statement

In batch mode, this statement displays a message only in the ODT log. The message text to be displayed must be enclosed in quotation marks (").

If CANDE is executing under VCS or GEMCOS, this statement displays a message text only to your workstation. The message text to be displayed must be enclosed in quotation marks (").

With VCS or GEMCOS, any or all of the following message emphasis types can be added to the display:

B	Blink
H	Highlight

Batch File Contents

R	Reverse video
U	Underline

Example

```
//DISPLAY BHUR "WARNING! ERROR 01"
```

Note: If the characters @ or ! are used in a display message, each occurrence must be followed by a space to avoid confusion with the primary CANDE meaning of the characters.

//END Statement

//BEGIN and //END form logical groups of statements in a CANDE batch file. There must be an //END for each //BEGIN.

//ELSE Statement

The //ELSE statement marks the branch of the //IF statement that is executed if the tested condition is false. Only a single statement can follow an //ELSE statement. To include more than one statement after //ELSE, use a //BEGIN and //END pair. If the //ELSE statement follows an //END statement, do not put slashes in front of the word ELSE; this allows you to put ELSE on the same line as //END.

//ENDJOB Statement

The last entry in a batch file must be an //ENDJOB statement. This statement terminates reading of the batch file. A single //ENDJOB statement terminates all previous //BEGINJOB statements; //BEGINJOB and //ENDJOB pairs cannot be nested.

//IF Statement

Syntax

```
//IF [NOT]
    [COMPILEDOK]
    [COMPLETEDOK]
    [FILEPRESENT<filetitle>]
    [RESPONSE = "<desired response match-max 60 chars>"]
```

The //IF statement tests whether a compile or an execution has finished successfully. The compile or execution must be a task that was started by the CANDE batch file. If the task compiled or executed successfully, the specified THEN clause is executed. If the task compiled or executed unsuccessfully, the branch at //ELSE or //END ELSE is executed.

You can test for these conditions: successful completion, unsuccessful completion, and file presence or absence.

CANDE is informed by the operating system of the successful or unsuccessful completion of task. Unsuccessful completion includes tasks terminated by a DS, RS, or QT system statement. The test works the same way whether you add COMPILEDOK or COMPLETEDOK to the //IF statement.

The NOT option applies inverse logic statements with COMPILEDOK, COMPLETEDOK or FILEPRESENT.

Once an //IF statement has been encountered, no further statements in the batch file can be executed until the //IF condition is satisfied. All tasks initiated before the //IF statement must be completed before the //IF statement can be evaluated. This includes tasks initiated while CANDE is waiting for the completion of previously initiated tasks.

You can nest //IF statements, as shown in the following example:

```

RUN PROG1
  //IF COMPLETEDOK THEN
  //IF FILEPRESENT FILEA THEN
  RUN PROG2
  //ELSE
  //DISPLAY "FILEA NOT PRESENT"
  //ELSE DISPLAY "PROG1 FAILED"

```

The //IF statement contrasts with the //AT statement. The branch of the //IF statement that is executed depends on the outcome of the test condition, whereas the single branch following an //AT statement is executed when the named task finishes, whether or not the task was successful.

The //IF FILEPRESENT statement can be used to check for the absence or presence of a file, for example:

```
//IF FILEPRESENT <filename> [ON <family>] THEN
```

This statement asks whether a file is on disk or disk pack and allows logical branching based on the result.

The set of statements in the following example provides interactive user dialog to direct the logic within a batch file. If CANDE is executing in batch mode, the interface is through the //DISPLAY and //ACCEPT statements. If executing with an MCS, the interface is through write and read operations to your workstation. In both interfaces, the response is limited to 60 characters.

```
//DISPLAY "<user interrogation statement>"(yes or no, for example)
//ACCEPT
//IF RESPONSE = "Y" THEN
```

You can use the //BEGIN and //END statements after an //IF statement to mark a block of code to be executed if the test is successful, as shown in the following

Batch File Contents

example. If the test is not successful, the marked code is not executed, and control “falls through” to the next statement. If you use an //ELSE statement after the //END statement, do **not** precede the ELSE with two slashes; this allows you to put ELSE on the same line as //END.

```
//IF FILEPRESENT FILEA THEN
//BEGIN
?COPY FILEA AS FILEB
RUN PROG1
//END
```

Note that a question mark (?) must precede the COPY command because the COPY system command is intended.

If you want to branch to a different block of code if the test fails, use an //ELSE statement as shown in the following example:

```
//IF FILEPRESENT FILEA THEN
?COPY FILEA AS FILEB
//ELSE
//DISPLAY "FILEA MISSING"
```

Using Labels to Execute Portions of a Batch File

You can execute individual jobs within a batch file by labeling the jobs. Each labeled job in the batch file begins with the LABELBEGIN keyword followed by a label. The label can contain up to 30 alphanumeric characters, but cannot contain a slash (/), a colon (:), or a hyphen (-). Following the label is the //BEGINJOB statement. Following the //BEGINJOB statement is the job itself.

To mark the end of the labeled job, use the //ENDJOB statement followed by the LABELEND keyword followed by the label you used previously. You must use the same label to mark the beginning and end of the job.

The following example shows how to label portions of text in a batch file. To execute the labeled job, use the @ <label> syntax of the DO command, described later in this section.

```
LABELBEGIN COMPILEJOB
//BEGINJOB
COMPILE PROG1
//ENDJOB
LABELEND COMPILEJOB
```

Using Variables in Batch Files

You can create a “generic” batch file that can be used in different ways by using variables with commands. You can use a variable in place of a regular syntax element of a command, for example in place of a file name. You define a variable

name in the batch file and then supply a value for the variable when you enter the DO command.

Variable names can be up to 17 characters long and must be enclosed in angle brackets (<>).

Examples

The following is an example of a batch file that prints a file. The variable is PRNAME, which specifies the name of the file to be printed.

```
//BEGINJOB  
PRINT <PRNAME>  
//ENDJOB
```

If the batch file in the preceding example is titled PRNT, the DO command syntax to execute the batch file and assign a value to the variable would be similar to the following:

```
DO PRNT <PRNAME> FIL2
```

The batch file is executed and FIL2 is substituted for the <PRNAME> variable. The batch file issues the command PRINT FIL2.

The following is an example of a batch file that copies a file by using the CANDE COPY command:

```
//BEGINJOB  
COPY <FILENAME1> AS <FILENAME2> ON <FAMNAME>  
//ENDJOB
```

If the batch file in the preceding example is titled CPY, the DO command syntax to execute the batch file and assign values to the variables would be similar to the following:

```
DO *CPY <FILENAME1> MYFL <FILENAME2> URFL <FAMNAME> USERS
```

Including V Series System Commands in a Batch File

You can include V Series MCP control instructions in a batch file. CANDE submits these instructions to the operating system under control of the logic in the batch file.

The *V Series Systems Operations Guide, Volume 2: System Commands* describes the V Series system commands, including the MCP control instructions and indicates whether each system command is an MCP control instruction or a keyboard input message. Only MCP control instructions are allowed in a batch file.

Batch File Contents

MCP control instructions that exceed 80 characters in length in a batch file must begin and end with a period (.). This is illustrated in the following example, which is a system command with 85 characters:

```
?COPY FILEA AS FILEA1, FILEB AS FILEB1, FILEC AS FILEC1 FROM ABC (PACK) TO DEF (PACK).
```

To enter multiline system commands in a batch file, begin and end the entire command string with periods. You may enter up to 24 lines of MCP control instructions in one string. The following example shows a three-line string of system commands:

```
.?COPY FILEA AS FILEA1, FILEB AS FILEB1, FILEC AS FILEC1  
FROM ABC(PACK) TO DEF (PACK)  
AND FILED AS FILED1 FROM GHI (PACK) TO JKL (PACK).
```

Using Multifile Identifiers in a Batch File

You can use a multifile identifier to specify the disk pack family location of a file. If you do so in a batch file, separate the multifile identifier from the file name with a slash.

If you are uncertain whether you will use a multifile identifier in your batch file, or if the multifile identifier could change with programmatic conditions, you can use a variable in place of the multifile identifier. In this case, do **not** include the slash in the batch file; instead, include the slash when you enter the multifile identifier in the variable value in the DO command. Because the slash is a special character, you must enclose the multifile identifier in quotation marks. The following examples show the use of file identifiers in a batch file.

Examples

```
//BEGINJOB  
COMPILE < INMF > < INFILE > WITH COBOL  
//ENDJOB
```

This example establishes a variable called `INFILE` for the file name of a COBOL source file, and a variable called `INMF` for the multifile identifier for the file. Assume this batch file has the file name `ABCD`.

```
DO ABCD < INMF > < INFILE > FILE1
```

This example initiates processing of the batch file called `ABCD`. It provides the value `FILE1` for the variable `INFILE`. The variable name `INMF` is replaced with spaces.

```
DO ABCD < INMF > "PACK1/" < INFILE > FILE1
```

This example also initiates processing of the batch file called `ABCD`. It provides the value `FILE1` for the variable `INFILE`. It also provides the value `PACK1/` for the variable `INMF`. The slash (/) is required because `INMF` represents a multifile identifier. Quotation marks are required because the slash is a special character.

Referencing Printer Files in a Batch File

To refer to a printer file generated by a task you are monitoring, prefix the printer file number or printer file name with an at sign (@).

Because printer backup numbers are not known before the files are created, use the internal name that the task gives the printer file. This name is included in the printer backup file message that is displayed when the task first opens the printer file.

The printer file can be any printer file; it is not restricted to compile listings.

It is possible for two tasks initiated by the batch file to use the same printer file name. If this happens, subsequent statements in the batch file that refer to the printer file by name will be taken as referring to the printer file that was opened first. For example, the task opens a printer backup file, and displays the following message:

```
@00032 LINE OPEN OUT PROG1/COBOL=018
```

If you want CANDE to print out the printer backup file whose internal file name is LINE, use the following command in your CANDE batch file:

```
?PRINT @LINE
```

If you want to copy the printer backup file to another disk pack, use the following command in your CANDE batch file:

```
?COPY @LINE AS CMLST FROM PACK TO LSTING(PACK)
```

Including Comments in a Batch File

You can add comments to a batch file to document the sequence of actions performed by the DO statements. Accepted comment characters are:

- & (ampersand)
- % (percent sign)
- * (asterisk)

These characters all have the same effect. Any text after the comment character on a line is ignored during compilation or execution as shown in the following example:

Batch File Contents

Examples

```
//BEGINJOB
//IF NOT FILEPRESENT CMPOBJ THEN
//BEGIN
COMPILE CMP1 ON OMEGA      & initiate compile. Could also be
                          & an EXECUTE or a COPY statement.
//IF COMPILEDOK THEN      & test if completed without syntax error
//BEGIN
?COPY @PRNTFI AS PRNTFI FROM PACK TO LSTING(PACK) & save listing
?COPY CMPOBJ TO OMEGA(PACK).                & save object code
//END
//IF COMPLETEDOK THEN
//BEGIN
//DISPLAY "CMPOBJ 2.0 GENERATION COMPLETE"
//END ELSE
//DISPLAY "CMPOBJ 2.0 GENERATION FAILED"
//END ELSE
//DISPLAY "CMPOBJ FILE PRESENT"
//ENDJOB
```

In the preceding example, the CMP1 file referred to in the COMPILE command would be a source file that includes compiler control records, as shown in the following example:

```
?CMP CMPOBJ WITH 'COBOL LIB; AFTER TIME/P1100. (P1100 = 11:00 p.m.)
?FILE LINE = PRNTFI
?DATA CARD
```

Including Pseudo Card Reader Commands in a Batch File

Pseudo card reader (PCR) commands in a batch file can be multiline. The double delimiter character (##) specifies that there is multiline input. The format is as follows:

```
PCR ##<text>
#<text> #<text>
#<text> ... up to 24 lines
PCREND
```

You can specify up to 24 lines of text. PCREND must be on a separate line and can be the only text on that line. Refer to the PCR command in Section 2 for further information.

Embedding Miscellaneous Text in a Batch File

Batch file text other than CANDE commands and MCP control instructions must be preceded by a special character sequence. If the input requires routing by a message control system (MCS), precede the input with the following special character sequence:

/*

When CANDE encounters a batch file string that is preceded by this character sequence, CANDE replaces the character sequence with the transaction initiator character when the string is executed by VCS, or an environmental control command (ECC) flag character when it is executed by GEMCOS. Then the command string is sent to your workstation, followed by an escape sequence to transmit the command to the MCS. Examples of this process are as follows:

```
/*STN-CTL,123,DOWN      (VCS transaction)
/*CHECK                 (GEMCOS command)
/*QUERY
/*CANDE
```

If the input must be routed directly to an application and will not require routing by an MCS, precede the input with the following special character sequence:

```
/**
In this case, CANDE deletes the characters and sends the text as is. For
example:

/**HELLO BIGBAD/WOLF
/**123456 THIS IS AN EDITOR TEXT INPUT LINE
```

To compensate for data communication processing delays between commands, you can use the `//WAIT<wait value>` batch file command. The wait value can be from 1 through 99999 seconds. An example of this is shown as follows.

```
/*CANDE
//WAIT 3
/** GET ABCS NORECALL
//WAIT 3
```

Executing a Batch File

You can execute a batch file by using one of the following methods:

- Use the DO command
- Give the batch file a special CANDE file name that the system recognizes and executes automatically when CANDE is initiated

These methods are described in detail in the following text.

Executing a Batch File with the DO Command

You can use the CANDE `DO` command to execute a CANDE batch file. If the file you name is a source code file rather than a batch file, the DO command initiates a compile. In this case, the DO command is synonymous with the START

Executing a Batch File

command. Keep in mind that a valid batch file must begin with either the LABELBEGIN keyword or the //BEGINJOB statement; otherwise CANDE will not interpret the file as a batch file.

Once you enter a DO command string, CANDE remembers the contents of the last DO command. To repeat the same sequence of commands, enter only the word DO. This is an efficient way of initiating several compiles in the course of debugging. Up to 160 characters of a DO command are retained for reuse in this way.

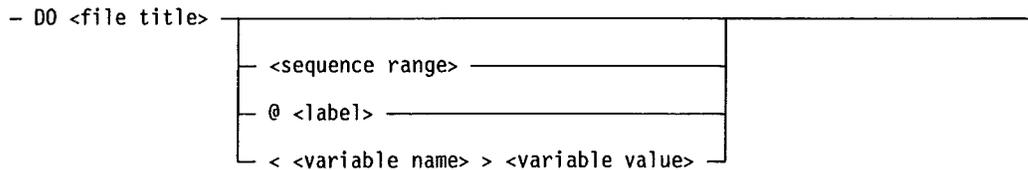
Be aware that while the DO, COMPILE and START commands are interchangeable when used with a file title, COMPILE or START used alone will attempt to initiate only the last job file or source file that was processed. The DO command used alone will initiate only the last batch file that was processed.

If the workstation page becomes full (causing the TANK command to be automatically invoked), processing of the batch file is suspended until a DETANK command is received from the station. Setting the workstation page size to zero will avoid this condition. Use the WORKSTATION PAGE command to set the workstation page size. (For information on setting a workstation page, see "WORKSTATION Command," in Section 3).

Tasks started before or after processing of the batch file is initiated have no effect on the batch file logical processing.

The syntax of the DO command is shown in the following diagram:

Syntax



Explanation

- | | |
|------------------|---|
| <file title> | The file title indicates the name and family location of the batch file. For the syntax of the file title, see the beginning of Section 2. |
| <sequence range> | This is the range of editor lines you want this command to affect. For the syntax of the sequence range, see the beginning of Section 2. |
| @ <label> | A label identifies a particular part of the batch file that you want to be executed. The label can contain up to 30 alphanumeric characters. Do not use a slash (/), a colon (:), or a hyphen (-) in a label. |

The following labels can be used anywhere within a batch file:

!PRIBKP	Replaced with the current MCP primary backup family.
!SECBKP	Replaced with the current MCP secondary backup family.
!DEFFAM	Replaced with the current CANDE session default family name.

**<variable name>
variable value**

This parameter lets you specify a value for a variable in a batch file. The variable name is the variable name as specified in the batch file. The variable value is the value you want to assign to the variable in the batch file.

For example, if the variable name is **PRNAME** and specifies a printer name, the variable value would be the name of the printer you want the batch file to use. The **DO** command might look like the following:

```
DO MYFILE <PRNAME> P105
```

You must enclose the variable name in angle brackets to differentiate the variable name from the variable value.

<variable value>

This parameter lets you assign a value to a variable that you have specified in the batch file. **CANDE** inserts this value at the location of the variable name when the batch file is executed. Thus, you can use the same batch file in different ways by supplying different values for the variables.

If you use a variable name in the **DO** command but do not supply a value, blanks are inserted in the batch file at the location of the variable when the batch file is executed.

If there is a blank or a special character in the variable value, you must enclose the value in quotation marks when you enter the **DO** command.

Automatic Execution When **CANDE** Is Initiated

When you use the name **UVF** or **SVF** in a batch file title, the system automatically executes the batch file when you initiate **CANDE**. **UVF** is the user variable file, and **SVF** is the system variable file.

Batch File Sample

You can create a system variable or user variable file by using the MAKE command. After creating the file, you can edit it to include any CANDE commands, system commands, MCS commands, and VCS transactions for which you are authorized. You could use the batch file to invoke one or more applications; to initialize other MCS session attributes, such as dialogs; to establish CANDE workstation and session parameters; and to execute a number of system commands.

UVF and SVF files can be either CANDE editor files or 80-byte, blocked 1-9 system files. System files must be data files (no sequence numbers).

The following are the possible file titles for user variable files and system variable files in the order in which the system searches for them. Only the first occurrence of a matching file title is used. In the following file titles, <s> represents the system number and <id> represents your 2-character user ID.

1. UVF<s><id> ON DISK
2. UVF:<id> ON <current default family>
3. UVF:<id> ON DISK
4. SVF<s> ON DISK
5. SVF: ON DISK

Batch File Sample

The following example shows several batch file jobs in the same file. Each job starts with BEGINJOB and ends with ENDJOB. The arrows in the left margin show the boundaries of the jobs in the batch file. They also show the boundaries of related blocks of code within jobs. Comments are included to describe the logical flow of the program.

```

┌── LABELBEGIN TST-COMPILE-DECK
├── //BEGINJOB
│   ┌── START CANDEXX ON DATCOM & initiate compile.
│   │   ┌── //IF COMPILEDOK THEN & test if completed without syntax error
│   │   │   ┌── //BEGIN
│   │   │   │   ┌── ?COPY @LXX AS CMLST FROM PACK TO LSTING(PACK)
│   │   │   │   │   ┌── ?PATCH PRGTST 0 0 0 5 UA ABCD
│   │   │   │   │   │   ┌── ?COPY PRGTST TO OBJECT(PACK)
│   │   │   │   │   │   │   ┌── .?EX PRGTST ; FILE INPUT <IN-MFID><IN-FID> %two line
│   │   │   │   │   │   │   │   ┌── FILE OUTPUT <OUT>. % execution string
│   │   │   │   │   │   │   │   │   ┌── //DISPLAY "PRGTST COMPILE SUCCESSFULLY COMPLETED"
│   │   │   │   │   │   │   │   │   │   ┌── //DISPLAY "PRGTST EXECUTION INITIATED"
│   │   │   │   │   │   │   │   │   │   └── //END
│   │   │   │   │   │   │   └── ELSE
│   │   │   │   │   │   └── //DISPLAY "PRGTST COMPILE FAILED"
│   │   │   │   └── ?PRINT @LXX
└──
```

continued

```

//IF FILEPRESENT PRGOUT ON !PRIBKP THEN
//DISPLAY "PRGTST OUTPUT FILE PRESENT"
//ENDJOB
LABLEND TST-COMPILE-DECK

LABELBEGIN PROG1-DECK
//BEGINJOB
?COPY PROG1 AS PROG2 FROM !DEFFAM(PACK)
//IF COMPLETEDOK THEN           & test if completed without error
//BEGIN
?EX PROG2 (1,"PARM1").           &initiate execute of PROG2
//DISPLAY "PROG2 EXECUTION INITIATED"
//AT COMPLETION THEN
LABELBEGIN CMP2-DECK
//BEGINJOB   & must be beginjob to follow labelbegin
FAMILY OMEGA
START CMP2
//DISPLAY "START OF CMP2 COMPILATION"
//IF COMPILEDOK THEN
?MOVE @CMP2L AS CMP2L FROM !PRIBKP(PACK) TO ARCHIV(PACK)
//ELSE
?PRINT @CMP2L SAVE
//END
ELSE
//DISPLAY "PROG2 COPY FAILED"
//ENDJOB   & this pairs with both beginjobs
LABLEND CMP2-DECK
LABLEND PROG1-DECK

```

Assume that the batch file for the preceding example is a usercoded CANDE editor file named TSTDaa on the disk pack family JOBPAK. The lowercase letters aa represent a usercode. The CANDE session is executing under the user name BILL.

The following example shows the DO command syntax to initiate the batch file called TST-COMPILE-DECK with values specified for the variables IN-MFID (the multifile identifier for the input file), IN-FID (the file name for the input file) and OUT (the file name for the output file):

```
DO 8181:TSTD ON JOBPAK @TST-COMPILE-DECK <IN-MFID> "TSTPAK/"
<IN-FID> MYFILE <OUT> NEWFIL DSK
```

If you want to use the default family for the input file, do not include a value for the variable IN-MFID. In this case, the DO command would look like this:

```
DO 8181:TSTD ON JOBPAK @TST-COMPILE-DECK < IN-FID > MYFILE < OUT > NEWFIL DSK
```

If the batch file is a usercoded file with a usercode other than the one you are logged on with, you can override the session usercode. The following example processes a batch file with the usercode associated with the user named JOHN:

```
DO (JOHN)TSTD ON JOBPAK @TST-COMPILE-DECK < IN-FID > MYFILE < OUT > NEWFIL DSK
```

()

()

()

Section 5

Using the CANUTL Utility

CANUTL is a utility that performs a variety of file conversion, comparison, copy, print, rebuild, and reorganization operations. You can execute CANUTL from CANDE or from VCS or GEMCOS. This section describes how to execute and use this utility.

Executing CANUTL from CANDE

To execute CANUTL from a CANDE session, enter

```
CANUTL
```

The screen illustrated in Figure 5-1 is displayed.

```

                                CANUTL MENU
Input file:                               on
Output file:                              on
  Precede non usercoded file name(s) with * (Default output = input filename)

Options - Select only one numeric choice by placing non blank char in field(s)
1. Reorganize/compress CANDE editor file
   Retain current records per area (RPA)
2. Rebuild corrupted CANDE editor file as a CANDE editor file
3. Rebuild corrupted CANDE editor file as an 80 byte system file
   Option 2 or 3: Include deleted records
4. Create a CANDE editor file from an 80/90 byte system file
   Editor output filekind:
5. Create a CANDE editor file from an EDIT-1 editor file
6. Create an 80 byte system file from an EDIT-1 editor file
   Option 5 or 6: Use columns 1-72 as text
7. Convert editor to output file of: (PRINT,PUNCH,TAPE,DISK,PACK,PCR)
   Output page size or blocking factor
8. Compare input to output editor files and generate difference report
9. Print first 50 lines of input 80 byte system and/or editor file(s)
   Options 8 or 9: Print page size (default = 56)

Options 4-7: Sequence range      .00000000.  THRU  .99999999.
Options 4-5: Resequence (base)  .00000000.  (+inc) .00000000.  XMIT here->
```

Figure 5-1. CANUTL Utility Menu

Executing CANUTL from VCS or GEMCOS

Enter the desired input and output file titles on the menu, and assign nonblank characters to options 1 through 9. Transmit from the last position on the menu.

Executing CANUTL from VCS or GEMCOS

To execute the CANUTL utility from VCS or GEMCOS, you must include an insert parameter as part of the CANUTL command syntax. You can also include file equations in the CANUTL command syntax. After you enter the CANUTL command, the screen illustrated in Figure 5-1 is displayed.

The insert clause is represented by `<obsssssssseeeeeeff>` in the following example:

```
EX CANUTL IN 0 20 <obsssssssseeeeeeff>
```

The parameters provided in the insert clause, `<obsssssssseeeeeeff>`, represent the following information:

Parameter	Description
o	This parameter represents the output file kind, which can be one of the following numeric values: 1 Printer 2 Card 3 Tape 4 Disk 5 Pack 6 Pseudo Card Reader (on disk)
b	This parameter determines the printer backup file and page heading status. You can select one of the following values: 0 No backup. No page heading. 1 Backup permitted. No page heading. 8 No backup. Page heading provided. 9 Backup permitted. Page heading provided.
ssssssss	This parameter represents an 8-digit starting sequence number for records to be copied. Records in the editor file with sequence numbers less than this will not be copied.
eeeeeeee	This parameter represents an 8-digit ending sequence number for records to be copied. Records in the editor

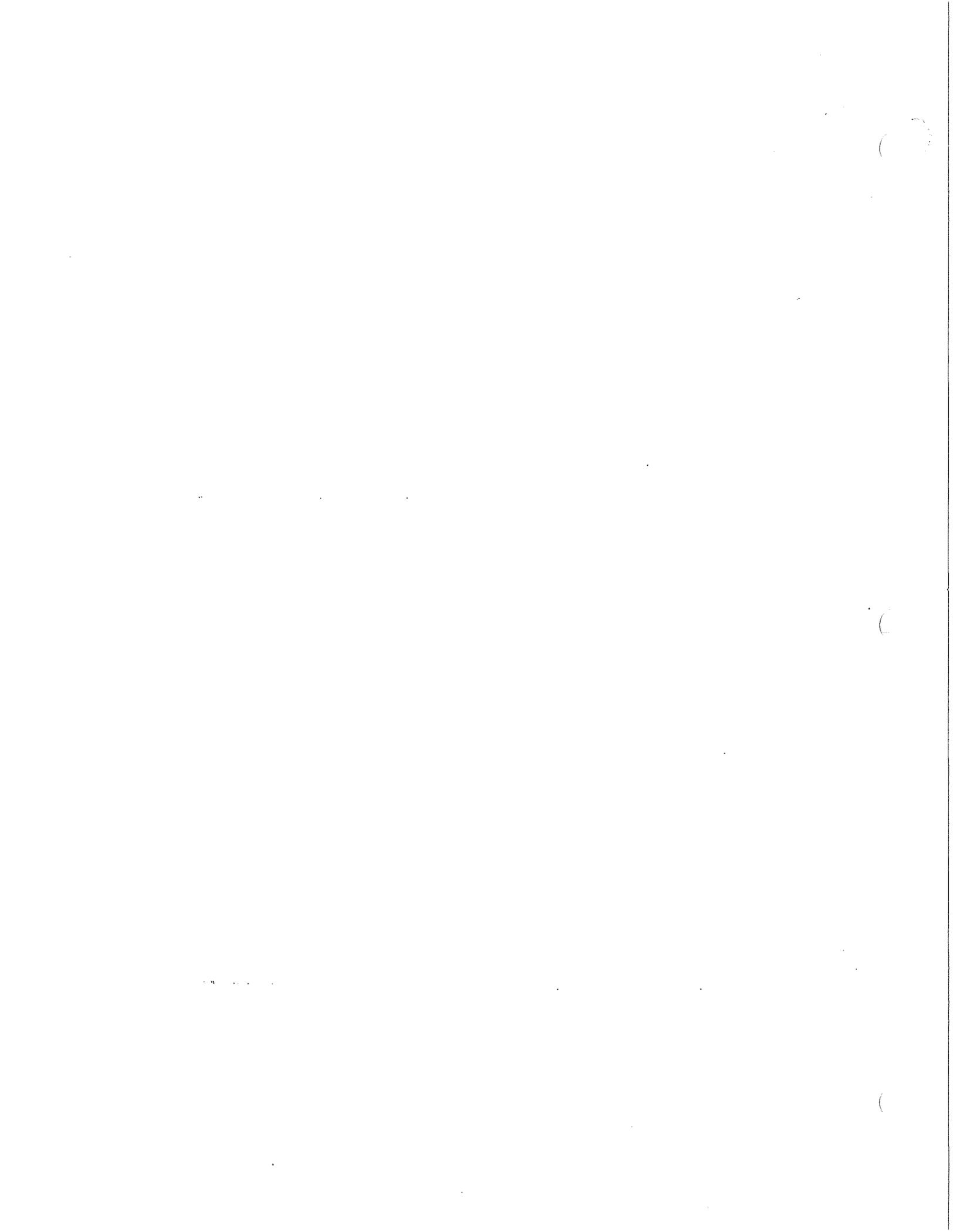
file with sequence numbers greater than this will not be copied.

ff This parameter represents a 2-digit output file blocking factor from 01 to 10. If you specified a printer output file as the file kind, ff represents the page break size, which is a value from 00 through 99.

You can also include file equations in the CANUTL command syntax when executed from VCS or GEMCOS. The syntax of the file equate statements is as follows:

```
FILE INPUT = <family name/> <file name> <hardware type>
FILE OUTPUT = <family name/> <file name> <hardware type>
```

Parameter	Description
INPUT	This is the internal name of the input file.
<family name>	This is the name of the disk pack family on which the file is located.
<file name>	This is the full, 6-character name of the file. You must include the 2-character user ID appended to the 4-character file name.
<hardware type>	This is a 3-character name that indicates the type of family media on which the file is located, either disk pack (DPK) or disk (DSK).
OUTPUT	This is the internal name of the output file.



Section 6

Understanding and Modifying the CANDE Format File

An optional CANDE format file, titled CANFMT, is included with the CANDE software. This file contains

- Responses displayed by the HELP command for CANDE and VF
- Errors messages for CANDE and the VF utility
- Predefined printer backup file formats for use by the VF utility
- An example of a user-defined tabs setting

Note: If the CANFMT file is unavailable, CANDE and VF use internally coded help and error message responses, and VF is restricted to the predefined formats RAW and TRANS.

You can modify all the data in the CANFMT file by editing the file during a CANDE editor session. You can customize the CANFMT file to any language you desire. In addition, you can add custom tab settings to be used by the CANDE editor and custom printer backup file formats to be used by the VF utility.

When editing the CANFMT file, you must use a workstation configured for visible/transmittable carriage return. This prevents trailing spaces in records less than 476 bytes from affecting the display.

This section describes the contents of the CANFMT file and provides guidelines for customizing this file.

Note: Because the record size of the CANFMT file is 476 bytes, you can edit it only with release 3.2 or greater of CANDE.

Structure of the CANFMT File

The CANFMT file is an editor file of file kind DATA. The default file title that CANDE and VF use for this file is *CANFMT ON DPKDPK*. The text length is 476 bytes.

Groups of sequence numbers in the CANFMT file are reserved for various functions as follows:

Sequence Numbers	Reserved For...
0 through 999	CANDE error messages
1000 through 1199	VF errors messages
1200 through 1899	CANDE help responses
1900 through 1999	VF help responses
2000 through 9999	Future use
10000 and above	VF formats supplied by Unisys, user-defined VF formats, and user-defined tab settings

Including Custom Formats and Tab Settings in the CANFMT File

A printer backup file format affects the content and format of a printer backup file listing. To create your own custom format, use the VF FORMAT command syntax to include the format in the CANFMT file. After you have created the format, you can access it from the VF utility by using the FORMAT command. Refer to Section 3 for detailed information on the FORMAT command.

Tab settings affect the location of workstation tabs values. To create custom tab settings, use the CANDE TABS command syntax to include the tab settings in the CANFMT file. After you have created a group of tab settings, you can access them by using the TABS command. Refer to Section 2 for detailed information on the TABS command.

Modifying Error Messages and Help Responses

When modifying the error messages and the help responses contained in the CANFMT file, you must observe these requirements:

- All text must be graphically representable. You can use the following syntax in substitution for control characters:

(pound sign) Message insert character.

[...] Control character sequences. Each character is translated from characters to integers. Leading nulls are used to replace the difference in the translated string length. Each control string must be enclosed in brackets [...]. No validation check is performed on the control characters.

- You must use the following text record layout for error message records:

```
Seq Nbr           (8 UA)
Restore Cmd Line Flag (1 UA) - non-blank character = true
Filler           (1 UA)
Text             (474 UA)
```

- You must use the following text record layout for help response records:

Seq Nbr	(8 UA)
Mult Rec Flag	(1 UA) - non-blank character = true
Filler	(1 UA)
Text	(474 UA)

When the multiple record flag contains a nonblank character, it indicates that the following sequence number record is part of the total text for the help response. This value is used when the total text requirement exceeds 474 bytes. You can use a maximum of four 474-byte records to make up a single help response.

Specifying an Alternate Name for the CANFMT File

You can specify an alternate name for the CANFMT file in either of the following ways:

- Include a CANFMT statement in either the user variable file (UVF) or the system variable file (SVF).
- Modify the name by using the VCSCVB utility of VCS.

It is recommended that you use a CANFMT statement to specify an alternate file name rather than modifying the name with VCSCVB. Changing the file title with that utility affects all users of CANDE and VF.

The syntax of the CANFMT statement is as follows:

```
CANFMT <file title>
```

For example, the following statement changes the CANFMT title to MYFMT ON MYPACK:

```
CANFMT MYFMT ON MYPACK
```

How the System Locates the CANFMT File

To locate the CANFMT file for error messages and responses to the HELP command, the system searches for the CANFMT file under the following file titles in order:

1. The internally coded global file title, which is *CANFMT ON DPKDPK* by default
2. The global file name, CANFMT, on the user's default family
3. The global file name, CANFMT, on DISK

To locate the CANFMT file for formats, the system searches for the CANFMT file under the following file titles in order:

How the System Locates the CANFMT File

1. The file title supplied with the **FORMAT** command
2. The internally coded user file title, which is *CANFMT ON DPKDPK* by default (the same as the global file title)
3. The internally coded global file title, which is *CANFMT ON DPKDPK* by default
4. The global file name, **CANFMT**, on the user's default family
5. The global file name, **CANFMT**, on **DISK**

To locate the **CANFMT** file for custom tab settings, the system searches for the **CANFMT** file under the following file titles in order:

1. The internally coded user file title, which is *CANFMT ON DPKDPK* by default (the same as the global file title)
2. The internally coded global file title, which is *CANFMT ON DPKDPK* by default
3. The global file name, **CANFMT**, on the user's default family
4. The global file name, **CANFMT**, on **DISK**

Note: *You can change the name of the internally coded user file title by using the **VCSCVB** utility of **VCS**. This change is also applied to the internally coded global file title.*

Section 7

Merging Editor and System Files

This section explains the patch merge utility, which lets you merge patch files with a master file.

Patch Merge Utility Guidelines

By using the patch merge utility, you can merge an unlimited number of 80-byte CANDE editor files and system files.

You cannot use system files with blank sequence numbers as input to the patch merge utility. Using such an input file results in the termination of the patch merge utility with an error message.

You can save processing time by merging multiple patch files first, and then merging the resulting output file with the master file. If you merge the master file with a single patch file first, the intermediate output file will contain the entire master file. This intermediate file is reread for further merges, possibly resulting in a lengthy processing time.

Patch Merge Utility Features

The patch merge utility provides the following features:

- Ability to list both the input patch files and the merged output file.
- Optional assignment of a temporary 10-character patch ID to each input file to identify records in the merged file. The printer listing will contain these patch IDs.
- Flagging in the printer listing of records that have matching sequence numbers. The sequence number from the input file takes precedence over the sequence number from the output file.
- Ignoring of \$VOID records that contain a smaller void range than a matching sequence number in the master file. The master file record is retained, and a warning message is generated in the printer listing. If you have not requested a printer listing, a message is displayed on the screen of your terminal.
- Optional insertion of a comment record within the output file. This record provides a history of each run in which the input file was used to create the output master patch file.

Files Used and Generated by the Patch Merge Utility

The files used and generated by the PATMRG utility are described in Table 7-1.

Table 7-1. Files Processed by the Patch Merge Utility

File Name	Description
SCRIPT	Instructional input file
PTCHIN	Input patch file
MSTRIN	Input master file (used only when more than two patch files are merged)
PATMST	Output file containing merged files
PRINTERFILE	Printer file

Input files can be of file kind EDITOR or SYSTEM. The output file (PATMST) defaults to the EDITOR file kind.

Single-pass execution (switch 7 = 1) results in a two-file merge using only the LIST option. This single-pass execution requires that you file-equate the MSTRIN, PTCHIN and PATMST files. For example:

```
EX PATMRG (1) FILE MSTRIN <mfid>/<fid> FILE PTCHIN<mfid>/<fid> FILE PATMST  
<mfid>/<fid>
```

Single-pass execution is limited to merging only two CANDE editor files.

Providing Instructions in the SCRIPT File

You can provide instructions to the patch merge utility by supplying a SCRIPT file. The SCRIPT file can contain the titles of files to be merged, as well as a variety of options that control the generation of printer listings and the insertion of a patch ID and a date and time stamp in the merged files. The SCRIPT file is read sequentially.

Including Options in the SCRIPT File

You must precede all options within the SCRIPT file with the following characters:

\$#

You can specify multiple options on the same line.

Each of the dollar options except FILE PATMST and SUMMARY can be specified at any point in the SCRIPT file.

The options you can include in the SCRIPT file are described as follows:

Options	Description
FILE	This option enables you to file-equate the output file, PATMST, to a file name of your choice. The syntax for this option is

- FILE PATMST - <file name>

=	ON-<file name>	<file kind>
---	----------------	-------------

The default family name is DISK.

The file kind can be EDITOR80 or SYSTEM80. The default file kind is EDITOR80.

LIST	This option generates a printer listing of the output file (named PATMST by default). The default value for this option is OFF (RESET).
LISTP	This option generates a printer listing of each input file. The default value for this option is OFF (RESET).
NODUP	This option causes an occurrence of a duplicate sequence number to be treated as a fatal error. The default value for this option is OFF (RESET).
PAGEP	This option inserts a page break after the printer listing of each file. This option requires that the LISTP option be ON (SET). The default value for this option is OFF (RESET).
PATCHID <10 char max string>	This option inserts a 10-character string into columns 81-90 of the printer listing of the output file. Certain file kinds also enable the PATCHID string to be inserted into the output file itself. To use the PATCHID option, the LIST option must be ON (SET). The default value for this option is OFF (RESET).
-	This option removes the patch record immediately preceding the \$#- record.
- sequence range	This option removes the specified range of patch records. Any occurrence of a matching record within the specified range is deleted from the master patch file when this input file is processed.

Providing Instructions in the SCRIPT File

SUMMARY <sequence number> This option inserts a date-and-time stamp record at the specified sequence number. If a record already exists at this location in the output file, the date-and-time stamp is inserted at the first available sequence number plus one. This option also lists each file used to create the output file, with its associated date-and-time stamp. The default value for this option is off (RESET).

Example

The following example shows possible dollar options in a SCRIPT file:

```
00001000$#SET LIST LISTP PAGEP00001500$#FILE PATMST = PATMST ON PATPCK
(SYSTEM80)00002000$#SUMMARY 10000003000$#PATCHID
NOV87MASTRO00004000$#MSTRcc ON MSTPCK00005000$#PATCHID
NOV87PAT.100006000$#PT1AaaON PATPCK00007000$#PATCHID NOV87PAT.2
```

Specifying File Titles in the SCRIPT File

You must specify file titles in the SCRIPT file when you are merging more than two files per execution. In addition, including the titles of input files in the SCRIPT file prevents you from having to provide file equations every time you run the patch merge utility. The format of each file specification in the SCRIPT file is as follows:

```
- <file name > _____|
      |_____|
      ON <family name>
```

The default family name is DISK.

Input files are indicated in the SCRIPT file by the absence of a flag character. The specification of a PATCHID record immediately preceding the file specification is associated with that file. The format of an input file specification is as follows:

```
- <file name >[ <family name >] [ <file kind >] _____|
```

The file kind applies only if the input patch file is a system file. The file kind can be any of the following:

ADS	DASDL
ASSMBLR	FORT77
BINDER	PASCAL
BPL	RPG
COBOL	WFL
COBOLV	

Including Comments in Input Files

You can insert comments in input files by preceding the comment with the following indicator:

```
 $#COMMENT
```

The comment indicator must start in the first text position of each file kind. For example, the first text position in a COBOL file is column 7.

Records marked as comments appear only in the printer listing of the output file and not in the output file on disk.

Example

The following example shows how comment records might look in an input file with a file kind of COBOL.

```
000100$#COMMENT This is a comment record that will appear
000200$#COMMENT only in the printer listing, if requested
000300$#COMMENT (LIST). It will not appear in the output
010300$#COMMENT PATMST file
014600      LIST-VARIABLE          PIC X
015700      MOVE INPUT-VAR TO LIST-VARIABLE
```

CANDE Menu Interface

CANDE provides a menu interface to the PATMRG utility. You can merge only two patch files with this interface.

To display the menu, enter *PATMRG*. The menu that is displayed is shown in Figure 7-1.

PATMRG MENU	
SCRIPT FILE:	ON
-----OR-----	
INPUT MSTRIN:	ON
INPUT PTCHIN:	ON
OUTPUT PATMST:	ON
TEN CHAR PATCHID FOR MSTRIN FILE (OPTIONAL)	
TEN CHAR PATCHID FOR PTCHIN FILE (OPTIONAL)	
PRECEDE NON-USERCODED FILE NAME(S) WITH * XMIT HERE ->	

Figure 7-1. PATMRG Utility Menu

Error and Warning Messages

Use of the SCRIPT file enables you to use DO files within CANDE to interact with PATMRG. If a fatal error occurs during the merge, and the merge was initiated through the COMPILE command, IX1 will be set to nonzero at the end of the job. This causes a syntax error message to be displayed. In processing the DO file, CANDE will detect this abnormal termination and take the branch statement in the file. For example:

```
//BEGINJOB
?CMP (8250) PATMRG FILE SCRIPT (SFILac) DSK
//IF COMPILEDOK THEN
//BEGIN
  COMPILER 8183:MFIL ON PATPCK % Contains file-equate
                               % statements for master
                               % source as MFILac is
                               % the patch file
//IF COMPILEDOK THEN
//BEGIN
?PRINT @MLIST
.
.
.
//END ELSE
DISPLAY "MYPROG COMPILATION FAILED"
//END ELSE
DISPLAY "MFILac PATMRG PROCESS FAILED"
//ENDJOB
```

Error and Warning Messages

The following error and warning messages can result during use of the PATMRG utility.

INVALID FILE TYPE FOUND IN PROCEDURE GET_TOKEN

- A call to the procedure GET_TOKEN did not contain a file specification of either the SCRIPT, MSTRIN or PTCHIN files.
- This is a fatal error and should be reported with the dump that is automatically generated.

TOO MANY RECORDS - MAX EDITOR DIRECTORY SIZE EXCEEDED!

- The maximum output CANDE editor file size has been exceeded based on the RPA (records per area) that was started with. An accept response is requested from the user indicating whether the user wishes to retain this incomplete file.
- To avoid this type of error, use smaller input patch files.

INVALID FILE TYPE FOUND IN PROCEDURE WRITE_PATMST_EDITOR

- A call to the procedure WRITE_PATMST_EDITOR did not contain a file specification of either the MSTRIN or the PTCHIN file.
- This is a fatal error and should be reported with the dump that is automatically generated.

WARNING! PREVIOUS VOID REC RETAINED.SEQ # = nnnnnnnn

- This warning message indicates a conflict between the ending void range of matching void records in the MSTRIN and PTCHIN files. The MSTRIN contains a higher ending value than has been retained. This message appears only when the LIST option is not specified.

WARNING! RECORD DROPPED W/GTR END VOID.SEQ # = nnnnnnnn

- This warning message indicates that a void range in the PTCHIN file has resulted in the deletion of a MSTRIN void record that contained a higher ending value. This message appears only when the LIST option is not specified.

NUMBER OF RECS DROPPED FOR INVALID EDITOR REC LENGTH = nnnnnnnn

- This is a warning message indicating the number of CANDE editor records that were found to contain an invalid length during the processing of the last PTCHIN file.

INVALID PATMST FILE TYPE IN CREATE_FINAL_PATMST

- A call to the procedure CREATE_FINAL_PATMST did not contain a recognized output file type specification (that is, EDITOR80 or SYSTEM80).
- This is a fatal internal error and should be reported with the dump that is automatically generated.

**EDITOR FILEKIND IS INVALID FOR DETERMINING COMMENT CHAR.
POSSIBLE INDICATION OF CORRUPTED EDITOR FILE**

- A failure occurred while attempting to determine the matching comment character for the file kind. This warning can only be generated in conjunction with the SUMMARY option.

<file title>:

RECORD DROPPED DUE TO LENGTH > 99 CHARS.

SEQ # = nnnnnnnn

POSSIBLE INDICATION OF CORRUPTED EDITOR FILE

- This is a warning message indicating that during the processing of a CANDE editor input file, a record length that exceeds the maximum allowed (99) was encountered. The record is dropped and processing continues with the next record in the block. This is possibly an indication of a corrupted editor file.

<file title>:

**INVALID SEQUENCE LENGTH. DROPPING REMAINDER OF BLOCK#
nnnnnnnn**

- This is a warning message indicating that during the processing of a CANDE editor file an invalid sequence length was found in a record. The block is skipped and processing continues with the next block of records in the file. This is possibly an indication of a corrupted EDITOR file.

<file title>:

**INVALID END VOID SEQ NUMBER IN REC SEQ NUMBER.
SEQ # nnnnnnnn**

- This is a warning message that a nonnumeric ending void range was found in a void record. This is counted as an error, which will result in the setting of IX1 to nonzero at EOJ and the display of the total error count. Processing continues with the next record.

**<file title> CONTAINS INVALID SEQ NUMBERS FOR SPECIFIED
FILEKIND**

- This is a fatal error message indicating that a non-numeric sequence number has been encountered in this system input file. This is based on the record positions that should match the sequence number locations as indicated by the user-specified file kind.
- Determine the correct file kind and reexecute the patch merge utility.

<file title> HAS A SEQUENCE NUMBER OF ALL BLANKS

- This is a fatal error message indicating that the system input file processed contains a missing sequence number. This could be the result of an incorrect file kind specification or an omitted sequence number. ALL system input files must contain a valid sequence number within each record.

INVALID SEQ NUMBER FOR SUMMARY OPTION

- An invalid sequence number was specified on the SUMMARY option. This option is ignored and processing continues.

INVALID PATMST FILE EQUATE

- The FILE PATMST option contains an invalid specification. This is a fatal error that results in PATMRG termination.

MISSING CCI PATCHID VALUE - IGNORED

- The PATCHID option did not contain a value. This option is ignored and processing continues.

INVALID CCI OPTIONAL: <token>

- An invalid \$ option was found. This is considered a fatal error and job execution is aborted.

<file title> HAS AN UNRECOGNIZED FILEKIND

- This is a fatal error indicating that the CANDE editor input file being processed does not contain a recognized file kind. This is a possible indication of a corrupted editor file.

INVALID FILETITLE: <token>

- This is a fatal error indicating the input file title specification in the SCRIPT file is invalid.

INVALID OR CORRUPTED SCRIPT FILE

- This is a fatal error indicating the specified SCRIPT file is not a CANDE editor file or an 80-byte system file.

INVALID OR CORRUPTED MSTRIN FILE

- This is a fatal error indicating the specified MSTRIN file is not a CANDE editor file or an 80-byte system file.

INVALID OR CORRUPTED PTCHIN FILE

- This is a fatal error indicating the specified PTCHIN file is not a CANDE editor file or an 80-byte system file.

INVALID FILE TYPE FOUND IN PROCEDURE VALIDATE_INPUT_FILE

- A call to the procedure VALIDATE_INPUT_FILE did not contain a file specification of either the SCRIPT, MSTRIN or PTCHIN file.
- This is an internal error and should be reported with the dump that is automatically generated.

<file title> IS NOT A CANDE editor FILE OR FILE IS CORRUPTED

- The input file being processed was not found to be a valid CANDE editor file as determined from examination of the header record. This is possibly an indication of a corrupted EDITOR file.

<file title> IS A PRIVATE FILE. ACCESS DENIED

- The accesscode that PATMRG is executing with does not match the one contained in this private CANDE editor file. Job execution terminates.

Error and Warning Messages

<file title> IS AN INVALID OR CORRUPTED CANDE editor FILEKIND

- Examination of the CANDE editor file header resulted in an invalid file kind specification. Job execution terminates. This is possibly an indication of a corrupted EDITOR file.

<file title> FILEKIND DOES NOT MATCH

- The file being processed was found to be a of a nonmatching file kind as determined initially by the MSTRIN file. Job execution terminates.

EMPTY SCRIPT FILE

- The specified SCRIPT FILE does not contain any records. Job execution terminates.

<file title> IS NOT PRESENT. RETAINING OUTPUT PATMST (IF CREATED)

- This is a fatal error indicating the file displayed is not present. If a PATMST file was created thus far, it is retained and PATMRG goes to end of job.

<file title>:

IS AN EMPTY FILE. PROCEEDING TO NEXT FILE IN SCRIPT

- The file being processed does not contain any records. This file is skipped and processing continues with the next file in the SCRIPT file.

UNEXPECTED EOF OR INVALID KEY ON INPUT FILE. DO YOU WANT TO SAVE THIS INCOMPLETE PATMST FILE?(Y OR N?)

- An error has occurred in which processing cannot continue.
- Respond to the error message to save or purge the PATMST output file generated thus far.

Appendix A

Error Messages

This appendix contains an alphabetical listing of the warning and error messages that you might encounter when using CANDE and provides corrective action when applicable.

Accesscode NOT authorized for PCR generation and/or job execution

- See your accesscode administrator to get authorization.

Active editfile not 4 chars. Cannot save as recall filename

- Your SAVE command was rejected because the active editor file is not a usercoded file.
- Use the *SAVE AS <file title>* form of the SAVE command.

Appending terminated. Line number too large

- The total number of records being appended exceeds the highest line number of the file.
- Perform the append operation again and use a smaller increment value.

Base parameter too large

- The base value exceeds the sequence number range in the editor file.
- Reenter the command and specify a smaller base value.

CANUTL NOT available. Required for file directory reconstruction

- The CANUTL utility was not found, so the editor file reconstruction cannot occur.
- Load CANUTL from a backup medium or from the release tape.

Corrupted CANFMT file

- The CANFMT file contains one or more corrupted records.
- Reload the CANFMT file from a backup copy or from the release tape.

Current format does not support error/warning and/or summary definitions

- The VF utility did not find error and warning strings or summary strings for the specified command.

Error Messages

- Use a format that includes these strings, or update the format in use by defining error, warning, and summary strings with the *VF FORMAT* command.

Current format does not support SEQ mode

- The format in use does not have a defined sequence number location.
- Use a format that supports sequence mode, or update the format in use by assigning offset and length values to the SEQNBR option of the *VF FORMAT* command.

Default family change NOT allowed

- Your accesscode does not permit you to change the default family.
- Contact your accesscode administrator to update your accesscode.

Duplicate file

- Another file exists on the same family with the same name.
- Remove or change the name of the existing file.

Duplicate NOT done, maximum increment = #####

- The increment value that you specified with the DUPLICATE command is too large for the available sequence range.
- Enter the DUPLICATE command again, and use an increment that is less than or equal to the value in the error message.

Duplicate NOT done, possible line number overlap

- The available sequence range does not allow for the addition of any new lines. Even an increment of 1 would result in line number overlap.
- Modify the source file to enlarge the available sequence range.

Editor: Invalid command

- The CANDE command you entered was misspelled or contained incorrect syntax.
- Refer to Section 2 for the syntax of all CANDE editor commands.

ERROR! DO label NOT specified

- A label did not follow the “at” sign (@) in the DO command syntax.
- Enter the DO command again, and type a label after the “at” sign (@). See Section 2 for the syntax of the DO command.

ERROR! Invalid label token

- The label following the “at” sign (@) in the DO command syntax contained invalid characters.
- Enter the DO command again with a correct label name. See Section 2 for the syntax of the DO command.

ERROR! Matching ending delim or string delims NOT found

- The command you entered contained string references that did not have matching delimiters.
- Enter the command again, enclosing each string with the same delimiters.

ERROR! Patch cmd greater than allowed 72 char spo maximum

- The PATCH system command contained more than 72 characters.
- Enter the command again using 72 characters or fewer.

File: #####/##### family NOT available

- The family specified in the file title of this command is not available.
- Enter the command again and correct a possibly misspelled family name or specify a family that is online.

File ##### NOT available

- The system cannot access the file you specified.
- Determine if the file is available by using the SHOW system command.

File: ##### Not present

- The system cannot find the file you specified.
- You probably typed the file name or the family name incorrectly. Determine the correct file title and correct your entry.

File: ##### Not present or invalid security access

- The specified file is missing or is unavailable due to denied security access. The file might have read-only status, and you are trying to write to it.
- Determine if there is a missing file situation and correct the problem. You can determine the security use of the file by entering the SECURITY command without parameters.

Filekind NOT changed. Line number magnitude too large

- The TYPE command was rejected because the target file contains sequence numbers that exceed the maximum value for the new file kind.
- Resequence the target file, specifying a smaller line number value.

Error Messages

Filekinds NOT identical

- The command could not be completed because the two editor files it was using have different file kinds.
- Change the file kinds by using the TYPE command.

Illegal security use for file #####

- The SECURITY command was rejected because of an invalid use type specification.
- Refer to "SECURITY Command" in Section 2 for the valid security use types. Then correct the syntax and reenter the command.

Illegal use of edit file

- The command was rejected because the active editor file was indicated as both the source and the target file.
- Enter the command again with the correct syntax.

Illegal use of reserved character

- The command syntax includes characters reserved for special use.
- Enter the command again with the correct syntax.

Increment parameter too large

- The specified increment generates sequence numbers that are greater than are valid with this file kind.
- Enter the command again and use a smaller increment value.

Input screen error count > 3. Remaining input buffer flushed!

- The input screen contains more than three command errors. CANDE treats the screen as erroneous input and ignores it.

Internal ViewFile error

- An internal logic error has been detected in the View File Utility.
- Submit a UCF with any program dumps that were generated to your Unisys customer service representative.

Invalid batch file. Text length > 80 chars

- The batch file contains a record that is greater than the allowed length of 80 characters.
- Correct the file and resubmit it.

Invalid column range

- The command syntax contained a column value of zero or a column value greater than the text length allowed for the file kind.
- Correct the syntax and reenter the command.

Invalid family name: #####

- The family name specified in the command syntax is invalid.
- Correct the family name and reenter the command.

Invalid filekind

- The file kind specified in the command syntax is invalid.
- You can list the file kinds valid for CANDE by entering *HELP LIST*. Then correct the file kind and reenter the command.

Invalid filename: #####

- The file name specified in the command syntax is invalid.
- See the beginning of Section 2 for the correct syntax for a file name. Then correct the file name and reenter the command.

Invalid filetype. Be sure to precede non-usercoded files with an asterisk

- The file title specified in the command syntax is invalid.
- See the beginning of Section 2 for the correct syntax for a file title. Then correct the file title and reenter the command.

Invalid format definition: #####

- The format definition you want to use with the View File utility contains invalid syntax.
- See the discussion of the **FORMAT** command in Section 3 for the correct syntax for a format definition. Then correct your format definition.

Invalid key or IO result on last read/write

- The reading or writing of the last record failed, usually indicating the corruption of the editor file directory.
- Reorganize the file by using the **REORGANIZE** command described in Section 2.

Invalid log info BCT response. CMD ABORTED

- The MCP response to a request for log information indicated a problem with the request.

Error Messages

- Submit a UCF with a dump of the CANDE editor to your Unisys customer service representative.

Invalid Mark/Bound sequence range

- The value used with the MARK or BOUND command is out of the range of the sequence numbers in the file.
- Correct the sequence range value and reenter the command.

Invalid numeric token: #####

- The command syntax contains an invalid numeric token.
- Correct the syntax and reenter the command.

Invalid or missing sequence range

- The command syntax does not contain a sequence range or contains an invalid sequence range.
- Include a sequence range or correct the current sequence range and reenter the command.

Invalid overriding usercode: #####

- The usercode you entered for the file name was incorrect.
- Reenter the command using the correct usercode.

Invalid security attributes

- The syntax for an attribute of the SECURITY command is incorrect.
- See Section 2 for the syntax of the SECURITY command. Then correct the syntax and reenter the command.

Invalid syntax. Missing family name

- The command syntax requires a family name, which is missing from the syntax.
- Supply a family name and reenter the command.

Invalid tab type value in editor file header record

- The editor file just opened with the GET command contains invalid tab values in the file header.
- Correct the tab values in the editor file header by using the TABS command with the SAVE option as described in Section 2.

Invalid task number or task not available

- The task number in the JMGR command syntax is invalid or unavailable.

- Reenter the command using a valid task number.

Invalid Teach/Help command or syntax requested

- The HELP or TEACH command contained invalid syntax or an incorrect command name.
- See Section 2 for the syntax of the HELP and TEACH commands. Then correct the syntax and reenter the command.

Invalid userid

- The command syntax references an invalid userid.
- Correct the userid and reenter the command.

Last command filetype NOT available

- The file title reference is not available.
- Correct the file title and reenter the command.

Line number too large

- The line number specified in the command syntax is larger than the sequence length allowed for the file kind of the active editor file.
- See Table 2-1 under the MAKE command in Section 2 for the valid sequence lengths for CANDE file kinds. Then correct the line number and reenter the command.

Masking chars NOT allowed in filetype. DO/START request IGNORED

- The file title contained masking characters.
- Correct the file title and reenter the command.

Matching listing name for batch file @ reference NOT found

- The reference to a printer file name in a batch file command was not found in previously monitored tasks.
- Correct the printer file name in the batch file reference.

Matching parameter NOT found: #####. Batch file ABORTED

- A parameter substitution reference was not found in the DO command syntax.
- See Section 2 for the syntax of the DO command. Then correct the syntax and reenter the command.

Maximum directory level exceeded

- The total record count for the editor file exceeds the maximum file capacity of approximately 3,500,000 records.

Missing BEGINJOB statement

- The batch file does not contain a BEGINJOB statement.
- Include a BEGINJOB statement and execute the batch file again. See Section 4 for a discussion of the batch file structure and syntax.

Missing Editfile

- The command you entered requires an active editor file.
- Establish an active editor file by opening an editor file with the GET command or creating a new editor file with the MAKE command. Then reenter the command.

Missing or invalid family name

- The command syntax contains an invalid family name or does not contain a family name.
- Supply a valid family name and reenter the command.

Missing parameter ending delim. Batch file ABORTED

- The DO command contains an invalid delimiter.
- See Section 2 for the syntax of the DO command. Then correct the syntax and reenter the command.

Missing quote in batch file display request. Display IGNORED

- The batch file DISPLAY command does not include a display string enclosed in quotation marks.
- See Section 4 for the syntax of the DISPLAY command. Then correct the syntax and resubmit the batch file.

Move NOT done, maximum increment = #####

- The MOVE command increment is too large for the available sequence range.
- Assign an increment that is less than or equal to the value in the error message and reenter the command.

Move NOT done, possible line number overlap

- The existing sequence number range is too small to accommodate the addition of new lines.
- Resequence the source file to expand the sequence number range.

Multiple active batch files is NOT allowed

- The batch file contained a DO command, which is illegal.
- Remove the DO command, and resubmit the batch file.

No ETX present in workstation input

- The input message did not contain a terminating ETX.
- Turn on tracing with the DEBUB command. Then recreate the error and submit a UCF to your Unisys customer service representative.

Non-existing line or mismatched filekinds

- The sequence number reference does not exist or is beyond the largest number in the editor file due to mismatched file kinds.
- Verify the file kinds and the sequence number range. Correct the necessary syntax and reenter the command.

No seq position defined in current format

- The View File format in use does not contain a sequence number definition.
- Correct the format by using the FORMAT command or use a different format.

Not viewing a printer backup file

- The View File command is valid only for viewing printer backup files.

No file ##### on #####

- The file title used in the command was not found.
- Correct the file title syntax and reenter the command. See the beginning of Section 2 for a discussion of the file title syntax.

Non-supported EDITOR filekind

- The editor file referenced in the command has a file kind that is not supported by CANDE.
- You can list the file kinds valid for CANDE by entering *HELP FILEKIND*.

Offset or length greater than record length

- The sequence number offset or record length value used with the command exceeds the record length of the file being viewed.
- Adjust the offset or length to conform with that specified by the file kind. See Table 2-1 under the MAKE command in Section 2 for the valid record lengths and sequence number positions for the various CANDE file kinds.

OUTPUT print file not open

- The VF *CLOSE* command did not find the printer output file open.
- Use the VF *OUTPUT* command to open the file and then use the *CLOSE* command.

Error Messages

Previous recall file found, BUT IN USE!! GET cmd aborted

- The RECALL command found the recall file in use.
- Correct the file-in-use situation , and then enter the RECALL command again.

Printing of entire log not allowed. Use log analyzer

- The RLOG or SLOG command did not contain options for printer output requests.
- Reenter the command, using at least one subset option.

Recall family change NOT allowed

- Your accesscode does not allow you to change the family on which your recall files are stored.
- Contact your accesscode administrator to get your accesscode updated.

Recall file NOT available

- The RECALL command did not find a recall file.
- Verify the family on which the recall file is stored, and then use the family name as part of the file title.

Requested batch/job file NOT present

- Verify the file title and reenter the command.

Requested display length exceeds maximum buffer capacity

- Reenter the command with options to subset the response, if desired.

Requested family not available

- Verify the family name and correct it if necessary. Then reenter the command.

Requested file: ##### is NOT an EDITOR file

- Make sure you typed the correct file name. Reorganize the file with the REORGANIZE command if necessary.

Requested file NOT available/present

- Verify the file title and correct it if necessary. Then reenter the command.

Requested file not present or invalid security access

- The file might be a private file that belongs to another user.
- Make sure the file exists and then determine the owner of the file.

Requested record beyond file EOF

- The VF command syntax requested a record length that extends beyond the end of the file.
- Reenter the command using a record value lower than the ending record of the file.

Requested skip value too large

- The skip option value for a multi-page response is larger than the total number of lines in the response buffer.
- Reenter the command using a smaller skip value.

Resequencing NOT done, maximum increment = #####

- The available sequence range is too small for the resequencing increment.
- Reenter the command using an increment value that is less than or equal to the value in the error message.

Resequencing NOT done. Result would generate duplicate line numbers

- The available sequence range is too small for resequencing.
- Reenter the command using a smaller base value or use the SEQUENCE command to enlarge the available sequence range.

Resequencing NOT done. Use 'MOVE' command

- Records being resequenced overlap records not being resequenced.
- Use the MOVE command to insert the records into the file.

Specified string greater than 80 characters

- Reenter the command using a string that is less than or equal to 80 characters.

Text size too big

- The text size is larger than that allowed by the file kind of the active editor file.
- See Table 2-1 under the MAKE command in Section 2 for the text lengths allowed by each of the valid CANDE file kinds. Then reenter the command using a valid text size.

There is currently not an active display file

- The VF command you entered was rejected because there was no active view file.

Error Messages

- Open a view file by using the VIEW command. See Section 3 for the syntax of this command.

Token text size too big

- Shorten the text of the token and reenter the command.

Unrecognized or invalid input record and/or block size

- The requested file does not match supported input file requirements.
- Correct the input file and reenter the command.

VF: Invalid Command

- The VF command contained invalid syntax.
- See Section 3 for the syntax of VF commands. Then correct the syntax and reenter the command.

VF utility not active. Use VIEW command to initiate

- The CANDE input command was directed to the VF utility, but VF was not active.
- Initiate the VF utility by entering the VIEW command, and then reenter the VF command.

Wrong filekind or missing comment char. Start jobfile ABORTED

- The requested file was determined to be of COMPTS format, but the format of the comment character control record was invalid for the file kind.
- Correct the file and resubmit it. See the discussion of the file-equated execution format (formerly called the COMPTS format) under the START command in Section 2.

Unable to support seq range with system format file

- The command syntax contained a reference to a system file and a sequence range.
- Reenter the command and omit the sequence range.

Zero is invalid for #####

- Use a value other than zero in the command syntax.

#####sequence number NOT found

- Reenter the command using a valid sequence number.

Appendix B

Installing CANDE

To install the 3.2 release of CANDE, follow these steps:

1. Load the release tape, which is titled CE1320.
2. Copy the entire contents of the tape by using the following syntax:

```
COPY == FROM CE1320 (TAPE)
```

3. If you are using GEMCOS, go to Step 4. If you are planning to run this release of CANDE in a VCS environment, perform the following steps:
 - a. Add CANDE and the View File utility as applications by using the APPL-ADD syntax as follows:

```
APPL-ADD, CANDE, COPY = JC
```

```
APPL-ADD, ViewFile, COPY = CANDE
```

- b. Add transactions to invoke CANDE and the View File utility by using the TRAN-ADD syntax as follows:

```
TRAN-ADD, CD, COPY = JCTRAN-ADD, CANDE, COPY = RSPO
```

```
TRAN-ADD, VIEW, COPY = CD
```

4. If you are planning to run this release of CANDE in a GEMCOS environment, you must configure your Transaction Control Language (TCL) program to use station security. You will also need to use MCP access security. In addition, you must add the following TCL program declarations:

- a. For CANDE, add the following TCL declaration:

```
PROGRAM CANDE (n):  
  INTERFACE = PORT.  
  SINGLEUSER = TRUE.  
  ASSIGNNOTIFY = TRUE.  
  SECURITYATTRIBUTES = TRUE.  
  EXECUTE = ON DEMAND.  
  MAXCOPIES - 30.  
  ATTRIBUTES = "(, "<MCS name> ", " <portfile name>") LOCK".
```

Customizing a CANDE Session

- b. For the View File utility, add the following TCL declaration:

```
PROGRAM ViewFile (n):  
  INTERFACE = PORT.  
  SINGLEUSER = TRUE.  
  ASSIGNNOTIFY = TRUE.  
  SECURITYATTRIBUTES = TRUE.  
  EXECUTE = ON DEMAND.  
  MAXCOPIES - 30.  
  ATTRIBUTES = "(,"<MCS name> "," <portfile name>") LOCK".
```

Note: The 3.2 release of CANDE can coexist on the same host with previous releases of CANDE. Be aware, however, that reverting to an release earlier than 3.2, and then editing a file created with the 3.2 release could destroy various file enhancement features, such as the internal password, creation date information, and so on.

Customizing a CANDE Session

You can customize each user session at initialization by including any of the CANDE 3.2 version commands in a user variable file (UVF) or a system variable file (SVF). For details on creating these batch files, refer to Section 4.

Appendix C

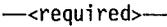
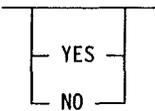
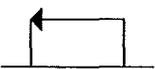
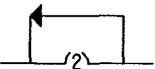
Understanding Railroad Diagrams

What Are Railroad Diagrams?

Railroad diagrams are diagrams that show you the rules for putting words and symbols together to form commands and statements that the computer can understand. These diagrams consist of a series of paths that show the allowable structure, constants, and variables for a command or a statement. Paths show the order in which the command or statement is constructed. Railroad diagrams are intended to

- Show the mandatory items.
- Show the user-selected items.
- Present the order in which the items must appear.
- Show the number of times an item can be repeated.
- Show the necessary punctuation.

The following table lists the basic symbols found in a railroad diagram.

Symbol	Explanation
	A terminating vertical bar indicates that the command or statement can be followed by another command or statement.
	A terminating right arrow indicates that the diagram is incomplete, and continues with the next diagram.
	A required item is a constant, variable, or punctuation symbol that must be entered in a command or statement.
	A user-selected item is a constant, variable, or punctuation symbol that can be entered in a command or statement. As in this example, a user-selected item might be optional.
	A loop indicates that an item or group of items can be repeated.
	A bridge indicates the maximum number of times a loop, a required item, or a user-selected item can be repeated.

Following the Paths of a Railroad Diagram

The paths of a railroad diagram lead you through the command or statement from beginning to end. Some railroad diagrams have only one path, while others have several alternate paths. The following examples show two railroad diagrams and possible command and statement constructions based on the paths of these diagrams.

Example 1



Sample Input

28 QT

28 QT END

28 QT ALL

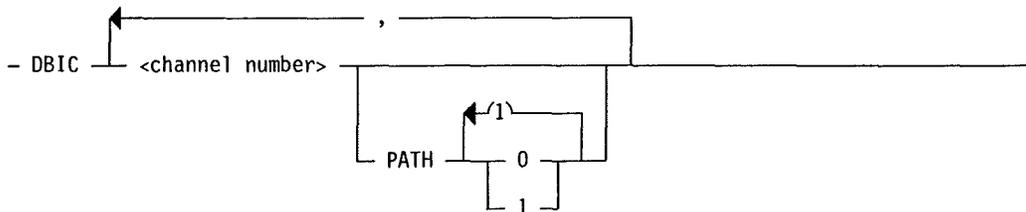
Explanation

QT is a required constant that cannot be altered. Because no part of the word is underlined, the entire word must be entered. The program mix number (28) is a required variable supplied by the user.

END is an optional user-selected item.

ALL is an optional user-selected item.

Example 2



Sample Input

DBIC 3

DBIC 3 PATH 1

DBIC 3 PATH 1, 4 PATH 0

Explanation

DBIC is a required constant that cannot be altered. Because no part of the word is underlined, the entire word must be entered. The channel number (3) is a required variable supplied by the user.

PATH is an optional user-selected item, and the path number (1) is a required user-selected item when a PATH is specified.

Because the diagram contains a loop, more than one channel number can be specified.

Bibliography

- B 2000/B 3000/B 4000/V Series BPL Compiler Programming Reference Manual* (5024789). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series COBOL ANSI-74 Compiler Programming Reference Manual* (5026644). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series DMSII Operations Reference Manual, Volume 1: Database Creation* (5026578). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series DMSII Operations Reference Manual, Volume 2: Database Administration* (5023039). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series DMSII Inquiry Operations Guide* (5023047). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series FORTRAN ANSI-77 Compiler Programming Reference Manual* (5023054). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series GEMCOS Implementation and Programming Reference Manual* (5026313). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series RPG Compiler Programming Reference Manual* (5023179). Unisys Corporation.
- B 2000/B 3000/B 4000/V Series WFL Compiler Programming Reference Manual* (5023229). Unisys Corporation.
- V Series Systems Operations Guide, Volume 1: Installation* (4127 4804). Unisys Corporation.
- V Series Systems Operations Guide, Volume 2: System Commands* (4127 4994). Unisys Corporation.
- V Series Systems Operations Guide, Volume 3: System Utilities* (4127 0000). Unisys Corporation.
- V Series Systems Operations Guide, Volume 4: System Messages and Recovery* (4127 0018). Unisys Corporation.
- V Series System Security Installation and Operations Reference Manual* (5031511). Unisys Corporation.

Bibliography

V Series VCS Implementation Reference Manual, Volume 1: Entities and Screens
(4127 0059). Unisys Corporation.

V Series VCS Implementation Reference Manual Volume 2: System Transactions
(4127 0067). Unisys Corporation.

Index

A

ABORT command (CANDE)
description, 2-4, 2-24
examples, 2-5, 2-6
parameters, 2-5
syntax, 2-4

aborting
commands and string searches, 3-22

ACCEPT statement (batch files), 4-2
example, 4-2

active editor file
automatically inserting sequence
numbers into the, 2-70
closing, 2-56
definition, 1-1
deleting lines in the, 2-88
displaying pages of the, 2-46
displaying the last page, 2-36
displaying the previous page, 2-51
flagging modified lines in the, 2-49
removing, 2-57
replacing with recall file, 2-55
saving, 2-67

ADM command (CANDE)
description, 2-5
details, 2-6
parameters, 2-6
syntax, 2-5

ADOPT option, 2-35

ADS files, comment indicator for, 2-81

ADSINP files, text and sequence number
information for, 2-40

ALL, use in a sequence range, 2-4

alternate name, specifying for the
CANFMT file, 6-3

ampersand (&), use in batch files, 4-9

APPEND command (CANDE)
description, 2-6
examples, 2-7
parameters, 2-7
syntax, 2-6

applications
adding CANDE and the View File
utility as applications with
VCS, B-1

assigning, an internal password, 2-31

asterisk (*)
use in a file title, 2-2
use in batch files, 4-9

AT command (VF utility)
description, 3-2
examples, 3-2
parameters, 3-2
syntax, 3-2

at sign (@)
use in a sequence range, 2-3

AT statement (batch files), 4-2
example, 4-2

attributes
displaying for your workstation, 3-21
establishing for your
workstation, 3-21

automatic display mode, activating, 2-5

automatic execution of batch files, 4-13

B

BACK command (CANDE)
description, 2-7
details, 2-8
examples, 2-8
parameters, 2-8
syntax, 2-7

backup family
default
label for batch files, 4-13
primary
label for batch files, 4-13
secondary
label for batch files, 4-13

batch files
automatic execution of, 4-13
contents, 4-1

Index

- batch files (cont.)
 - description, 4-1
 - executing, 4-11
 - executing specific jobs within, 4-6
 - including comments in, 4-9
 - including miscellaneous text in, 4-10
 - labels for, 4-13
 - preventing suspension of, 4-12
 - temporarily suspending the processing of, 4-2
 - use of BEGINJOB statement in, 4-1
 - use of ENDJOB statement in, 4-1
 - use of environmental control command (ECC) flag character in, 4-11
 - use of LABELBEGIN keyword in, 4-1
 - use of LABELBEGIN statement in, 4-6
 - use of LABELEND keyword in, 4-1, 4-6
 - use of transaction initiator character in, 4-11
 - using MCP control instructions in, 4-7
 - using multifile identifiers in, 4-8
 - using printer backup file references in, 4-9
 - using pseudo card reader (PCR) commands in, 4-10
 - using system commands in, 4-7
 - using variables in, 4-7
 - BEGIN statement (batch files), 4-3
 - example, 4-3
 - BEGINJOB statement (batch files)
 - description, 4-3
 - examples, 4-3
 - usage, 4-1
 - BINDER files
 - comment indicator for, 2-81
 - input file control statements for, 2-81
 - text and sequence number information for, 2-40
 - BLOCK command (CANDE)
 - description, 2-8
 - details, 2-9
 - example, 2-9
 - parameters, 2-9
 - syntax, 2-8
 - BOUND command (CANDE)
 - description, 2-9
 - details, 2-10
 - examples, 2-10
 - parameters, 2-10
 - BOUND command (CANDE) (cont.)
 - syntax, 2-9
 - BPL files
 - comment indicator for, 2-81
 - input file control statements for, 2-81
 - text and sequence number information for, 2-40
 - BYE command (CANDE)
 - description, 2-10
 - details, 2-11
 - parameters, 2-10
 - syntax, 2-10
 - BYE command (VF utility)
 - description, 3-3
 - details, 3-3
 - syntax, 3-3
- ## C
- CANDE
 - error messages generated by the, A-1
 - files (table), 1-2, 1-3
 - help facility, 1-4
 - installing, B-1
 - invoking, 1-4
 - terminating, 1-5
 - CANFMT file
 - absence of the, 6-1
 - contents of the, 6-1
 - default file title of the, 6-1
 - editing the, 6-1
 - file kind of the, 6-1
 - including custom formats in the, 6-2
 - including custom tab settings in the, 6-2
 - location of error messages in the, 6-2
 - location of help responses in the, 6-2
 - modifying error messages in the, 6-2
 - modifying help responses in the, 6-2
 - reserved sequence numbers in the, 6-2
 - specifying an alternate name for the, 6-3
 - text length of the, 6-1
 - CANUTL command (CANDE)
 - description, 2-11
 - details, 2-11
 - syntax, 2-11
 - CANUTL utility
 - executing, 5-1
 - menu displayed by the, 5-2

- CANUTL utility (cont.)
 - parameters, 5-2, 5-3
- case sensitivity, enabling in CANDE
 - commands, 2-1
- CATALOG command (CANDE)
 - description, 2-12
 - details, 2-13, 2-14
 - examples, 2-14
 - parameters, 2-13
 - syntax, 2-13
- CHANGE command (CANDE)
 - description, 2-15
 - details, 2-15
 - examples, 2-15
 - parameters, 2-15
 - syntax, 2-15
- changing the name of a file, 2-15
- CLOSE command (VF utility)
 - description, 3-3
 - syntax, 3-3
- closing
 - a file (VF utility), 3-19
 - the active editor file, 2-56
- COBOL files
 - comment indicator for, 2-81
 - input file control statements for, 2-82
 - text and sequence number information
 - for, 2-40
- column indicators
 - example, 2-39
- commands
 - aborting, 3-22
 - directing to CANDE, 2-24
 - directing to the View File utility, 2-87
 - directing to VF, 2-24
 - display the last entered, 2-56
 - displaying the last entered (VF utility), 3-17
 - enabling case-sensitivity in, 2-1
- commands, CANDE
 - ABORT
 - description, 2-4, 2-24
 - examples, 2-5, 2-6
 - parameters, 2-5
 - syntax, 2-4
 - ADM
 - description, 2-5
 - details, 2-6
 - parameters, 2-6
 - syntax, 2-5
 - commands, CANDE (cont.)
 - APPEND
 - description, 2-6
 - examples, 2-7
 - parameters, 2-7
 - syntax, 2-6
 - BACK
 - description, 2-7
 - details, 2-8
 - examples, 2-8
 - parameters, 2-8
 - syntax, 2-7
 - BLOCK
 - description, 2-8
 - details, 2-9
 - example, 2-9
 - parameters, 2-9
 - syntax, 2-8
 - BOUND
 - description, 2-9
 - details, 2-10
 - examples, 2-10
 - parameters, 2-10
 - syntax, 2-9
 - BYE
 - description, 2-10
 - details, 2-11
 - parameters, 2-10
 - syntax, 2-10
 - CANUTL
 - description, 2-11
 - details, 2-11
 - syntax, 2-11
 - CATALOG
 - description, 2-12
 - details, 2-13, 2-14
 - examples, 2-14
 - parameters, 2-13
 - syntax, 2-13
 - CHANGE
 - description, 2-15
 - details, 2-15
 - examples, 2-15
 - parameters, 2-15
 - syntax, 2-15
 - COMMENT
 - description, 2-16
 - details, 2-16
 - syntax, 2-16
 - COMPILE
 - description, 2-16
 - details, 2-17

commands, CANDE (cont.)

COMPILE (cont.)

- parameters, 2-16
- syntax, 2-16

COPY

- description, 2-17
- details, 2-18
- examples, 2-18
- parameters, 2-17, 2-18
- syntax, 2-17

CUT

- description, 2-19
- parameters, 2-19
- syntax, 2-19

DEBUG

- description, 2-19
- parameters, 2-19
- syntax, 2-19

DELETE

- description, 2-20
- details, 2-20
- parameters, 2-20
- syntax, 2-20

DETANK

- description, 2-20
- details, 2-21
- syntax, 2-21

DO

- description, 2-21
- details, 2-22
- example, 4-15
- parameters, 2-21, 2-22
- repeating, 4-12
- syntax, 2-21

DUPLICATE

- description, 2-22
- parameters, 2-23
- syntax, 2-23

EMODE

- description, 2-24
- syntax, 2-24

END

- details, 2-24
- syntax, 2-24

FAMILY

- description, 2-24
- details, 2-25
- examples, 2-26
- parameters, 2-25
- syntax, 2-25

FILES

- description, 2-26

commands, CANDE (cont.)

FILES (cont.)

- examples, 2-26
- parameters, 2-26
- syntax, 2-26

FIND

- description, 2-27
- details, 2-29
- parameters, 2-27, 2-28
- syntax, 2-27

FIX

- description, 2-29
- details, 2-31
- parameters, 2-29, 2-30
- syntax, 2-29

FPW

- description, 2-31
- examples, 2-31
- parameters, 2-31
- syntax, 2-31

GET

- description, 2-32
- details, 2-33
- examples, 2-33
- parameters, 2-32
- syntax, 2-32

HARDCOPY

- description, 2-33
- details, 2-34
- parameters, 2-34
- syntax, 2-33

HELP

- description, 2-34
- details, 2-35
- syntax, 2-34

JMGR

- description, 2-35
- parameters, 2-34, 2-35
- syntax, 2-35

LAST

- description, 2-36
- syntax, 2-36

LESN

- description, 2-36
- details, 2-36
- example, 2-37
- parameters, 2-36
- syntax, 2-36

LIST

- description, 2-37
- details, 2-38
- parameters, 2-37

- commands, CANDE (cont.)
- LIST (cont.)
 - syntax, 2-37
 - MAKE
 - description, 2-39
 - parameters, 2-39, 2-40
 - syntax, 2-39
 - MARK
 - description, 2-42
 - details, 2-43
 - examples, 2-43
 - parameters, 2-42
 - syntax, 2-42, 2-43
 - MERGE
 - description, 2-43
 - details, 2-44
 - parameters, 2-43
 - MOVE
 - description, 2-44
 - parameters, 2-44
 - syntax, 2-44
 - NEXT
 - description, 2-45
 - details, 2-46
 - examples, 2-46
 - parameters, 2-45, 2-46
 - syntax, 2-45
 - use with LIST command, 2-38
 - PAGE
 - description, 2-46
 - examples, 2-47
 - parameters, 2-47
 - syntax, 2-47
 - PASTE
 - description, 2-48
 - examples, 2-49
 - parameters, 2-48
 - syntax, 2-48
 - PATCHID
 - description, 2-49
 - details, 2-50
 - examples, 2-50
 - parameters, 2-49
 - syntax, 2-49
 - PCR
 - description, 2-50
 - details, 2-51
 - parameters, 2-50, 2-51
 - syntax, 2-50
 - PLOG
 - description, 2-51
 - details, 2-51
- commands, CANDE (cont.)
- PLOG (cont.)
 - syntax, 2-51
 - PREVIOUS
 - description, 2-51
 - details, 2-52
 - examples, 2-52
 - parameters, 2-52
 - syntax, 2-52
 - PSCREEN
 - description, 2-53
 - details, 2-53
 - examples, 2-53
 - syntax, 2-53
 - PURGE
 - description, 2-53
 - details, 2-54
 - parameters, 2-54
 - syntax, 2-54
 - QUIT
 - description, 2-54
 - parameters, 2-54
 - syntax, 2-54
 - REBUILD
 - description, 2-55
 - details, 2-55
 - parameters, 2-55
 - syntax, 2-55
 - RECALL
 - description, 2-55
 - details, 2-56
 - parameters, 2-56
 - syntax, 2-56
 - REDO
 - description, 2-56
 - details, 2-56
 - syntax, 2-56
 - RELEASE
 - description, 2-56
 - details, 2-57
 - parameters, 2-57
 - syntax, 2-57
 - REMOVE
 - description, 2-57
 - parameters, 2-58
 - syntax, 2-58
 - RENAME (*See also* CHANGE command), 2-15
 - REORGANIZE
 - description, 2-58
 - details, 2-58
 - parameters, 2-58

commands, CANDE (cont.)

REORGANIZE (cont.)

syntax, 2-58

REPLACE

description, 2-59

details, 2-60

examples, 2-61

parameters, 2-59, 2-60

syntax, 2-59

RESEQUENCE

description, 2-61

details, 2-62

example, 2-62, 2-61

syntax, 2-61

RLOG

description, 2-62

details, 2-64

example, 2-64

parameters, 2-63

record types displayed by
the, 2-64

syntax, 2-62

RMERGE

description, 2-65

details, 2-65

parameters, 2-65

syntax, 2-65

RUN

description, 2-65

details, 2-66

example, 2-66

parameters, 2-65

syntax, 2-65

SAME

description, 2-66

details, 2-67

parameters, 2-66

syntax, 2-66

SAVE

description, 2-67

details, 2-67

parameters, 2-67

syntax, 2-67

SEARCH

description, 2-67

details, 2-68

parameters, 2-68

syntax, 2-68

SECURITY

description, 2-69

details, 2-70

parameters, 2-69

commands, CANDE (cont.)

SECURITY (cont.)

syntax, 2-69

SEQUENCE

description, 2-70

details, 2-71

example, 2-71

parameters, 2-70

syntax, 2-70

SESSION

description, 2-72, 2-76

parameters, 2-73, 2-74, 2-75

syntax, 2-72

SLOG

details, 2-78

example, 2-78

parameters, 2-77, 2-78

syntax, 2-77

START

description, 2-79

details, 2-79

parameters, 2-79

syntax, 2-79

START (*See also* COMPILE
command), 2-16

TABS

description, 2-83

details, 2-84

parameters, 2-83

syntax, 2-83

TANK

description, 2-85

details, 2-85

syntax, 2-85

terminating, 2-20

TEACH (*See also* HELP command), 2-34

TYPE

description, 2-85

details, 2-86

parameters, 2-86

syntax, 2-86

VIEW

description, 2-86

details, 2-87

parameters, 2-87

syntax, 2-86

VMODE

description, 2-87

details, 2-87

syntax, 2-87

VOID

description, 2-88

- commands, CANDE (cont.)
 VOID (cont.)
 details, 2-88
 parameters, 2-88
 syntax, 2-88
 WHATS
 description, 2-88
 details, 2-89
 examples, 2-89
 parameters, 2-88
 screen displayed by, 2-89
 syntax, 2-88
 WORKSTATION
 description, 2-90
 parameters, 2-92, 2-93, 2-94
 syntax, 2-91
 - <nn>
 description, 2-95
 parameters, 2-95
 syntax, 2-95
 + <nn>
 description, 2-95
 parameters, 2-95
 syntax, 2-95
- commands, system
 using in batch files, 4-7
- commands, VF utility
 AT
 description, 3-2
 examples, 3-2
 parameters, 3-2
 syntax, 3-2
 BYE
 description, 3-3
 details, 3-3
 syntax, 3-3
 CLOSE
 description, 3-3
 syntax, 3-3
 END (*See also* BYE command), 3-3
 FORM
 description, 3-3
 syntax, 3-4
 FORMAT
 description, 3-4
 details, 3-8
 examples, 3-8
 parameters, 3-5, 3-6, 3-7
 syntax, 3-4, 3-5
 HELP
 description, 3-9
 example, 3-9
- commands, VF utility (cont.)
 HELP (cont.)
 syntax, 3-9
 LIST
 details, 3-12
 parameters, 3-10, 3-11
 syntax, 3-9, 3-10
 NEXT
 description, 3-13
 parameters, 3-13, 3-14
 syntax, 3-13
 OUTPUT
 description, 3-14
 parameters, 3-15, 3-16, 3-17
 syntax, 3-14, 3-17
 REDO
 description, 3-17
 syntax, 3-17
 REMOVE
 description, 3-18
 syntax, 3-18
 SAVE
 description, 3-18
 syntax, 3-18
 SESSION
 description, 3-18
 sample screen response from, 3-18
 syntax, 3-18
 VIEW
 description, 3-19
 details, 3-20
 parameters, 3-20
 syntax, 3-19
 WHAT
 description, 3-20
 parameters, 3-21
 syntax, 3-21
 WORKSTATION
 description, 3-21
 parameters, 3-22
 syntax, 3-21
 ?BRK
 description, 3-22
 syntax, 3-22
 - <nn>
 description, 3-22
 parameters, 3-23
 syntax, 3-23
 + <nn>
 description, 3-23
 parameters, 3-23
 syntax, 3-23

Index

- comment characters for batch files, 4-9
- COMMENT command (CANDE)
 - description, 2-16
 - details, 2-16
 - syntax, 2-16
- comment indicators for various file kinds,
 - (table) 2-81
- comments
 - including in input files, 7-5
 - indicator location for, 7-5
- COMPILE command (CANDE)
 - description, 2-16
 - details, 2-17
 - parameters, 2-16
 - syntax, 2-16
- compiler control records,
 - file-equated, 2-80
- compiling an editor file, 2-16
- COMPTS format, 2-80
- control instructions, MCP
 - using in batch files, 4-7
- COPY command (CANDE)
 - description, 2-17
 - details, 2-18
 - examples, 2-18
 - parameters, 2-17, 2-18
 - syntax, 2-17
- copying
 - the screen display into the job log file, 2-53
- corrupted editor file
 - automatic recovery of, 1-5
- creating
 - a pseudo card reader file, 2-50
 - an editor file, 2-39
- CUT command (CANDE)
 - description, 2-19
 - parameters, 2-19
 - syntax, 2-19
- D**
- DASDL files
 - comment indicator for, 2-81
 - input file control statements for, 2-82
 - text and sequence number information for, 2-40
- data files
 - comment indicator for, 2-81
 - format of, 2-41
- data files (cont.)
 - maximum records supported by CANDE, 1-1
 - text and sequence number information for, 2-40
- date and time stamp
 - inserting in the PATMRG output file, 7-4
- DEBUG command (CANDE)
 - description, 2-19
 - parameters, 2-19
 - syntax, 2-19
- default
 - family label for batch files, 4-13
- DELETE command (CANDE)
 - description, 2-20
 - details, 2-20
 - parameters, 2-20
 - syntax, 2-20
- deleting
 - lines in the editor file, 2-88
- DETANK command (CANDE)
 - description, 2-20
 - details, 2-21
 - syntax, 2-21
- directing commands to the View File utility, 2-87
- DISPLAY statement (batch files), 4-3
 - example, 4-4
- displaying
 - a file at your workstation, 3-9
 - information about files, 2-12
 - pages of the active editor file, 2-46
 - the current attributes of your terminal, 2-90
 - the editor file in backward paging mode, 2-7
 - the last CANDE command entered, 2-56, 2-36
 - the last VF command entered, 3-17
 - the next page of a file you are viewing, 3-13
 - the next page of the active editor file, 2-45
 - the previous page of the active editor file, 2-51
 - the run log, 2-62
 - the same page again, 2-66
 - the status of files, 2-88
- DO command (CANDE)
 - description, 2-21
 - details, 2-22

DO command (CANDE) (cont.)

- example, 4-15
- parameters, 2-21, 2-22
- repeating, 4-12
- syntax, 2-21

dump, CANDE, 1-5

DUPLICATE command (CANDE)

- description, 2-22
- parameters, 2-23
- syntax, 2-23

E

ECC (environmental control command),

- flag character, 4-11

editor file

- automatic recovery, 1-5

editor file, active

- automatically inserting sequence numbers into the, 2-70

closing, 2-56

definition, 1-1

deleting lines in the, 2-88

displaying pages of the, 2-46

displaying the last page, 2-36

displaying the previous page, 2-51

flagging modified lines in the, 2-49

removing, 2-57

replacing with recall file, 2-55

saving, 2-67

editor files

compiling, 2-16

creating, 2-39

listing, 2-37

removing, 2-57

ELSE statement (batch files), 4-4

EMODE command (CANDE)

description, 2-24

syntax, 2-24

END

- use in a sequence range, 2-4

END command (CANDE)

details, 2-24

syntax, 2-24

END command (VF utility) (*See also* BYE command), 3-3

END statement (batch files), 4-4

ending

- a CANDE session, 1-5

ENDJOB statement (batch files), 4-4

- usage, 4-1

environmental control command (ECC)

- flag character, 4-11

equal sign (=), use in a sequence

- range, 2-4

error messages

- generated by CANDE, A-1

- generated by the PATMRG

- utility, 7-6

- generated by the View File

- utility, A-1

error messages, location in the CANFMT

- file, 6-2

examples

- of a batch file, 4-14

- of column indicators, 2-39

- of comment indicators in a batch

- file, 4-10

- of file titles, 2-2

- of input file control statements

- for BINDER, 2-81

- for BPL, 2-81

- for COBOL, 2-82

- for DASDL, 2-82

- for FORTRAN, 2-82

- for PASCAL, 2-82

- for RPG, 2-83

- for WFL, 2-83

- of sequence ranges, 2-4

- of the ABORT command, 2-5, 2-6

- of the ACCEPT statement (batch files), 4-2

- of the APPEND command, 2-7

- of the AT command (VF utility), 3-2

- of the AT statement (batch files), 4-2

- of the BACK command, 2-8

- of the BEGIN statement (batch files), 4-3

- of the BEGINJOB statement (batch files), 4-3

- of the BLOCK command, 2-9

- of the BOUND command, 2-10

- of the CATALOG command, 2-14

- of the CHANGE command, 2-15

- of the COPY command, 2-18

- of the DISPLAY statement (batch files), 4-4

- of the DO command (CANDE), 4-15

- of the FAMILY command, 2-26

- of the FILES command, 2-26

- of the FORMAT command (VF utility), 3-8

- of the FPW command, 2-31

examples (cont.)

- of the GET command, 2-33
 - of the HELP command (VF utility), 3-9
 - of the LESN command, 2-37
 - of the MARK command, 2-43
 - of the NEXT command, 2-46
 - of the PAGE command, 2-47
 - of the PASTE command, 2-49
 - of the PATCHID command, 2-50
 - of the PREVIOUS command, 2-52
 - of the PSCREEN command, 2-53
 - of the REPLACE command, 2-61
 - of the RESEQUENCE command, 2-62
 - of the RLOG command, 2-64
 - of the RUN command, 2-66
 - of the SEQUENCE command, 2-71
 - of the SLOG command, 2-78
 - of the WHATS command, 2-89
 - of using multifile identifiers in a batch file, 4-8
 - of using variables in a batch file, 4-7
- executing
- batch files
 - automatically, 4-13
 - with the DO command, 2-21, 4-12
 - CANDE, 1-4
 - programs, 2-65
 - specific jobs within a batch file, 4-6
 - the CANUTL utility, 5-1
 - the VF utility, 2-86
 - the View File utility, 3-1

F

family

- default label for batch files, 4-13
- primary backup label for batch files, 4-13
- secondary backup label for batch files, 4-13

FAMILY command (CANDE)

- description, 2-24
- details, 2-25
- examples, 2-26
- parameters, 2-25
- syntax, 2-25

family name

- use in a file title, 2-2

- file equating the PATMST output file, 7-3

file kind

- changing for an editor file, 2-17
- changing the, 2-85
- for input files to PATMRG utility, 7-2
- numeric values for CANUTL utility, 5-2
- of the CANFMT file, 6-1

file kinds

- comment indicators for various (table), 2-81
- for input files specified in a SCRIPT file, 7-4
- supported by CANDE, 1-1
- valid for CANDE, (table) 2-40

FILE option

- using in a SCRIPT file, 7-3

file titles

- examples, 2-2
- specifying in the SCRIPT file, 7-4
- syntax for, 2-1
- use of asterisks in, 2-2
- use of family names in, 2-2
- use of file names in, 2-2
- use of numerical user IDs in, 2-2
- use of usercodes in, 2-2
- use of VCS installation numbers in, 2-2

- file-equated compiler control records, format for, 2-80

file-equated execution function

- statements used to initiate various compilers with the, 2-81

- file-naming conventions, 2-2

files

active editor

- definition, 1-1

- active editor, definition, 2-1

batch

- automatic execution of, 4-13

- contents, 4-1

- description, 4-1

- executing, 2-21, 4-11

- executing specific jobs within, 4-6

- including comments in, 4-9

- including miscellaneous text in, 4-10

- labels for, 4-13

- preventing suspension of, 4-12

- temporarily suspending the

- processing of, 4-2

- use of BEGINJOB statement in, 4-1

files (cont.)

batch (cont.)

- use of ENDJOB statement in, 4-1
- use of LABELBEGIN keyword in, 4-1
- use of LABELBEGIN statement in, 4-6
- use of LABELEND keyword in, 4-1, 4-6
- use of transaction initiator characterin, 4-11
- using MCP control instructions in, 4-7
- using multifile identifiers in, 4-8
- using printer backup file references in, 4-9
- using pseudo card reader (PCR) commands in, 4-10
- using system commands in, 4-7
- using variables in, 4-7

CANFMT

- absence of, 6-1
- contents of, 6-1
- default file title of, 6-1
- editing, 6-1
- file kind of, 6-1
- including custom formats in, 6-2
- including custom tab settings in, 6-2
- location of error messages in the, 6-2
- location of help responses in the, 6-2
- modifying error messages in, 6-2
- modifying help responses in, 6-2
- reserved sequence numbers in the, 6-2
- specifying an alternate name for, 6-3
- text length of, 6-1

closing with the VF utility, 3-19

data

- definition, 1-1
- displaying at your workstation, 3-9
- displaying the next page with the View File utility, 3-13
- displaying the status of, 2-88
- format of data, 2-41
- format of RPG, 2-41
- format of sequential, 2-41
- identifier example, 4-8
- information, displaying, 2-12

files (cont.)

- listing, 2-26
 - MSTRIN, 7-2
 - on CANDE release tape, (table) 1-2, 1-3
 - opening, 2-32
 - opening with the VF utility, 3-19
 - PATMST, 7-2
 - generating a printer listing of, 7-3
 - printer backup
 - closing, 3-3
 - closing and saving, 3-18
 - sending data to, 3-14
 - PRINTERFILE, 7-2
 - pseudo card reader, creating, 2-50
 - PTCHIN, 7-2
 - purging (VF utility), 3-18
 - recall, 2-32
 - creation, 1-1
 - definition, 1-1
 - replacing the active editor file with the, 2-55
 - status when active editor file is closed, 2-57
 - updating of, 1-1
 - renaming, 2-15
 - saving (VF utility), 3-18
 - SCRIPT, 7-2
 - description, 7-2
 - including options in, 7-2
 - using the FILE option in, 7-3
 - using the LIST option in, 7-3
 - using the LISTP option in, 7-3
 - using the - option in, 7-3
 - using the - SEQUENCE RANGE option in, 7-3
 - using the NODUP option in, 7-3
 - using the PAGEP option in, 7-3
 - using the PATCHID option in, 7-3
 - using the SUMMARY option in, 7-4
 - sequential
 - definition, 1-1
 - source file format for
 - compilation, 2-80, 2-80
 - used and generated by the PATMRG utility, 7-2
- FILES command (CANDE)
- description, 2-26
 - examples, 2-26
 - parameters, 2-26
 - syntax, 2-26

Index

- FIND command (CANDE)
 - description, 2-27
 - details, 2-29
 - parameters, 2-27, 2-28
 - syntax, 2-27
- FIX command (CANDE)
 - description, 2-29
 - details, 2-31
 - parameters, 2-29, 2-30
 - syntax, 2-29
- flagging modified lines, 2-49
- FORM command (VF utility)
 - description, 3-3
 - syntax, 3-4
- format
 - COMPTS, 2-80
 - for file-equated compiler control
 - records, 2-80
 - of source file for compilation, 2-80
- FORMAT command (VF utility)
 - description, 3-4
 - details, 3-8
 - examples, 3-8
 - parameters, 3-5, 3-6, 3-7
 - syntax, 3-4, 3-5
- format file
 - supplied with CANDE, 6-1
- formats
 - including in the CANFMT file, 6-2
- forms mode
 - turning on and off, 3-3
- FORTRAN files
 - comment indicator for, 2-81
 - input file control statements for, 2-82
 - text and sequence number information
 - for, 2-40
- FPW command (CANDE)
 - description, 2-31
 - examples, 2-31
 - parameters, 2-31
 - syntax, 2-31
- G**
- GEMCOS
 - TCL declarations for CANDE, B-1
 - TCL declarations for the View File utility, B-1
- GET command (CANDE)
 - description, 2-32
 - details, 2-33
- GET command (CANDE) (cont.)
 - examples, 2-33
 - parameters, 2-32
 - syntax, 2-32
- greater-than sign (>), use in a sequence
 - range, 2-3
- H**
- HARDCOPY command (CANDE)
 - description, 2-33
 - details, 2-34
 - parameters, 2-34
 - syntax, 2-33
- HELP command (CANDE)
 - description, 2-34
 - details, 2-35
 - syntax, 2-34
- HELP command (VF utility)
 - description, 3-9
 - example, 3-9
 - syntax, 3-9
- help facility, 1-4
- help responses
 - location in the CANFMT file, 6-2
 - modifying in the CANFMT file, 6-2
- hyphen (-), use in a sequence
 - range, 2-3
- I**
- identifiers, multifile, using in batch files, 4-8
- IF statement (batch files), 4-4
- input files
 - including comments in, 7-5
 - specifying in a SCRIPT file, 7-4
- inserting cut text into the editor
 - file, 2-48
- installation number, use in a file
 - title, 2-2
- installing CANDE, B-1
- invoking, CANDE, 1-4
- IPS files, text and sequence number information for, 2-40

J

JMGR command (CANDE)
description, 2-35
parameters, 2-34, 2-35
syntax, 2-35
job log
preparing for printing, 2-51
writing messages to the, 2-33

L

LABELBEGIN keyword, use in a batch
file, 4-1, 4-6
LABELEND keyword, use in a batch
file, 4-1, 4-6
labels
for batch files, 4-13
LAST command (CANDE)
description, 2-36
syntax, 2-36
LESN command (CANDE)
description, 2-36
details, 2-36
example, 2-37
parameters, 2-36
syntax, 2-36
less-than sign (<)
use in a sequence range, 2-4
LIST command (CANDE)
description, 2-37
details, 2-38
parameters, 2-37
syntax, 2-37
LIST command (VF utility)
details, 3-12
parameters, 3-10, 3-11
syntax, 3-9, 3-10
LIST option
using in a SCRIPT file, 7-3
listing
a file, 3-9
an editor file, 2-37
files, 2-26
LISTP option
using in a SCRIPT file, 7-3

M

MAKE command (CANDE)
description, 2-39
parameters, 2-39, 2-40
syntax, 2-39
MARK command (CANDE)
description, 2-42
details, 2-43
examples, 2-43
parameters, 2-42
syntax, 2-42, 2-43
Master Control Program (MCP)
using control instructions in batch
files, 4-7
menu
displayed by the PATMRG
utility, 7-5
menu for the CANUTL utility, 2-11, 5-2
MERGE command (CANDE)
description, 2-43
details, 2-44
parameters, 2-43
messages
displaying those in the message
tank, 2-20
emphasis types for, 4-3
error
generated by CANDE, A-1
generated by the PATMRG
utility, 7-6
generated by the View File
utility, A-1
location in the CANFMT file, 6-2
modifying in the CANFMT file, 6-2
purging, 2-53
resetting the counter, 2-21
stopping the display to your
station, 2-85
writing to the CANDE job log, 2-33
mode override character, 2-24
modified lines
flagging, 2-49
MOVE command (CANDE)
description, 2-44
parameters, 2-44
syntax, 2-44
multifile identifiers
example, 4-8
using in batch files, 4-8

N

- name, specifying an alternate for the
CANFMT file, 6-3
- naming conventions for file names, 2-2
- NEXT command (CANDE)
 - description, 2-45
 - details, 2-46
 - examples, 2-46
 - parameters, 2-45, 2-46
 - syntax, 2-45
- NEXT command (VF utility)
 - description, 3-13
 - parameters, 3-13, 3-14
 - syntax, 3-13
- NODUP option, using in a SCRIPT
file, 7-3
- nonusercoded file, example, 1-4
- numerical user ID use in a file title, 2-2

O

- ODT log
 - displaying and printing, 2-76
- ODT Log
 - searching, 2-78
- opening
 - a file (CANDE), 2-32
 - a file (VF utility), 3-19
- operating parameters
 - establishing for a session, 2-72
- options
 - FILE
 - using in a SCRIPT file, 7-3
 - LIST
 - using in a SCRIPT file, 7-3
 - LISTP
 - using in a SCRIPT file, 7-3
 - NODUP
 - using in a SCRIPT file, 7-3
 - PAGEP
 - using in a SCRIPT file, 7-3
 - patch ID
 - using in a SCRIPT file, 7-3
 - SUMMARY
 - using in a SCRIPT file, 7-4
 - - using in a SCRIPT file, 7-3
 - SEQUENCE RANGE
 - using in a SCRIPT file, 7-3

- OUTPUT command (VF utility)
 - description, 3-14
 - parameters, 3-15, 3-16, 3-17
 - syntax, 3-14, 3-17

P

- page
 - redisplaying the same, 2-66
- page break
 - inserting after printout of each
PATMRG input file, 7-3
- PAGE command (CANDE)
 - description, 2-46
 - examples, 2-47
 - parameters, 2-47
 - syntax, 2-47
- page, displaying the next, 2-45
- PAGEP option
 - using in a SCRIPT file, 7-3
- parameters
 - for the CANUTL utility, 5-2, 5-3
- Pascal files
 - comment indicator for, 2-81
- PASCAL files
 - input file control statements for, 2-82
- Pascal files
 - text and sequence number information
for, 2-40
- password, internal
 - assigning, 2-31
 - removing, 2-31
- PASTE command (CANDE)
 - description, 2-48
 - examples, 2-49
 - parameters, 2-48
 - syntax, 2-48
- patch ID
 - inserting into PATMRG output
files, 7-3
- PATCHID command (CANDE)
 - description, 2-49
 - details, 2-50
 - examples, 2-50
 - parameters, 2-49
 - syntax, 2-49
- PATCHID option
 - using in a SCRIPT file, 7-3
- PATMRG utility
 - error messages generated by the, 7-6
 - features, 7-1

- PATMRG utility (cont.)
 - files used and generated by the, 7-2
 - guidelines for using the, 7-1
 - menu displayed by the, 7-5
 - single-pass execution of, 7-2
 - use of SCRIPT file with the, 7-2
 - PATMST file
 - file equating, 7-3
 - generating a printer listing of the, 7-3
 - PCR command (CANDE)
 - description, 2-50
 - details, 2-51
 - parameters, 2-50, 2-51
 - syntax, 2-50
 - percent sign (%), use in batch files, 4-9
 - PLOG command (CANDE)
 - description, 2-51
 - details, 2-51
 - syntax, 2-51
 - PREVIOUS command (CANDE)
 - description, 2-51
 - details, 2-52
 - examples, 2-52
 - parameters, 2-52
 - syntax, 2-52
 - primary backup family label for batch files, 4-13
 - printer backup files
 - closing, 3-3
 - closing and saving, 3-18
 - including formats in the CANFMT file, 6-2
 - sending data to, 3-14
 - printer listing
 - generating for the PATMRG output file, 7-3
 - printer listing, generating for the PATMRG output file, 7-3
 - printing
 - files, 3-1
 - processing delays, compensating for, 4-11
 - programs
 - executing, 2-65
 - PSCREEN command (CANDE)
 - description, 2-53
 - details, 2-53
 - examples, 2-53
 - syntax, 2-53
 - pseudo card reader (PCR)
 - file creation, 2-50
 - pseudo card reader (PCR) (cont.)
 - including command in a batch file, 4-10
 - PURGE command (CANDE)
 - description, 2-53
 - details, 2-54
 - parameters, 2-54
 - syntax, 2-54
 - purging
 - a file (VF utility), 3-18
 - messages, 2-53
- ## Q
- QUIT command (CANDE)
 - description, 2-54
 - parameters, 2-54
 - syntax, 2-54
- ## R
- REBUILD command (CANDE)
 - description, 2-55
 - details, 2-55
 - parameters, 2-55
 - syntax, 2-55
 - RECALL command (CANDE)
 - description, 2-55
 - details, 2-56
 - parameters, 2-56
 - syntax, 2-56
 - recall file
 - creating, 2-32
 - creation, 1-1
 - definition, 1-1
 - replacing the active editor file with the, 2-55
 - status when active editor file is closed, 2-57
 - updating of the, 1-1, 2-32
 - record types
 - displayed by the RLOG command, 2-64
 - records
 - file-equated compiler control, 2-80
 - longer than 80 characters, editing, 2-42
 - removing from the master patch file, 7-3

- recovery
 - of corrupted editor file, 1-5
 - redisplaying
 - the same page of the editor file, 2-66
 - REDO command (CANDE)
 - description, 2-56
 - details, 2-56
 - syntax, 2-56
 - REDO command (VF utility)
 - description, 3-17
 - syntax, 3-17
 - RELEASE command (CANDE)
 - description, 2-56
 - details, 2-57
 - parameters, 2-57
 - syntax, 2-57
 - REMOVE command (CANDE)
 - description, 2-57
 - parameters, 2-58
 - syntax, 2-58
 - REMOVE command (VF utility)
 - description, 3-18
 - syntax, 3-18
 - removing
 - an editor file, 2-57
 - an internal password, 2-31
 - RENAME command (*See also* CHANGE command), 2-15
 - renaming a file, 2-15
 - REORGANIZE command (CANDE)
 - description, 2-58
 - details, 2-58
 - parameters, 2-58
 - syntax, 2-58
 - repeating the DO command, 4-12
 - REPLACE command (CANDE)
 - description, 2-59
 - details, 2-60
 - examples, 2-61
 - parameters, 2-59, 2-60
 - syntax, 2-59
 - RESEQUENCE command (CANDE)
 - description, 2-61
 - details, 2-62
 - example, 2-62
 - parameters, 2-61
 - syntax, 2-61
 - reserved sequence numbers in the CANFMT file (table), 6-2
 - RLOG command (CANDE)
 - description, 2-62
 - details, 2-64
 - RLOG command (CANDE) (cont.)
 - example, 2-64
 - parameters, 2-63
 - record types displayed by the, 2-64
 - syntax, 2-62
 - RMERGE command (CANDE)
 - description, 2-65
 - details, 2-65
 - parameters, 2-65
 - syntax, 2-65
 - RPG files
 - comment indicator for, 2-81
 - format of, 2-41
 - input file control statements for, 2-83
 - text and sequence number information for, 2-40
 - RUN command (CANDE)
 - description, 2-65
 - details, 2-66
 - example, 2-66
 - parameters, 2-65
 - syntax, 2-65
 - run log
 - displaying, 2-62
- ## S
- SAME command (CANDE)
 - description, 2-66
 - details, 2-67
 - parameters, 2-66
 - syntax, 2-66
 - SAVE command (CANDE)
 - description, 2-67
 - details, 2-67
 - parameters, 2-67
 - syntax, 2-67
 - SAVE command (VF utility)
 - description, 3-18
 - syntax, 3-18
 - saving
 - a file (VF utility), 3-18
 - the active editor file, 2-67
 - screen display
 - copying into the job log file, 2-53
 - SCRIPT file
 - description, 7-2
 - including options in the, 7-2
 - specifying file titles in the, 7-4
 - specifying input files in a, 7-4
 - using the FILE option in the, 7-3

- SCRIPT file (cont.)
 - using the LIST option in the, 7-3
 - using the LISTP option in the, 7-3
 - using the - option in the, 7-3
 - using the - SEQUENCE RANGE option in the, 7-3
 - using the NODUP option in the, 7-3
 - using the PAGEP option in the, 7-3
 - using the PATCHID option in the, 7-3
 - using the SUMMARY option in the, 7-4
 - valid file kinds for input files, 7-4
- SEARCH command (CANDE)
 - description, 2-67
 - details, 2-68
 - parameters, 2-68
 - syntax, 2-68
- secondary backup family label for batch files, 4-13
- SECURITY command (CANDE)
 - description, 2-69
 - details, 2-70
 - parameters, 2-69
 - syntax, 2-69
- SEQUENCE command (CANDE)
 - description, 2-70
 - details, 2-71
 - example, 2-71
 - parameters, 2-70
 - syntax, 2-70
- sequence numbers
 - automatic insertion of, 2-70
 - automatically generated, 1-1
 - displaying the last edited, 2-36
 - length for various file kinds, 2-40
 - position for various file kinds, 2-40
 - reserved in the CANFMT file (table), 6-2
 - treating duplicates as fatal errors in the PATMRG utility, 7-3
- sequence range
 - defining for a block, 2-8
 - examples, 2-4
 - syntax for, 2-3
 - use of a beginning line number in a, 2-3
 - use of a hyphen (-) in a, 2-3
 - use of an ending line number in a, 2-4
 - use of equal sign (=) in a, 2-4
 - use of ALL in a, 2-4
 - use of END in a, 2-4
- sequence range (cont.)
 - use of the greater-than sign (>) in a, 2-3
 - use of the less-than sign (<) in a, 2-4
- sequential files
 - format of, 2-41
 - maximum records supported by CANDE, 1-1
 - text and sequence number information for, 2-40
- session
 - CANDE
 - ending, 1-5
 - starting, 1-4
 - establishing operating parameters for a, 2-72
 - VF, displaying characteristics about a, 3-18
- SESSION command (CANDE)
 - description, 2-72, 2-76
 - parameters, 2-73, 2-74, 2-75
 - syntax, 2-72
- SESSION command (VF utility)
 - description, 3-18
 - sample screen response from, 3-18
 - syntax, 3-18
- SLOG command (CANDE)
 - details, 2-78
 - example, 2-78
 - parameters, 2-77, 2-78
 - syntax, 2-77
- source file format for compilation, 2-80
- START command (CANDE)
 - description, 2-79
 - details, 2-79
 - parameters, 2-79
 - syntax, 2-79
- START command (*See also* COMPILE command), 2-16
- starting
 - a CANDE session, 1-4
- statements, batch file
 - ACCEPT, 4-2
 - AT, 4-2
 - BEGIN, 4-3
 - BEGINJOB, 4-3
 - DISPLAY, 4-3, 4-4
 - ELSE, 4-4
 - END, 4-4
 - ENDJOB, 4-4
 - IF, 4-4

Index

status

of files, displaying the, 2-88

SUMMARY option, use in a SCRIPT file, 7-4

suspension of a batch file, preventing, 4-12

SVF (*See* system variable files)

syntax

ABORT command, 2-4
ADM command, 2-5
APPEND command, 2-6
AT command (VF utility), 3-2
BACK command, 2-7
BLOCK command, 2-8
BOUND command, 2-9
BYE command, 2-10
BYE command (VF utility), 3-3
CANUTL command, 2-11
CATALOG command, 2-13
CHANGE command, 2-15
CLOSE command (VF utility), 3-3
COMMENT command, 2-16
COMPILE command, 2-16
COPY command, 2-17
CUT command, 2-19
DEBUG command, 2-19
DELETE command, 2-20
DETANK command, 2-21
DO command, 2-21
DUPLICATE command, 2-23
EMODE command, 2-24
END command, 2-24
FAMILY command, 2-25
file name, 2-2
file title, 2-1
FILES command, 2-26
FIND command, 2-27
FIX command, 2-29
for custom tab settings in the CANFMT file, 2-84
FORM command (VF utility), 3-4
FORMAT command (VF utility), 3-4, 3-5
FPW command, 2-31
GET command, 2-32
HARDCOPY command, 2-33
HELP command, 2-34
HELP command (VF utility), 3-9
JMGR command, 2-35
LAST command, 2-36
LESN command, 2-36
LIST command, 2-37

syntax (cont.)

LIST command (VF utility), 3-9, 3-10
MAKE command, 2-39
MARK command, 2-42, 2-43
MOVE command, 2-44
NEXT command, 2-45
NEXT command (VF utility), 3-13
OUTPUT command (VF utility), 3-14, 3-17
PAGE command, 2-47
PASTE command, 2-48
PATCHID command, 2-49
PCR command, 2-50
PLOG command, 2-51
PREVIOUS command, 2-52
PSCREEN command, 2-53
PURGE command, 2-54
QUIT command, 2-54
REBUILD command, 2-55
RECALL command, 2-56
REDO command, 2-56
REDO command (VF utility), 3-17
RELEASE command, 2-57
REMOVE command, 2-58
REMOVE command (VF utility), 3-18
REORGANIZE command, 2-58
REPLACE command, 2-59
RESEQUENCE command, 2-61
RLOG command, 2-62
RMERGE command, 2-65
RUN command, 2-65
SAME command, 2-66
SAVE command, 2-67
SAVE command (VF utility), 3-18
SEARCH command, 2-68
SECURITY command, 2-69
SEQUENCE command, 2-70
sequence range, 2-3
SESSION command, 2-72
SESSION command (VF utility), 3-18
SLOG command, 2-77
START command, 2-79
TABS command, 2-83
TANK command, 2-85
TYPE command, 2-86
usercode, 2-2
VIEW command, 2-86
VIEW command (VF utility), 3-19
VMODE command, 2-87
VOID command, 2-88
WHAT command (VF utility), 3-21
WHATS command, 2-88

syntax (cont.)
 WORKSTATION command, 2-91
 WORKSTATION command (VF utility), 3-21
 ?BRK command (VF utility), 3-22
 - <nn> command, 2-95
 - <nn> command (VF utility), 3-23
 + <nn> command, 2-95
 + <nn> command (VF utility), 3-23
 system commands, using in batch files, 4-7
 system variable files, creating, 4-13

T

tab settings
 including in a format file, 2-84
 including in the CANFMT file, 6-2
 TABS command (CANDE)
 description, 2-83
 details, 2-84
 parameters, 2-83
 syntax, 2-83
 TANK command (CANDE)
 description, 2-85
 details, 2-85
 syntax, 2-85
 terminating, 2-20
 tasks
 adopting and abandoning, 2-35
 terminating, 2-4
 terminating those initiated during the current session, 2-54
 TEACH command (*See also* HELP command), 2-34
 terminals
 displaying and defining attributes of, 2-90
 terminating
 a CANDE session, 1-5, 2-10, 2-24
 tasks, 2-4
 tasks initiated during the current session, 2-54
 the command in progress, 3-22
 the TANK command, 2-20
 termination of CANDE after three dumps, 1-5
 text
 including miscellaneous in a batch file, 4-10

text (cont.)
 length for various file kinds (table), 2-40
 transaction control language (TCL)
 adding CANDE declarations with, B-1
 adding View File utility declarations with, B-1
 transaction initiator character, 4-11
 transactions
 adding CANDE and View File transactions with VCS
 TYPE command (CANDE)
 description, 2-85
 details, 2-86
 parameters, 2-86
 syntax, 2-86

U

user ID
 assignment of the, 1-3
 description, 1-3
 example, 1-4
 numerical, 2-2
 user variable files, creating, 4-13
 usercode
 requirements, 1-4
 use in a file title, 2-2
 usercoded file, example, 1-4
 utility programs
 CANUTL
 executing, 5-1
 menu, 5-2
 parameters, 5-2, 5-3
 PATMRG
 features, 7-1
 files used and generated by, 7-2
 guidelines for use, 7-1
 single-pass execution, 7-2
 use of SCRIPT file with the, 7-2
 UVF (user variable files), 4-13

V

variables, using in batch files, 4-7
 video highlights, 2-92
 VIEW command (CANDE)
 description, 2-86
 details, 2-87
 parameters, 2-87

VIEW command (CANDE) (cont.)
 syntax, 2-86

VIEW command (VF utility)
 description, 3-19
 details, 3-20
 parameters, 3-20
 syntax, 3-19

View File utility
 error messages generated by the, A-1
 executing, 2-86

View File utility, executing, 3-1

viewing
 files, 2-86, 3-1

VMODE command (CANDE)
 description, 2-87
 details, 2-87
 syntax, 2-87

VOID command (CANDE)
 description, 2-88
 details, 2-88
 parameters, 2-88
 syntax, 2-88

W

WFL files
 comment indicator for, 2-81
 input file control statements for, 2-83
 text and sequence number information
 for, 2-40

WHAT command (VF utility)
 description, 3-20
 parameters, 3-21
 syntax, 3-21

WHATS command (CANDE)
 description, 2-88
 details, 2-89
 examples, 2-89
 parameters, 2-88
 screen displayed by the, 2-89
 syntax, 2-88

WORKSTATION command (CANDE)
 description, 2-90
 parameters, 2-92, 2-93, 2-94
 syntax, 2-91

WORKSTATION command (VF utility)
 description, 3-21
 parameters, 3-22

WORKSTATION command (VF
utility) (cont.)
 syntax, 3-21

Special Characters

*

 use in a file title, 2-2
 use in batch files, 4-9

&

 use in batch files, 4-9

%

 use in batch files, 4-9

=

 use in a sequence range, 2-4

- <nn> command (CANDE)
 description, 2-95
 parameters, 2-95
 syntax, 2-95

- <nn> command (VF utility)
 description, 3-22
 parameters, 3-23
 syntax, 3-23

- option
 using in a SCRIPT file, 7-3

- SEQUENCE RANGE option
 using in a SCRIPT file, 7-3

+ <nn> command (CANDE)
 description, 2-95
 parameters, 2-95
 syntax, 2-95

+ <nn> command (VF utility)
 description, 3-23
 parameters, 3-23
 syntax, 3-23

>

 use in a sequence range, 2-3

>@

 use in a sequence range, 2-3

><

 use in a sequence range, 2-4

?BRK command (VF utility)
 description, 3-22
 syntax, 3-22

!DEFFAM label for batch files, 4-13

!PRIBKP label for batch files, 4-13

!SECBKP label for batch files, 4-13

Help Us To Help You

Publication Title _____

Form Number _____

Date _____

Unisys Corporation is interested in your comments and suggestions regarding this manual. We will use them to improve the quality of your Product Information. Please check type of suggestion:

Addition

Deletion

Revision

Error

Comments _____

Name _____

Title _____

Company _____

Address (Street, City, State, Zip) _____

Telephone Number _____

Help Us To Help You

Publication Title _____

Form Number _____

Date _____

Unisys Corporation is interested in your comments and suggestions regarding this manual. We will use them to improve the quality of your Product Information. Please check type of suggestion:

Addition

Deletion

Revision

Error

Comments _____

Name _____

Title _____

Company _____

Address (Street, City, State, Zip) _____

Telephone Number _____

Help Us To Help You

Publication Title _____

Form Number _____

Date _____

Unisys Corporation is interested in your comments and suggestions regarding this manual. We will use them to improve the quality of your Product Information. Please check type of suggestion:

Addition

Deletion

Revision

Error

Comments _____

Name _____

Title _____

Company _____

Address (Street, City, State, Zip) _____

Telephone Number _____



No Postage
necessary
if mailed in the
United States

BUSINESS REPLY MAIL

First Class Permit No. 817 Detroit, MI 48232

Postage Will Be Paid by Addressee

Unisys Corporation
ATTN: Product Information
460 Sierra Madre Villa
Pasadena, CA 91107-9853 USA



No Postage
necessary
if mailed in the
United States

BUSINESS REPLY MAIL

First Class Permit No. 817 Detroit, MI 48232

Postage Will Be Paid by Addressee

Unisys Corporation
ATTN: Product Information
460 Sierra Madre Villa
Pasadena, CA 91107-9853 USA



No Postage
necessary
if mailed in the
United States

BUSINESS REPLY MAIL

First Class Permit No. 817 Detroit, MI 48232

Postage Will Be Paid by Addressee

Unisys Corporation
ATTN: Product Information
460 Sierra Madre Villa
Pasadena, CA 91107-9853 USA

