

# SWAP V5.0

## OpenVMS Software Maintenance Tool

### Quick Links

<a href="#">↶</a>	New Features of Version 4	4
<a href="#">↶</a>	Controlling SWAP	16
<a href="#">↶</a>	Commands	20
<a href="#">↶</a>	Menu	22
<a href="#">↶</a>	Qualifier Reference	28
<a href="#">↶</a>	Error Messages	39
<a href="#">↶</a>	ASCII Table	45

# Saiga Systems Software

#215, 801 - 6th Street SW  
Calgary, Alberta, Canada  
T2P 3V8

Voice (sales and general calls): 1-800-561-8876  
Technical Support: 1-866-573-8182

Fax	(403) 263-0744
Telephone	(403) 263-1151 (sales and general calls) (403) 248-2004 (technical support)
Internet	sales@saiga.com support@saiga.com
WWW	<a href="http://www.saiga.com/">http://www.saiga.com/</a>
Listserver	cohort@saiga.com

October 2004

The information in this document is subject to change without notice and should not be construed as a commitment by Saiga Systems Inc. Saiga Systems Inc. assumes no responsibility for any errors or omissions that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

Copyright © 2004 by Saiga Systems Inc.  
All rights reserved

DECNET, VAX, OpenVMS, Alpha AXP and VAXCLUSTER are trademarks of:  
Compaq

## Table of Contents

New Features in SWAP Version .....	4
Overview .....	7
SWAP Usage Summary .....	8
Getting Started .....	10
Introduction to SWAP .....	11
Specifying Files .....	12
Controlling the Search .....	14
Controlling Output .....	16
Removing Tabs .....	16
Truncating Records .....	17
Limits .....	18
Logicals .....	19
SWAP Commands .....	20
MENU .....	22
SWAP Main Menu .....	22
Search & Replace Menu .....	22
Modify Setup Menu .....	23
Information Screen .....	23
View File .....	24
Using Variables to Reformat Files .....	25
Qualifiers .....	28
Appendix A - Error Messages .....	39
Appendix B - ASCII Table .....	44
Command Summary .....	47
Index .....	48

## New Features in SWAP Version 5

1. Swap now includes a variable that allows you to break a single line into multiple lines while reformatting using variables. Insert %/% into the replacement string anyplace you would like the line split into two lines. You can insert as many splits into a line as you wish.
2. It is possible to insert a variable in the replacement string more than once. This feature was not previously documented though it worked in some cases. With V5 it has been extensively tested so it now works consistently.
3. In all previous versions of Swap on sequential files with carriage return carriage control could be opened and updated. V5 now supports two additional files formats; stream and stream\_lf. These two file formats are commonly found in files transferred to the VAX or AXP system from a PC or saved on a file share existing on the VAX or AXP.
4. Swap now supports specifying up to 3 search and replace string pairs in a single command. The new DCL command line format is:

**\$ Swap [/qualifiers] file.type string1 replace1 [string2 replace2] [string3 replace3]**

the second and third pairs of search and replace strings are optional.

5. It is now possible to store your search and replace string(s) in a file so they do not have to be specified on the command line. This can be particularly useful when using variables to reformat a large line since it is possible to exceed the DCL limitation for line length. The /substitution qualifier is used to specify the file containing the strings. It's format is simple; enter a search string on one line and the replacement string on the next line. You can specify up to 3 pairs. For documentation purposes any lines in these files beginning with an exclamation point "!" will be treated as comments and ignored. When using /substitution the default directory is swap\_dat: and the default file type is .sub; if the source directory and file type are omitted from the file specification these values will be used.

```
! reformat_data substitutions file for SWAP
```

```
! first remove commas from the data lines
```

```
,
```

```
! next insert a line number at the beginning of the line
```

```
%5Z5,1000,100%
```

```
! finally reformat our data line, splitting it into two lines
```

```
%5a%%19b%%6c%%25d%
```

```
%a%%b%%/%a%%c%%d%
```

on body text in file general.txt:  
person 1           ,dept 1,Human Resources Dept.  
person 2           ,dept 2,Contract Administration

**\$ swap general.txt /subs=reformat\_data /column=(1,1)**

would result in  
01000person 1  
01000dept 1Human Resources Dept.  
01100person 2  
01100dept 2Contract Administration

NB: When working with variables to reformat a line using /column=(1,1) guarantees that the reformatting is only performed once on each line.

6. It is now possible to perform substitutions on a null string effectively allowing you to insert a text string into the output record without performing any substitutions. Two examples:

1. To insert a line number at the beginning of each line:

**\$ swap /column=(1,1) file.type "" "%5Z5,100,10%"**

this command will insert a 5 digital zero filled number at the beginning of each line. The /column=(1,1) is necessary to prevent swap from insert a number before every character in the line. In this case the line numbers would begin with 100 and increment by 10.

2. To insert a text string at column 25 of each line:

**\$ swap /column=(25,25) file.type "" "this string is being inserted"**

the specified text would be inserted beginning at column 25 of each record.

7. In earlier versions of Swap it was not possible to include the double quote character in search or replace strings because of the way DCL command interpreter parses strings and treats the double quote character as a delimiter. You can now include double quote characters in search strings by using two together. For example; to change all double quote characters in a file to single quotes:

**\$ Swap file.type "" ""**

since double quote double quote would be interpreted as NULL two double quotes are enclosed in double quotes for the search string and a single quote is enclosed in double quotes for the replacement string. When using the menu system you could simply specify a double quote for the search string and a single quote for the replacement string since these strings are not passed through the DCL command line interpreter.

8. Several minor problems with variable processing were discovered and corrected.

# Overview

This chapter gives a brief description of the SWAP utility. SWAP scans a set of ASCII files and replaces one string with another; with V5 up to 3 strings can be processed at one time. For example, the command:

```
$ SWAP *.FOR "mxu" "maximum_users"
```

replaces all occurrences of the obscure variable name MXU with the more readable MAXIMUM\_USERS in all FORTRAN source programs in the default directory.

```
$ SWAP *.FOR "mxu" "maximum_users" "mnu" "minimum_users" "au" "average_use"
```

replaces all occurrences of mxu with maximum\_users, mnu with minimum\_users and au with average\_use for all fortran source programs in the default directory.

SWAP combines the power of the SEARCH command with a sophisticated string substitution utility to produce an effective, easy to use software maintenance tool.

With the new variables feature added to V4 SWAP and some enhancements made to this feature in V5 Swap can also be used to reformat data files and/or generate and add a numeric value to the data lines; Swap can even be used to split lines into multiple lines duplicating any necessary values into each line.

## File Set

Specify a list of files or select a group of files using wildcards. You can include device and directory information or use logical names. Use /EXCLUDE to omit special files from the selected files.

## Strings

Each search string can be up to 255 characters long or null. It can contain leading or trailing blanks and both upper and lower case characters. The replacement string can be null, shorter than, equal to or longer than the search string. If the replacement string is null, the search string is removed from files.

If you specify /EXACT, SWAP will consider UPPERCASE and lowercase as different characters in the search string. Otherwise, they will be considered the same.

You can limit the search to a particular range of lines within each file (/RANGE) or a particular set of columns within each record (/COLUMNS). Replacement can be automatic with /NOCONFIRM or prompted with /CONFIRM. If you specify /CONFIRM, each occurrence is displayed and you have the options:

- Make the change
- Ignore this substitution but continue searching the file

- Ignore this file and continue with the rest of files
- Abort the command
- Help

When dealing with complicated or long strings, especially if there is more than one search and replace pair version 5 now allows you to store them in a file and specify that file as /substitution=file. This new feature will be particularly useful if the total length of the search and replace string pairs exceeds the single line DCL character limit or if the same substitutions are performed frequently.

### **Special Characters**

If the files contain special characters, you can eliminate them or substitute printable characters using the "%" character to designate the decimal or hexadecimal value of special characters. For example,

```
$ SWAP *.RNO %X0C ".page"
```

would replace the ASCII FORM FEED character with the text string ".page". You can only perform one special character substitution per SWAP command.

### **Record Length**

To specify a maximum output record length use /LENGTH. SWAP will truncate longer records to the length you specify. Shorter records will be left unchanged.

### **Remove Tabs**

SWAP can replace TAB characters with blanks. By default SWAP replaces each TAB character with 8 blanks. You can change the number 8 to other values if you wish.

SWAP is easy to learn and use. It acts like an OpenVMS utility. SWAP commands look like normal DCL commands. It comes with complete documentation, on-line HELP and comprehensive OpenVMS-style error messages.

**NOTE:** SWAP requires OpenVMS V5.4 or later. No additional OpenVMS layered products or licenses are necessary.

### **Reformatting Records**

SWAP V4 includes a new variable feature that supports quickly and easily reformatting data lines and/or generating and including a numeric value in the data lines of an output file. For a detailed explanation of how to use this feature refer to the section "Using SWAP Variables" on page ?.

## **SWAP Usage Summary**



This chapter describes how to use SWAP.

### **Invoking SWAP's DCL Interface**

To invoke SWAP, type SWAP at the command level along with the desired qualifiers and parameters.

#### **DON'T DO THIS!**

It is **not** necessary to assign a symbol such as \$SWAP == "RUN SWAP.EXE". SWAP is defined as a command by the installation procedure. Defining a symbol like this may prevent SWAP from running.

### **Invoking SWAP's Menu Interface**

To invoke SWAP's menu interface, type SWAP/MENU at the DCL level.

### **Help**

Enter "HELP SWAP" at the command level to get help. Within the menu a help overview screen is available from the main menu. A helpful prompt is also displayed at the bottom of the menu screen for each item on the menu as you scroll/move through them.

# Getting Started

This chapter describes how to get started with SWAP after you have installed it.

We recommend you do the following to get started:

1. Test the program

Do this by typing a command of the type:

```
$ SWAP/EXACT file_spec "search_string" "replacement_string"
```

where the replacement string is exactly the same as the search string. Under these conditions, SWAP acts like the OpenVMS SEARCH command. This allows you to see what it would find and replace without actually making any changes. You will get a sample of the output that SWAP generates.

**NOTE:** This will create a new version of the file\_spec file. The file created will be exactly the same as the original file. You may edit it to make sure.

Also try this command with the /STATISTICS qualifier to see the statistical data that is available.

2. Try a similar change through the menu interface. In the following examples the equivalent DCL level qualifier/command is shown in italics. While in the menu press the PF/3 key to go up one menu level or CTRL/Z to exit from the menu and return to DCL.

Enter the following command to start up SWAP's menu interface:

```
$ SWAP /MENU
```

Select **Modify Setup** if you want to change the following defaults

Confirm:	N	<i>/confirm</i>
Exact:	N	<i>/exact</i>
Log:	Y	<i>/log</i>
Stat:	N	<i>/statistics</i>
Truncation length:	2048	<i>/length</i>
Spaces/Tab:		<i>/tab</i>
Column Range:	1:2048	<i>/column</i>
Line Range:	1:2000000000	<i>/range</i>

Select **Search and Replace** when you're ready to begin modifying data files and enter:

Search String:	<i>command line parameter 2</i>
Replace String:	<i>command line parameter 3</i>
Search String 2	<i>command line parameter 4</i>
Replace String 2	<i>command line parameter 5</i>
Search String 3	<i>command line parameter 6</i>
Replace String 3	<i>command line parameter 7</i>
Search Files:	<i>command line parameter 1</i>
Exclude Files:	<i>/exclude</i>
Output File:	<i>/output</i>

enter <return> to skip past the 2<sup>nd</sup> and 3<sup>rd</sup> search and replacement strings if they are not being specified. After SWAP is finished changing the file(s) you can select View file from the main menu and SWAP will display the file for you to examine and verify that the changes have been made.

## Introduction to SWAP

This chapter provides a basic overview of how SWAP works. SWAP commands have the form:

```
SWAP [/qualifier] file_specification search_string replacement_string -  
      [search_string2 replacement_string2] [search_string3 replacement_string3] -  
      [/substitution=search_replace.sub]
```

### File specification

The file\_specification parameter designates one or more files to be changed. This parameter is mandatory. *In the menu enter this parameter on the Search & Replace Menu.*

### Search string

The search\_string parameter specifies the string to look for. This parameter is mandatory. It may be null. *In the menu enter this parameter on the Search & Replace Menu.*

### Replacement string

The replacement\_string parameter specifies the string that replaces the search\_string. This parameter is mandatory. It may be null. *In the menu enter this parameter on the Search & Replace Menu.*

### 2<sup>nd</sup> and 3<sup>rd</sup> search/replace pair

These pairs are optional and can be omitted.

### Substitution file

The substitution file contains the search and replace strings to be used for the specified file(s).

When using this qualifier omit all search and replace strings from the command line. Refer to /substitution in the qualifier section of this manual for a more detailed explanation.

### **Qualifiers**

You control the search and replacement operation using qualifier such as /EXACT and /RANGE.

The following chapters describe the parameters in more detail. Syntax examples are contained in the chapter on commands.

### **Specifying Files**

SWAP allows you to make a change to a group of input files. This chapter describes how to specify sets of files to be processed by SWAP.

### **Node**

The file specification CANNOT contain a node specification.

### **Devices and Directories**

The file specification can contain normal OpenVMS device and directory specifications such as:

```
"DUA0:[SWAP.SRC]*.FOR"
```

If device and directory aren't specified, the existing user defaults are used.

### **Logical Names**

The file specification can contain logical names in place of the device and directory specification. For example:

```
"USER$DISK1:*.COB".
```

### **Using a File List**

You can specify a list of files by separating the names with commas. Here's a sample:

```
MAIN.FOR,EXTRACT.FOR,REPORT.FOR,COLLECT.FOR
```

### **Wildcards**

You can use standard OpenVMS wildcard specifications in the file specifications. For example, "\*.FOR" specifies all FORTRAN source programs and "SYS%A.MAR" will select MACRO source such as SYS1A.MAR, SYS2A.MAR, etc. For more information about wildcards, refer to the OpenVMS DCL concepts manual.

**NOTE:** Use expressions such as \*.FOR or \*.COB or \*.RNO rather than \*.\* unless you are sure everything in the directory is an ASCII text file. SWAP does NOT check that input files are ASCII text files before attempting to process them.

### **Extentions**

SWAP does NOT assume "\*" if the file extension is omitted. If you don't specify any file

extension, SWAP will search for the file name with no extension (ie TEMP.;). If it is not found, an informational message is displayed on the screen saying the file was not found.

### **Excluding Files**

You can specify a group of files then exclude specific files in the group using the /EXCLUDE qualifier. For example,

```
$ SWAP/EXCLUDE=MAIN.FOR *.FOR
```

selects all FORTRAN programs except MAIN.FOR. The exclude list can contain more than one file if you enclose the names in brackets "()" and separate them with commas. Wildcards are allowed.

### **Specifying Strings**

SWAP looks for a search string in the input files and substitutes a replacement string if one has been specified. This chapter describes how to specify search and replacement strings.

### **String Specification**

#### ***Maximum Length***

Maximum string length is 255 characters. This applies to both the search and replacement strings. *The maximum string length for the menu is significantly smaller.*

#### ***Embedded Blanks or Double Quotes***

If a string contains leading, trailing or embedded blanks, or variables you must enclose the string in quotes such as "THIS STRING CONTAINS EMBEDDED BLANKS". If the string consists of a single word, quotes are not necessary but it will be converted to uppercase. If a string contains double quotes it must be enclosed in double quotes and two double quotes must be specified to prevent the DCL command line interpreter from incorrectly parsing the string; "THIS STRING CONTAINS ""DOUBLE QUOTES"" AROUND A STRING".

#### ***Case***

SWAP will automatically convert all strings to UPPERCASE unless you enclose them in quotes.

#### ***Null Strings***

A null string is entered by entering two quotes together (""). Both the search and replacement strings can be null.

#### ***Special Characters***

To enter special characters use the format %Xnn or %Dnnn where "%" designates a special character, "X" or "D" indicates hexadecimal or decimal and "nn" contains a hexadecimal or decimal value. For example the following command removes the ASCII LINE FEED character:

```
$ SWAP *.TXT %D010 ""
```

Here's another example which replaces the ASCII FORM FEED character with the string ".PAGE" character:

```
$ SWAP *.RNO %X0C ".page"
```

A table of ASCII characters with their decimal and hexadecimal values is contained in an appendix at the end of this manual. *Special characters may be entered this way through the menu interface as well.*

**NOTE:** Special character strings are limited to 1 character only. They must start with the character "%". The special character specification may be enclosed in quotes (ie "%D010"). If a string contains a special character specification at the beginning of the string, it will be interpreted as a special character (ie "%X20abcd" - this will be interpreted as a special character and the "abcd" will be ignored).

### **Search String**

The search string must be present on the command line unless it is included in a substitutions file (/substitution=file.sub). It may be null (when the /TAB or /LENGTH qualifiers are used and no string substitutions are required). With V5 of Swap there maybe up to 3 search strings specified in a single command or substitutions file.

### **Replacement String**

The replacement string must be present on the command line unless it is included in a substitutions file (/substitution=sub). It can be null, shorter than, equal to or longer than the search string. If the replacement string is null, the search string is removed from the files. With V5 of Swap there maybe up to 3 replacement strings specified in a single command or substitutions file.

You can specify a special character in the replacement string as well as the search string.

Search and Replacement strings must always be specified in pairs or an error will be reported.

## **Controlling the Search**

SWAP looks for a search string in the input files and substitutes a replacement string if one has been specified. If more than one pair has been specified all substitutions are performed on a single line before moving on to the next line. This chapter describes how to control the search.

### **Sequence of Operations**

If you specify record truncation and/or TAB removal, the operations are done in the following order:

- Remove TAB characters
- Replace strings
- Truncate record

## Case

SWAP will take case (ie UPPERCASE or lowercase) into account or ignore it depending on the /EXACT qualifier. If case is ignored (the default), the following strings: "TEST" and "test" would match the string "TEST" and be replaced. If case is significant, only "TEST" would match and be replaced. *When using the menu interface control case matching through the Modify Setup Menu.*

The search string is the only parameter the /EXACT has any effect on. The replacement string will always be substituted in as it appears.

## Limiting the Search Area

SWAP allows you to limit the records within each file and the columns within each record that are searched. This allows you to leave portions of the file untouched even if they contain the search string.

### *Ignoring Records*

Specify which records are to be searched by specifying a range using the /RANGE qualifier. SWAP numbers the records in a file starting at 1 and incrementing by 1. For example, if you specify /RANGE=(27,450), only records between 27 and 450 inclusive will be searched. Records numbered 1-26 and 450 to the end of the file would be left unchanged. *When using the menu interface control range through the Line Range item on the Modify Setup Menu.*

**NOTE:** If the /RANGE or /COLUMNS qualifiers are used, NO changes will be made outside of the specified borders. This means that tabs and truncation will not be performed outside of these borders.

### *Ignoring Columns*

Specify which columns are to be searched by specifying a set of columns using the /COLUMNS qualifier. SWAP numbers the columns in a record starting at 1 and incrementing by 1. For example, if you specify /COLUMNS=(7,72), only columns between 7 and 72 inclusive will be searched. Columns 1-6 and 73 to the end of the record would be left unchanged. When using variables to reformat lines specify /COLUMNS=(1,1) to prevent the reformatting from being done more than once to the same data line. *When using the menu control columns through the column range item on the modify setup menu.*

## Confirm

If you specify /CONFIRM, SWAP will ask you what you want to do each time it finds the search string. This allows you to control the replacement procedure more closely. You have the following options:

- Make the change
- Ignore the change but continue searching the record and file.
- Ignore this file but continue searching the remaining files. The current file will remain unchanged and the search will continue with the next file.
- Abort the search and return to command level. The current file will remain unchanged, SWAP will terminate and the DCL prompt will appear.

- Help

*When using the menu control confirm through the confirm item on the modify setup menu.*

## Controlling Output

SWAP can display a log of the changes it makes to files. This chapter gives an overview of ways to control the output.

### Log

If you specify /LOG (this is the default), SWAP will show the contents of each line before and after it is changed. This gives you an exact record of which lines in which files were changed.

*When using the menu controlling logging through the log item on the modify setup menu.*

**NOTE:** If the input records are longer than 80 characters and the terminal is set to WRAP, they will be displayed on the screen as wrapped lines.

SWAP displays the name of each file that is changing, how the line looked before and how the line looked after the change(s). It also displays the line number of the line it changed. If multiple changes are being performed on each line this can generate a lot of output; consider specifying /nolog with /output to write this output into a file instead of displaying it on the user terminal. If the swap command produces unexpected results the /output file can be reviewed to determine where the problem is occurring.

### Output

Copy the log data to an output file using the /OUTPUT qualifier. *When using the menu enter an output file name for output file on the search and replace menu.*

### Statistics

SWAP accumulates statistics on the changes it makes. If you specify the /STATISTICS qualifier, it includes these numbers in the log report and the output file. *When using the menu control displaying statistics through the stat item on the modify setup menu.*

It is recommended that the /STATISTICS qualifier be used when many substitutions are to be performed on a wild carded file specification. This provides a good audit trail if something goes wrong. /STATISTICS is the ONLY way to find out how many tabs were substituted or how many records were truncated.

## Removing Tabs



This section describes how to use SWAP to replace TAB characters with blanks. By default, SWAP does not replace TAB characters with blanks, if /TAB is specified without a value SWAP will substitute eight (8) blanks for each TAB character. To change this number use the /TAB=n qualifier. *When using the menu enter the number of spaces tabs should be changed to in the spaces/tab item on the modify setup menu; if this value is blank no substitution will take place.*

If you want to change all the tabs in a file (or group of files) with spaces, but do not want to perform a substitution, use the following command:

**\$ SWAP/TAB=n/STATISTICS file-name.ext "" ""**

The above command will convert all tabs in file-spec.ext to n spaces. It will not perform any other substitution.

**NOTE:** Tab removal is done BEFORE the record is truncated or searched. Tabs are replaced within the /RANGE and /COLUMNS boundaries ONLY. It is strongly suggested that the /STATISTICS qualifier be used with the /TAB qualifier. This will provide statistics regarding the tab substitutions made.

## Truncating Records

This chapter gives an overview of the SWAP procedure for truncating records.

SWAP allows you to truncate records to a maximum length. Records less than the maximum length are left unchanged. You specify the maximum record length using the /LENGTH qualifier. If you specify a length greater than the RMS maximum record length (EDT creates text files with a maximum length of 255 bytes) for the file, the lesser value will be used. *When using the menu control record length through the Truncate Length item on the modify setup menu.*

If you want to truncate all the records in a file (or group of files) but do not want to perform a substitution, use the following command:

**\$ SWAP/LENGTH=n/STATISTICS file-name.ext "" ""**

The above command will truncate all records in file-spec.ext to n bytes. It will not perform any other substitution.

**NOTE:** Record truncation is done AFTER the record is searched for strings. Records are truncated within the /RANGE boundaries ONLY. It is strongly suggested that the /STATISTICS qualifier be used with the /LENGTH=n qualifier. This will provide statistics regarding the number of records truncated.

## Limits

This chapter describes the limits SWAP applies to various lists.

Maximum record length	2,048
Maximum search and replacement string lengths	255
Maximum length of file name(s)	255
Maximum number of above window lines	5
Maximum number of below window lines	5
Maximum number of search files in list	64
Maximum number of exclude files in list	64

## Logicals

This section describes the logicals SWAP requires to run. These logicals, if defined, must be in the system table.

**SWAP\_CDU** - This logical points to the [SWAP.CDU] directory. It is used to access the SWAP command definition table. SWAP\_CDU is optional.

**SWAP\_COM** - This logical points to the [SWAP.COM] directory. It is used to access SWAP DCL command procedures. SWAP\_COM is optional.

**SWAP\_DAT** - This logical points to the [SWAP.DAT] directory. This logical is maintained for compatibility with other products by Saiga Systems. SWAP\_DAT is optional.

**SWAP\_DOC** - This logical points to the [SWAP.DOC] directory. The help files are stored in this directory. SWAP\_DOC is optional.

**SWAP\_EXE** - This logical points to the [SWAP.EXE] directory. It is used to access the SWAP executables. It is required. If you cannot access SWAP check to see that this logical is defined in the system table.

**SWAP\$VAR\_DELIM** - If this logical is defined to a single character that character will be treated as the delimiter for variables instead of the default character, "%." For example, to make the vertical bar character your variable delimiter:

```
$ define swap$var_delim "|" [/system]
```

if /system is added the variable delimiter will be set for everybody using swap on this node. Omitting /system will set the delimiter for your current process only.

# SWAP Commands

This chapter describes the SWAP commands that can be issued at the DCL level.

It provides a complete command reference "manual" for the experienced SWAP user.

## SWAP

The SWAP command activates SWAP.

SWAP commands look and act like DCL commands. The program was designed to conform to normal OpenVMS DCL command syntax.

SWAP commands have the format:

```
$ SWAP[/qualifier...] file_spec search_string replacement_string -  
[search_string2 replacement_string2 search_string3 replacement_string3]
```

SWAP is the name of the command and /qualifier is the name of a command qualifier. There are at least three parameters. Lowercase and uppercase characters in command and qualifier names are equivalent. As in DCL, '/' serves as a delimiter and command lines can be continued on to another line by terminating the line with a hyphen, typing <RETURN> and typing more of the command on the next line.

## Abbreviating Commands

As with DCL, all SWAP commands and switches can be truncated to a unique string (eg "SWAP" can be truncated to "SW" and "CONFIRM" can be truncated to "CON"). This provides compact command entry for experienced users. In command procedures, it is best to avoid abbreviating command names and qualifiers. Please refer to the OpenVMS DCL manual for more details.

The following sections describe the purpose of each SWAP command and lists the qualifiers that are valid to use with it. Then it lists all the qualifiers in alphabetical order and describes each qualifier in detail.

## Editing Command Lines

As with DCL, SWAP commands and qualifiers can be edited using terminal keys. Command-line editing is most useful for modifying long command lines.

## Help

The HELP command provides access to SWAP's on-line HELP text.

Format:

**\$ HELP SWAP**

SWAP's HELP text is stored in the system's HELP library along with the OpenVMS help text.

# MENU

An example of all the SWAP menu screens follows. Enter \$SWAP/MENU to enter the menu interface. Pressing the PF3 key will move you back one menu level; CTRL/Z will exit you from any menu to the DCL level.

## SWAP Main Menu

<b>SWAP MAIN MENU</b>
Search & replace Modify setup View file Help overview Information Exit to DCL
Please select a menu option

## Search & Replace Menu Modify Setup Menu

<b>FILE ENTRY SCREEN</b>
Search string: █ Replace string: Search files: Exclude files: Output file:
Enter the string to search for

MODIFY SETUP

Confirm: N  
Exact: N  
Log: Y  
Stat: N  
Truncation length: 2048  
Spaces/Tab:  
Column Range: 1:2048  
Line Range: 1:2000000000

Enter "Y" or "N"

Information Screen  
View File

INFORMATION SCREEN

SWAP 4.0  
Copyright Saiga Systems Inc. 1989-1996, All Rights Reserved  
Phone 1 (800) 561 8876  
1 (403) 263 1151  
Fax 1 (403) 263 0744  
BBS 1 (403) 233 8053

Licensed to: SAIGA SYSTEMS SOFTWARE INC.  
License name: COHORT3VAX  
License node: -SITE-  
License expiry: 12/31/2099

Current date: 2-DEC-1999 09:40:29.08  
Node name: EAGLE  
VMS version: V5.5-1  
HW type: MicroVAX 3100  
Cluster member: Y  
Number of nodes: 3

Press PF3 to exit

FILE ENTRY SCREEN

File name:

Enter the file name you wish to view



## Using Variables to Reformat Files

Swap allows you to quickly and easily reformat files using string variables for text information or a numeric variable to automatically generate and include a numeric value in the output records. A new variable %/% added to V5 allows you to split the final output record into multiple lines wherever it is included in the replacement string. *When doing reformatting operations it is important to specify /column=(1,1) or the reformatting may be performed more than once with unexpected results. We also recommend not performing string substitutions in the same SWAP run as reformatting.*

Variable operations can also be successfully performed from within the Swap menu if you enter the “modify setup” screen first and set the column range to “1:1”.

When reformatting data it is not necessary to account for the entire data line with variables. Any text remaining on the data line will be copied exactly to the end of the new data line. Also if SWAP encounters any variables specified in the search string and not specified in the replace string the data assigned to that variable will be discarded during output of the new record.

### SYNTAX

The location of a variable determines the syntax used. When specifying string variables in the search string they must include a length value; in the replace string the length cannot be specified - it is inherited from the search string value. The numeric string variables “N” and “Z” and the split variable “/” can only be specified in the replacement string; they are not valid in the search string.

The following table shows examples of valid variable specifications. Following the table there are several examples of how this new feature could be used.

Location	Variable Type	Syntax
Search String	String	###A% - assign ## characters to the variable A
Replace String	String	%A% - this is where the 15 characters stored in variable A should be included in the output string
Search String	Numeric	%N% - this variable is reserved for numeric values and CANNOT be included in the search string! %Z% - is also reserved for numeric values.
Replace String	Numeric	##1N#2,#3,#4% - this variable reserves #1 positions for a numeric value with leading spaces. The value will be #2 positions, starting at #3 and incremented by #4 each time it is inserted. ##1Z#2,#3,#4% - is identical to the “N” variable but the generated numeric string will be filled with leading zeros.

Search String	Split	%% - this variable is reserved for splitting the output line into multiple lines and CANNOT be included in the search string!
Replace String	Split	%% - this variable tells Swap to split the output line after all substitutions and reformatting have been completed. You can split the final output line into more than two lines by specifying extra %% variables.

## EXAMPLES

- Reordering columns 2-4 in a data file.  

```
$ SWAP /COL=(1,1) test.dat "%5a%%5b%10c%5d%" "%a%%d%%b%%c%"
```

Sample data line before (for clarity spaces are indicated with the ^ character):

```
^^100^1.25Item^3701b^^147Customer 217011/30/2004
^^200^1.23Item^4006c^^^21Customer 217011/30/2004
```

after:

```
^^100^^147^1.25Item^3701bCustomer 217011/30/2004
^^200^^^21^1.23Item^4006cCustomer 217011/30/2004
```

Notes: The A variable was used to keep the first five characters of the data line unchanged, the last portion of the data line was copied exactly from the old record to the new record.

- Reordering columns 2-4 in a data file; we'll add a 6 character position, zero filled line counter to the output file. The counter will start at 1,000 and be incremented by 100.  

```
$ SWAP /COL=(1,1) test.dat "%5a%%5b%10c%5d%" "%6z6,1000,100%%a%%d%%b%%c%"
```

Sample data line before (for clarity spaces are indicated with the ^ character):

```
^^100^1.25Item^3701b^^147Customer 217011/30/2004
^^200^1.23Item^4006c^^^21Customer 217011/30/2004
```

after:

```
001000^^100^^147^1.25Item^3701bCustomer 217011/30/2004
001100^^200^^^21^1.23Item^4006cCustomer 217011/30/2004
```

Notes: The A variable is again used to preserve a portion of the record, the numeric variable z is included in only the replace string and the last portion of the data line was copied exactly from the old record to the new record.

- It is possible using variables to take the last portion of a variable length record and move it elsewhere in the record. The text will be padded with spaces to the specified length and the final output records will all be the same length. In this example we'll also add line numbers to the records that are not zero-filled.

```
$ SWAP test.dat "site 1%a20%" "site 2%3N3,1,1%%a%"
```

Sample data line before (for clarity spaces are indicated with the ^ character):

```
Simpson Gilbert      site 1Technical Support
Norma Garriock      site 1Sales
```

after:

```
Simpson Gilbert      site 2 1Technical Support
Norma Garriock      site 2 2Sales
```

Notes: This is the only situation where it is safe to omit the /column qualifier when using variables; if the substitution should take place beginning with a string that is found only once in a data line. Since the line is changed beginning with the matched string the first portion of the data line remains unchanged. There is a potential for unexpected results when manipulating records this way; please verify the output carefully; if the data file is not too large consider using /confirm.

3. Reformatting a data line containing monthly sales values into multiple lines, one per month, with a header record. During reformatting a key value will be created for each group starting at 1000 and incrementing by 100.

```
$ swap /column=(1,1) custdat.txt /substitutions=reformat
```

the reformat file contains:

```
! Reformat customer records for 1 record / month and add key
! 1st search string is null and replace adds key
```

```
"%5Z5,1000,100%"
```

```
! 2nd search string outlines record while replace reformats
! And splits it into 3 lines adding some double quotes and
! commas
```

```
"%5x%15a%10b%7c%11d%7e%11f%"
```

```
"%x%, "%a%", "%b%"/%x%, "%c%", %d%/%x%, "%e%", %f%"
```

```
Company xyz      Location 101/2004$ 1,479.4502/2004$ 247.45
Company abc      Location 201/2004$ 444.4402/2004$ 0.00
Company lmn      Location 101/2004$79,321.0002/2004$ 7,987.00
```

after:

```
01000,"Company xyz      ","Location 1"
01000,"01/2004", $ 1,479.45
01000,"02/2004", $ 247.45
01100,"Company abc      ","Location 2"
01100,"01/2004", $ 444.44
01100,"02/2004", $ 0.00
01200,"Company lmn      ","Location 1"
01200,"01/2004", $79,321.00
01200,"02/2004", $ 7,987.00
```

## Qualifiers

This section describes the purpose of each SWAP qualifier. Then it lists all the qualifiers in alphabetical order and describes each qualifier in detail.

It provides a complete qualifier reference "manual" for the experienced SWAP user.

### Qualifier Syntax

SWAP qualifiers look and act like DCL qualifiers. They conform to normal OpenVMS DCL command qualifier syntax.

A qualifier consists of a keyword, or a keyword followed by a value or list of values. The keyword starts with a slash. All qualifiers apply to the entire command. SWAP qualifiers have one of the following formats:

SWAP/qualifier

SWAP/qualifier=n (a numeric value)

SWAP/qualifier=x (a alphanumeric value)

SWAP/qualifier=( ) (a list of values separated by commas)

All the qualifiers have defaults.

### Wildcards

The wildcard characters "\*" and "%" can be used in filenames for searching or with the /exclude qualifier.

## SWAP Qualifier List

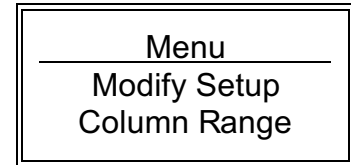
This page contains a complete list of all SWAP qualifiers. Subsequent pages contain detailed descriptions of each qualifier.

The following qualifiers can be used with SWAP

/COLUMNS=(start-column[,end-column])	Define columns to search
/CONFIRM	Prompt user for action
/EXACT	Use case in search
/EXCLUDE=(file-spec[,...])	Files to omit
/LENGTH=maximum-record-length	Truncate records
/LOG	Send messages to terminal
/MENU	Invoke SWAP's menu interface
/OUTPUT[=file-spec]	File for messages
/RANGE=(start-number[,end-number])	Define records to search
/STATISTICS	Report substitution statistics
/SUBSTITUTION=file-spec	File containing search and replace strings
/TAB[=number-of-spaces]	Replace TAB with blanks
/VERSION	Show the SWAP version you are running
/NOCONFIRM	Don't prompt user
/NOEXACT	Don't consider case
/NOLOG	Don't send messages
/NOOUTPUT	Don't record messages
/NOSTATISTICS	Don't report statistics
/NOTAB	Don't replace TAB's

## **/COLUMNS**

Indicates which columns to search in each record. If /COLUMNS isn't specified, all columns from 1 to EOL (End of Line) will be searched. Start column must be 1 or greater. End column must be greater than or equal to start column. If end column is omitted, EOL is assumed. The default is /COLUMNS=(1,EOL).



## **FORMAT**

/COLUMNS=(start-column[,end-column])  
default /COLUMNS=(1,EOL)

## **KEYWORDS**

start-column

The start-column is the first column of each record in the file that SWAP should begin checking for matches at. The default value is 1.

end-column

The end-column is the last column of each record in the file that SWAP should check for matches at. The default value is the end of each line (EOL).

## **EXAMPLES**

**\$ SWAP /COLUMN=(10,40) \*.COM "Changed by" "Modified by"**

Change every occurrence of the string "Changed by" to "Modified by" in columns 10 to 40 of every command procedure in the current directory.

**\$ SWAP/COLUMNS=(7,72) \*.\* "Johnny" "Bobby"**

Only columns 7 through 72 inclusive will be searched for "Johnny" (ignoring case).

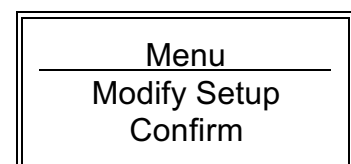
Columns 1-6 and 73 to EOL will be left unchanged. All files in the default directory will be searched.

**NOTE:** A tab character is considered to take up one character space. When a tab is displayed on the screen, it usually is shown as 8 spaces. The /COLUMNS qualifier considers tabs to be one character. Therefore, if a line contains a tab, then a space, it will be displayed as 9 spaces but column one will contain the tab and column 2 will contain the space.

**NOTE:** When using variables the command line should always include /column=(1,1) or the column value set to "1:1" in the modify setup menu item. This will prevent Swap from moving over 1 column after the substitution is done and performing it again until the end of the line is reached.

## **/CONFIRM**

Controls whether the user is to be prompted before each substitution. If the user specifies /CONFIRM, SWAP will provide the following options:



- Make the substitution
- Ignore this substitution, continue with the file
- Ignore this file, continue with other files
- Abort the command
- Help

The default action is for SWAP not to prompt for confirmation before substitutions. This qualifier is for string substitutions ONLY. The /TAB and /LENGTH qualifiers cannot be confirmed, they are performed regardless of the state of the /CONFIRM qualifier.

#### FORMAT

/CONFIRM  
/NOCONFIRM (Default)

#### KEYWORDS

None

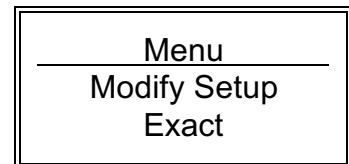
#### EXAMPLES

##### **\$ SWAP/CONFIRM \*.\* A B**

SWAP will ask the user what action to take each time it finds the search string "A". All files in the default directory will be searched.

#### **/EXACT**

Controls whether SWAP is case sensitive during the search; the upper and lower case of the search string should be considered. If the user specifies /EXACT, each character of the search string must agree in case with the input string for a match to occur. The default action is to not consider case when looking for matches to the search string.



#### FORMAT

/EXACT

#### KEYWORDS

None

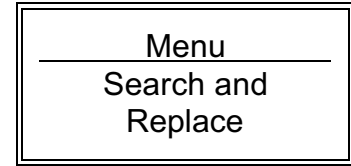
#### EXAMPLE

##### **\$ SWAP/EXACT \*.DAT "John" "Ralph"**

SWAP will test for case during the search (it will look for "John"). It will substitute "John" for "Ralph" in all .DAT files in the default directory.

## **/EXCLUDE**

Use the /EXCLUDE qualifier to provide SWAP with a list of files that should not be processed. If /EXCLUDE isn't specified, all files specified will be searched.



## **FORMAT**

/EXCLUDE=(file-spec[,...])

## **KEYWORDS**

file-spec

The file specification of the file or files you want SWAP to not process. To specify multiple files, separate the file names with commas, and enclose the list in parenthesis. If a single file is specified, the parenthesis are not necessary. You cannot use device, directory, or logicals in the /EXCLUDE clause but wildcards are allowed. If the file exists in two directories (specified in the input specification), and that file is in the /EXCLUDE the file will be ignored in both directories.

## **EXAMPLES**

**\$ SWAP/EXCLUDE=MAIN.FOR \*.FOR "test" "temp"**

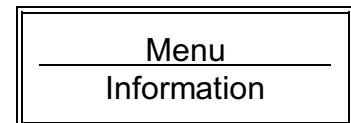
All FORTRAN source programs in the default directory will be processed except MAIN.FOR. "test" (ignoring case) will be substituted with "temp".

**\$ SWAP/EXCLUDE=(CHARLIE.RNO,SYSUPD.\*) \*.RNO "test" "temp"**

All runoff (.RNO) files in the default directory will be processed. CHARLIE.RNO and all files named "SYSUPD" will be ignored. "test" (case ignored) will be substituted with "temp".

## **/INFORMATION**

This qualifier specifies that you want SWAP to display its information screen. This screen will display various system and license related items. The /VERSION qualifier also displays the same information.



## **FORMAT**

/INFORMATION

## **KEYWORDS**

None

## **EXAMPLES**

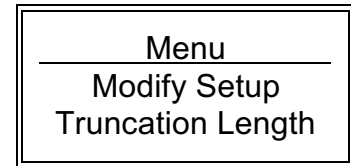
**\$ SWAP/INFORMATION**

SWAP will display the information screen.



## **/LENGTH**

Indicates the maximum record length for the output file. Records longer than this will be truncated. Shorter records will be left unchanged. If /LENGTH isn't specified, records up to the RMS maximum record length will be unchanged. It is suggested that the /STATISTICS qualifier be used with the /LENGTH qualifier. This will provide important audit information about truncated records.



## **FORMAT**

/LENGTH=max-record-length  
/LENGTH=RMS\_max\_record\_length (Default)

## **KEYWORD**

max-record-length

The maximum record length that SWAP will write out when modifying a file. Max-record-length must be greater than 1. If it is larger than the RMS maximum record length for the file, the smaller value is used. The RMS maximum record length of text input files is frequently 255 bytes, this is the length the EDT text editor creates files with.

## **EXAMPLES**

**\$ SWAP/LENGTH=80 \*.\* "" ""**

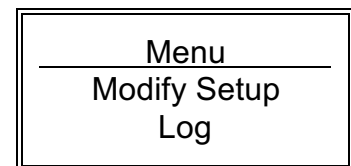
Input records longer than 80 characters will be truncated to 80 for all files in the default directory. No other substitution will be performed.

**\$ SWAP/LENGTH=72 \*.\* "coffee" "tea"**

All files in the default directory will be searched for "coffee" (ignoring case). "tea" will be substituted for each occurrence of "coffee". Each record will be truncated (if necessary) to 72 characters.

## **/LOG**

Controls whether swap logs its actions to the terminal as they are taken. There are numerous information messages that will be displayed on the screen along with the report. These all start with SWAP-I-. These are only informational messages and not errors. They are there to help you monitor SWAP as it processes the files. The default action for SWAP is to log its actions.



## **FORMAT**

/LOG  
/NOLOG (Default)

## **KEYWORDS**

None

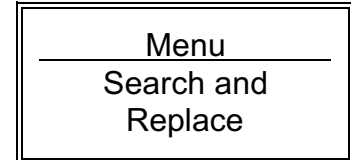
## EXAMPLE

**\$ SWAP/NOLOG \*.\* jack "Jill"**

All files in the default directory will be searched for the string "jack" (ignoring case). "Jill" will be substituted in for all occurrences of "jack" (ignoring case). No messages will be written to the screen.

## /OUTPUT

Controls whether SWAP writes a copy of the action messages to a file while it's processing your request. Use /OUTPUT if you want to keep or print a copy of all the changes SWAP has made.



## FORMAT

/OUTPUT=file-spec

## KEYWORDS

file-spec

Can be any valid file specification. If no file specification is provided with the /OUTPUT qualifier the messages will be written to SWAP\_CHANGES.LOG in the current directory.

## EXAMPLES

**\$ SWAP/OUTPUT \*.\* "a" "A"**

All files in the default directory will be searched for "a" (ignoring case). For each occurrence found, "A" will be substituted in. A record of all substitutions will be written to SWAP\_CHANGES.LOG.

**\$ SWAP/OUTPUT=CHANGES.SWP \*.\* "telephone" "dictaphone"**

All files in the default directory will be searched for "telephone" (ignoring case). For each occurrence found, "dictaphone" will be substituted in. A record of all substitutions will be written to CHANGES.SWP.

## /MENU

This qualifier specifies that you want to use the menu interface for SWAP.

## FORMAT

/MENU

## KEYWORDS

None

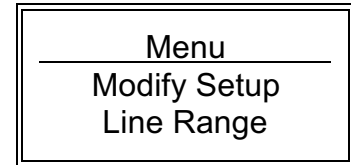
## EXAMPLES

**\$ SWAP/MENU**

SWAP will display the menu interface.

## **/RANGE**

Indicates which records to search in each file. If /RANGE isn't specified, all records from 1 to EOF (End of File) will be searched. Start record must be 1 or greater. End record must be greater than or equal to start record. If end record is omitted, EOF is assumed.



## **FORMAT**

/RANGE=(start-record[,end-record])  
/RANGE=(1,EOF) (Default)

## **KEYWORDS**

start-record

The start-record is the first record of each file that SWAP is to begin checking for matches at. The default value is 1.

end-record

The end-record is the last record of each file that SWAP is to check for matches in. The default value is to check to the end of the file (EOF).

## **EXAMPLES**

**\$ SWAP/RANGE=(30,400) \*.\* "cord" "wire"**

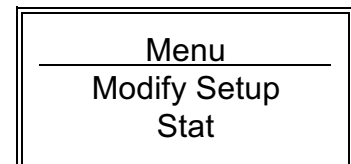
All files in the default directory will be searched for "cord" (ignoring case). All occurrences found will be changed to "wire". For each file, records 1-29 and 401 through end of file will be ignored. The rest will be searched.

**\$ SWAP/RANGE=7 \*.\* "smell" "taste"**

All files in the default directory will be searched for "smell" (ignoring case). All occurrences found will be changed to "taste". For each file, the first 6 records will be ignored. The rest will be searched.

## **/STATISTICS**

Controls whether string substitution statistics are reported to the screen (/LOG) and/or to the output file (/OUTPUT). It is suggested that this qualifier be used when the /TAB or /LENGTH qualifiers are used. This is the best way to keep track of these changes. The default action is to not report statistics.



## **FORMAT**

/STATISTICS  
/NOSTATISTICS (Default)

## **KEYWORDS**

None

## **EXAMPLE**

### **\$ SWAP/STATISTICS \*.FOR "FORTRAN" "COBOL"**

All files with .FOR extensions will be searched for "FORTRAN" (ignoring case). All occurrences found will be changed to "COBOL". SWAP will report which files were changed and how many substitutions were made. There are also a number of other statistics that will be included.

## **/SUBSTITUTION**

Use this qualifier to tell Swap the name and location of a text file containing the search and replace strings it should process. This qualifier is particularly useful if there are multiple search and replace strings (the format of this file would make updating these strings easier), long strings that might exceed the 255 character DCL command line limit or variables which can make the search and replace strings quite complicated. Any other qualifiers specified, such as /TAB, /LENGTH, /RANGE or /COLUMNS will be processed normally.

There is no menu equivalent of /substitution since it is assumed that this new feature of V5 will be used primarily for search and replace operations that are done frequently, often in batch jobs.

This qualifier is optional but when it is used the search and replace string should not be included on the command line.

## **FORMAT**

/SUBSTITUTION=filespec  
/NOSUBSTITUTION (default)

## **KEYWORDS**

filespec

The substitution qualifier must include a valid OpenVMS file specification for the file that contains the search and replace string(s); up to 3 pairs maybe specified. If no location is specified swap assumes the substitution file is in the SWAP\_DAT: directory; if no filetype is specified .sub is assumed. If /substitution is not specified Swap obtains the search and replace strings from the DCL command line.

Substitutions files must be sequential text files. Any line beginning with an exclamation mark is treated as a comment and ignored. A search string must be specified on one line with the replacement string specified on the next line. There must be one valid search/replace pair in the file, however, up to 3 search/replace pairs maybe included.

## **EXAMPLE**

The following example is meant to highlight some of the new features of V5 as well as illustrate the use of substitution files. A simple substitution file with 1, 2 or 3 pairs of search and replace strings will also be illustrated. A group of substitutions/reformats are commonly performed on a data file created by an existing application so it can be more easily imported into a spreadsheet. First all line feed characters are stripped from the file since the application inserts these as a field delimiter and then the 1- string is removed

from phone numbers and finally the record is rewritten with a different phone number format and a space is added into the postal code. The company name and phone number are written out on a separate line as well. For clarity some spaces have been omitted

**Data (textfile.dat):**

```
company 1  contact      street address      city  ABANANAN 1-(000)-000-0000x000
```

```
Substitutions file #1 (swap_dat:sub1.sub)
! Perform substitutions to remove line feeds and 1- from
! telephone numbers
%X0A
```

```
! 2nd string
1-(
(
```

```
Substitutions file #2 (swap_dat:sub2.sub)
! Because we are reformatting and must use /column=(1,1)
! This is done in a separate file. Using /column=(1,1) with
! Our earlier substitutions would prevent them from being
! performed. Each half of the postal code will be stored in
! a separate variable so we can insert the space
%10a%%10b%%20c%%5d%%2e%%3f%%3g%%1h%%3i%%2j%%8k%%1l%%3m%
%a%,%b%,%c%,%d%,%e%,%f% %g%/%%a%, ""i%-k%ext.%m%"
```

```
$ swap textfile.dat /substitutions=sub1
$ swap textfile.dat /substitutions=sub2 /column=(1,1)
```

**Output:**

```
company 1 ,contact      ,street address      ,city ,PR,ANA NAN
company 1,"000-000-0000ext.000"
```

**EXAMPLE**

A simple substitution with 3 strings.

```
The substitutions file (dev:varsub.dat):
! This file is used to change 3 very cryptic variables
! to more descriptive names in our cobol source files
```

```
mxu
"Maximum_User_Count"
mnu
"Minimum_user_Count"
avgust
"Average_of_User_Totals"
```

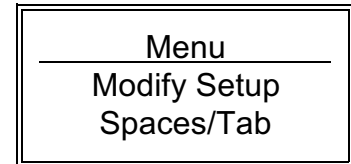
```
$ SWAP *.cob /substitution=dev:varsub.dat /confirm
```

This will make the substitutions included in the file on all .cob source files in the current directory. For safety sake /confirm has been specified so the user will have a chance to verify all substitutions before they are performed. Since these substitutions are being made on files in multiple

directories the use of a substitution file guarantees that the change will be consistent among all directories; if the strings were entered manually a capitalization might be changed or an underscore missed resulting in a different variable name in that directories files. The search strings are not included in quotes because we do not want them to be case sensitive; enclosing the replacement strings in quotes guarantees that case will be preserved.

### **/TAB**

Indicates the number of blanks to substitute for a TAB character. If /TAB isn't specified, TAB characters will be unchanged. If /TAB is specified without an argument, /TAB=8 will be used. Number of spaces must be equal to or greater than 1. Only tab characters within the boundaries specified by the /RANGE and /COLUMNS qualifiers will be changed.



It is suggested that the /STATISTICS qualifier be used with the /TAB qualifier. This will provide important audit information about tab substitutions.

### **FORMAT**

/TAB[=number-of-spaces]  
/NOTAB (Default)

### **KEYWORDS**

number-of-spaces

The number of spaces that each TAB character seen should be replaced with during processing. The default value is 8.

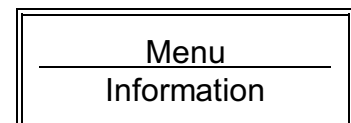
### **EXAMPLES**

#### **\$ SWAP/TAB=4 TEST.FOR "MAXIMUM" "MINIMUM"**

TEST.FOR will be searched for the string "MAXIMUM" (ignoring case). All occurrences will be changed to "MINIMUM". TAB characters will be replaced with 4 blanks in ALL records of TEST.FOR.

### **/VERSION**

Use the /VERSION qualifier to show the current version of SWAP you are using as well as all the product and license information. This qualifier is the same as /INFORMATION.



### **FORMAT:**

/VERSION

### **EXAMPLE:**

#### **\$ SWAP/VERSION**

Display which version of swap you are using and the information screen.

## Appendix A - Error Messages

SWAP uses OpenVMS-style error messages. This appendix lists every SWAP message in alphabetical order. The error messages are divided into the following severities:

- Warning                    %SWAP-W-   SWAP should still run
- Informational            %SWAP-I-   SWAP will still run
- Fatal                     %SWAP-F-   SWAP will terminate
- Error                     %SWAP-E-   SWAP will terminate

In all the following messages, 'file-spec' will contain a file specification when the message is signalled to the screen. The file specification will vary each time the message is written. The rest of the appendix lists error messages and recommended action.

**ABORTED**, Aborted procedure, changes to 'file-spec' will NOT be saved

The abort option of the confirm menu was chosen. The current file will not be changed (not even the changes up to the point abort was requested) and no more files will be processed. All files that were changed before the abort request will contain changes.

**BLANK**, Returned blank value

A blank value was entered in a field that requires a value. Please enter a value in the field.

**ERRCENTEXT**, Text was too wide to be properly centered

There was an error while centering a text string within a limited space. The string will be truncated.

**ERRCLOINP**, Error closing input file

There was an error closing the input file. The error messages that follow this error will further explain the problem.

**ERRCLOLOG**, Error closing log file (SYSS\$OUTPUT:)

There was an error closing the log file which is SYSS\$OUTPUT:. The error messages that follow this error will further explain the problem.

**ERRCLOOUTP**, Error closing output file

There was an error closing the output file (listing of all substitutions and statistics from the /OUTPUT qualifier). The error messages that follow this error will further explain the problem.

**ERRCLOTOUTP**, Error closing temporary output file

There was an error closing the temporary output file. The error messages that follow this error will further explain the problem.

**ERRCONVINCR**, Error converting incrementing value, increase the field width

During processing an incrementing value exceeding the maximum value that could be output in the space provided. Reenter the command and allow more space for the value OR decrease the increment amount.

**ERRORINCRVALUE**, Error parsing increment value or incrementing variable

There was an error parsing the increment value from an incrementing variable. Reenter the command with the incrementing variable formatted “#N#,#,#” or “#Z#,#,#”; all numeric values must be supplied.

**ERRINQINP**, Error inquiring about input file

There was an error when inquiring about the input file maximum record length. The error messages that follow this error will further explain the problem.

**ERRINQSUB**, Error inquiring about substitutions file

There was an error when inquiring about the attributes of the substitutions file. The error messages that follow this error will further explain the problem.

**ERROPEINP**, Error opening input file

One of the input files could not be opened. This is only a warning, but it should be noted. The error messages that follow this error will further explain the problem.

**ERROPELOG**, Error opening log file (SYSS\$OUTPUT:)

There was an error opening the log file which is SYSS\$OUTPUT:. The error messages that follow this error will further explain the problem.

**ERROPENFILE**, Error opening the file !AS

There was an error opening the specified file. Additional error messages should be displayed that provide additional information.

**ERROPEOUTP**, Error opening output file

There was an error opening the output file (listing of all substitutions and statistics from the /OUTPUT qualifier). The error messages that follow this error will further explain the problem.

**ERROPESUB**, Error opening the substitutions file

There was an error opening the substitutions file. The error messages that follow this error will further explain the problem. Substitutions files default to directory SWAP\_DIR and filetype .sub. If your substitutions file is located in different directory or has a different filetype try the swap command again giving a more complete file name.

**ERROPETOUTP**, Error opening temporary output file

There was an error opening the temporary output file. The error messages that follow this error will further explain the problem.



**ERRORPARSINCR**, Error parsing incrementing variable

There was an error parsing the incrementing variable. Reenter the command with the incrementing variable formatted “#N#,#,#” or “#Z#,#,#”; all numeric values must be supplied.

**ERRRNAGE**, Error processing the range !AS

There was an error while trying to transfer the specified value to an integer range value. Please verify the value and re-enter.

**ERRREAINP**, Error reading input file

There was an error reading the input file. The error messages that follow this error will further explain the problem.

**ERRREALOG**, Error reading log file (SYSS\$OUTPUT:)

There was an error reading the log file which is SYSS\$OUTPUT:. This is only a warning and you will be prompted for input again. The error messages that follow this error will further explain the problem.

**ERRRENAMING**, Error renaming temporary output file 'file-spec'

There was an error trying to rename the temporary file to a version higher than the input file. File-spec will be the name of the file. The error messages that follow this error will further explain the problem.

**ERRSTARTVALUE**, Error parsing start value or incrementing variable

There was an error parsing the incrementing variable; the starting value is either invalid or missing. Reenter the command with the incrementing variable formatted “#N#,#,#” or “#Z#,#,#”; all numeric values must be specified as valid integers.

**ERRVARSIZE**, Error converting ‘value’ to a size for the variable ‘variable’>

The value specified for the size of an incrementing variable has not been specified correctly. Reenter the command with the incrementing variable in the format %N#,#,#% or %Z#,#,#%; all numeric (#) values must be specified as valid integers.

**ERRWRITLOG**, Error writing log file (SYSS\$OUTPUT:)

There was an error writing to the log file which is SYSS\$OUTPUT:. The error messages that follow this error will further explain the problem.

**ERRWRITOUTP**, Error writing temporary output file

There was an error writing to the temporary output file. The error messages that follow this error will further explain the problem.

**FILE\_IGNORED**, File 'file-spec' was not processed

The specified file was not processed. It was ignored.

**FILETRUNC**, The file being loaded was truncated, line limit exceeded

The file being loaded to view was too large. Try viewing the file from the DCL prompt

with the TYPE command or a default VMS editor.

**FNF**, file !AS was not found>

The specified file was not found. Please check for spelling errors and re-enter the command.

**IGNORED**, Ignore file, changes to 'file-spec' will NOT be saved

The ignore option of the confirm menu was chosen. The current file will not be changed (not even the changes up to the point ignore was requested).

**INVALIDCOLS**, Invalid column value(s) specified

The /COLUMNS= entry was invalid. Check that the starting column is less than or equal to the ending column number.

**INVALIDLENGTH**, Invalid length (record) value was specified

The /LENGTH= entry was invalid. Check to make sure a value was specified with the qualifier. If a value was specified, make sure it was between 1 and 2048.

**INVALIDRANGE**, Invalid range value(s) specified

The /RANGE= entry was invalid. Check that the starting range is less than or equal to the ending range number.

**INVALIDTAB**, Invalid tab value was specified

The /TAB= entry was invalid. Check to make sure the specified value was between 1 and 255.

**INVCHAR**, !AS is an invalid character

An invalid character was specified. Please re-enter with a valid character.

**INVINTRANGE**, !ZL is invalid, you must enter a value between !ZL and !ZL

An invalid integer value was specified. The value must be between the indicated values in the error message.

**INKEY**, Last key entered was not valid for this screen

An invalid key was entered. Please refer to the manual for valid keys.

**NOINCRTYPE**, No increment variable type, must supply N or Z

A variable that appears to be specified as incrementing (ie: includes commas in the variable definition) is not named N or Z. Only N, leading spaces, or Z, leading zeros, are valid variable names for incrementing variables.

**NORMAL**, Normal successful completion

There was a successful completion.

**NOVAR**, There was no variable name within the variable delimiters

A delimiter was found without a closing delimiter OR two delimiters were found without a variable specified within them. If the % sign is the delimiter variables must be defined as %variable-name###% where ### is the length of the variable in the search string or %variable-name% with no length specified for variables in the replacement string.

**PROVIDE**, The !AS parameter must be supplied

The specified parameter must be supplied. Please re-enter the command with the required parameter.

**REQFIELDSIZE**, Must supply the field size for the incrementing variable

The value specified for the field size of an incrementing variable has not been specified correctly or has been omitted. Reenter the command with the incrementing variable in the format %#N#,#,% or %#Z#,#,%; all numeric (#) values must be specified as valid integers.

**RSPECIALUSED**, Replacement string was interpreted as a special character

The replacement string was interpreted as a special character (ie %x010). If the replacement string was not meant to be a special character, re-enter the command.

**SCREENEXIT**, User hit an exit or escape key

This is an internal status that should not be signalled. There is no user action required.

**SEARCHEXCL**, Search file 'file-spec' was excluded

This message tells you which files were excluded because they matched one of the exclude file specifications.

**SEARCHFNF**, Search file not found. File 'file-spec' not found.

One of the input file specifications was not found.

**SSPECIALUSED**, Search string was interpreted as a special character

The search string was interpreted as a special character (ie %x010). If the search string was not meant to be a special character, re-enter the command.

**TOOMANYEXCL**, Too many exclude files were specified

There were too many exclude files specified in the list on the command line. Re-enter the files with a maximum of 64 file specifications.

**TOOMANYSEAR**, Too many search files were specified

There were too many search files specified in the list on the command line. Re-enter the files with a maximum of 64 file specifications.

**VALIDINPUT**, Enter one of the following only: Y, N, I or A - Please reenter

This message displays all valid input. If invalid input is entered, this message is displayed as a short help message.

## Appendix B - ASCII Table

This table gives decimal and hexadecimal values of ASCII characters.

Dec	Hex	ASCII Character	Dec	Hex	ASCII Character
0	0	Null, Idle	24	18	Cancel
1	1	Start of Heading	25	19	End of Medium
2	2	Start of Text	26	1A	Substitute
3	3	End of Text	27	1B	Escape
4	4	End of Transmission	28	1C	File Separator
5	5	Enquiry	29	1D	Group Separator
6	6	Acknowledge	30	1	Record Separator
7	7	Bell	31	1F	Unit Separator
8	8	Backspace	32	20	Space
9	9	Horizontal Tab	33	21	! Exclamation Mark
10	0A	Line Feed	34	22	“ Double Quote
11	0B	Vertical Tab	35	23	# Pound Sign
12	0C	Form Feed	36	24	\$ Dollar Sign
13	0D	Carriage Return	37	25	% Percent Sign
14	0	Shift Out	38	26	& And sign
15	0F	Shift In	39	27	‘ Apostrophe
16	10	Data Like Escape	40	28	( Left Parenthesis
17	11	Device Control 1 XON	41	29	) Right Parenthesis
18	12	Device Control 2	42	2A	* Asterisk
19	13	Device Control 3 XOFF	43	2B	+ Plus Sign
20	14	Device Control 4	44	2C	, Comma
21	15	Negative Acknowledge	45	2D	- Hyphen
22	16	Synchronous Idle	46	2	. Period
23	17	End of Transmission Block	47	2F	/ Slash

Dec	Hex	ASCII Character	Dec	Hex	ASCII Character
48	30	0	75	4B	K
49	31	1	76	4C	L
50	32	2	77	4D	M
51	33	3	78	4E	N
52	34	4	79	4F	O
53	35	5	80	50	P
54	36	6	81	51	Q
55	37	7	82	52	R
56	38	8	83	53	S
57	39	9	84	54	T
58	3A	: Colon	85	55	U
59	3B	; Semi-colon	86	56	V
60	3C	< Less Than Sign	87	57	W
61	3D	= Equals Sign	88	58	X
62	3E	> Greater Than Sign	89	59	Y
63	3F	? Question Mark	90	5A	Z
64	40	@ Ampersand	91	5B	[ Left Square Bracket
65	41	A	92	5C	\ Backslash
66	42	B	93	5D	] Right Square Bracket
67	43	C	94	5E	^ Circumflex
68	44	D	95	5F	_ Underscore
69	45	E	96	60	` Accent grave
70	46	F	97	61	a
71	47	G	98	62	b
72	48	H	99	63	c
73	49	I	100	64	d
74	4A	J	101	65	e

Dec	Hex	ASCII Character	Dec	Hex	ASCII Character
102	66	f	115	73	s
103	67	g	116	74	t
104	68	h	117	75	u
105	69	i	118	76	v
106	6A	j	119	77	w
107	6B	k	120	78	x
108	6C	l	121	79	y
109	6D	m	122	7A	z
110	6E	n	123	7B	{ Left Brace
111	6F	o	124	7C	Vertical Bar
112	70	p	125	7D	} Right Brace
113	71	q	126	7E	~ Tilde
114	72	r			

## Command Summary

This page provides a quick reference guide to SWAP command and qualifiers that can be used at the DCL command level.

SWAP Start the SWAP program.

/COLUMNS=(start-column[,end-column])	Define columns to search
/CONFIRM	Prompt user for action /EXACT Use case in search
/EXCLUDE=(file-spec[,...])	Files to omit
/LENGTH=maximum-record-length	Truncate records
/LOG	Send messages to terminal
/MENU	Start the SWAP menu interface
/OUTPUT[=file-spec]	File for messages
/RANGE=(start-number[,end-number])	Define records to search
/STATISTICS	Report substitution statistics
/TAB[=number-of-spaces]	Replace TAB with blanks
/NOCONFIRM	Don't prompt user
/NOEXACT	Don't consider case
/NOLOG	Don't send messages
/NOOUTPUT	Don't record messages
/NOSTATISTICS	Don't report statistics
/NOTAB	Don't replace TAB's
HELP SWAP	Access the on-line help information.

# Index

## ASCII

- Changing Tabs to Spaces, 29
- Table, 34

## Case

- Making SWAP case sensitive, 24

## Controlling Swap

- Confirming changes, 23
- Logging changes, 26
- Making SWAP case sensitive, 24
- Redirecting output to a file, 27
- Specifying a line range, 27
- Specifying a maximum record length, 25
- Specifying what portion of a line should be searched, 23

## Error Messages, 30

## Excluding Files, 24

## Menu

- Information Screen, 19
- Main menu, sample, 18
- Modify Setup Menu, Sample, 19
- Search & Replace Menu, Sample, 18
- View File Menu, Sample, 20

## Qualifiers

- /Columns, 23
- /Confirm, 23
- /Exact, 24
- /Exclude, 24
- /Information, 25
- /Length, 25
- /Log, 26
- /Menu, 27
- /Output, 27
- /Range, 27
- /Statistics, 28
- /Tab, 29
- /Version, 29
- List of valid, 22

## Release Notes, 4

## Specifying Files

- Using /Exclude, 24

## SWAP

- System Information, 25, 29