

DSSUTL

**Hydrologic Engineering Center
Data Storage System Utility Program**

User's Manual

**Version 6.8
March 1995**

**Hydrologic Engineering Center
U.S. Army Corps of Engineers
609 Second Street
Davis, California 95616-4687
(916) 756-1104**

DSSUTL General Commands

AR Archive	CA Catalog
CH Check	CO Copy
DC Data Compression	DE Delete
DP Display Pathnames	DU Duplicate
ED Edit	EF Exchange Format
FC File Check	EV Exchange Variable
FQ File Query	EX Export
HE Help	FI Finish
IM Import	FO Format
PA Pathname	OP Open
RN Rename	RE Read Data
SQ Squeeze	RT Record Tag
TA Tabulate	ST Status
UD Undelete	TI Time
WR Write Data	

DSSUTL

Table of Contents

Chapter	Page
1. Introduction	1
1.1 Program Execution	1
1.2 Command Structure	1
2. Differences from Previous Version	3
2.1 New Commands	3
2.2 Additional Command Capabilities	3
3. General Commands	5
3.1 Archive	5
3.2 Catalog	7
3.3 Check	10
3.4 Copy	11
3.5 Data Compression	13
3.6 Delete	15
3.7 Display Pathnames	16
3.8 Duplicate	17
3.9 Edit	18
3.10 Exchange Format	20
3.11 Exchange Variable	26
3.12 Export	28
3.13 File Check	29
3.14 Finish	30
3.15 File Query	30
3.16 Format	31
3.17 Help	32
3.18 Import	33
3.19 Open	34
3.20 Pathname	35
3.21 Read Data	36
3.22 Record Tag(s)	37
3.23 Rename	39
3.24 Squeeze	40
3.25 Status	42
3.26 Tabulate	43
3.27 Time	45
3.28 Undelete	46
3.29 Write Data	47

4. Secondary Commands	49
4.1 Close	49
4.2 Debug	49
4.3 Inquire	50
4.4 No Operation	50
4.5 Set	50
4.6 Verify	50
5. Selective Catalog Capabilities	51
5.1 Introduction	51
5.2 Sort Order	51
5.3 Selection of Pathnames Based on Last Written Date or Program	51
5.4 Selection of Pathnames Based on Pathname Parts	52
6. Export - Import	53
6.1 Introduction	53
6.2 Procedure	53
6.3 Examples	54

Chapter 1

Introduction

DSSUTL provides a means of performing utility functions on data stored in the Hydrologic Engineering Center's Data Storage System (HEC-DSS). These functions include tabulating, editing, copying, renaming, and deleting data. The program also offers the capability of formatting and copying data into an ASCII sequential file for transfer to another computer, or for use by a program without DSS capabilities.

1.1 Program Execution

The program is executed by entering its name. The program responds by asking for the name of the DSS file to access. On MS DOS personal computers, the name may be selected by moving the cursor to one of the names in the list that is displayed on the screen (which are files that end in ".dss" in the current directory), then pressing the carriage return. After the DSS file name has been entered, the program gives the prompt "U>", where the user enters commands. To terminate the session, the command FINISH (or just FI) is entered.

If desired, the commands may be specified in a separate input file, and the program run in a "batch mode". To accomplish this, the DSS file name and commands are placed into a file. Comments may also be placed in the input file by using an asterisks (*) at the beginning of the comment line. To execute the program with an input command file, enter:

```
DSSUTL INPUT=file-name
```

The DSS file and output file may also be designated on the execution line by specifying "DSSFILE=dss-file", and "OUTPUT=output-file".

On MS DOS personal computers DSSUTL must reside in a directory that is set in your path (e.g., \HECEXE), or reside in your current directory, otherwise an error of "Overlay not found" will occur.

1.2 Command Structure

A typical command for DSSUTL has the following structure:

```
Command.options, parameters
```

If a command uses an input or output file, and that file has not been specified in an earlier command, the structure is:

```
Command.options, TO=file-name, parameters
```

Command options (if used) follow the command and are separated by a period (.) and no spaces. If a file name is given, it follows any options separated by a comma and/or blanks. Once a file has been designated, it typically does not need to be given again. A comma and/or blanks separate any parameters from the command (or file name).

The last pathname accessed, and the time window (if set) are stored in a short-term memory. They may be displayed by the STATUS command. The time window will automatically be used by most commands, if it has been set.

Informative output for a single command may be suppressed in DSSUTL by preceding that command with a period (.) For example:

```
.RN C=FLOW
```

will suppress informative output during that command. The verify command will suppress informative output for all commands.

All commands, parameters and pathnames may be given in either upper or lower case. On UNIX computers, the case of filenames are retained.

Chapter 2

Differences from Previous Versions

The following is a brief list of the modifications made since the July 1990 version. For information on a particular change, refer to the documentation for the specific command.

2.1 New Commands

The following commands have been added:

- FC -- File Check
- Import - Export Commands:
- EF -- Exchange Format
- EV -- Exchange Variable
- EX -- Export Data
- IM -- Import data

2.2 Additional Command Capabilities

1. The selective catalog has been enhanced to select pathnames from a (non-abbreviated) catalog based on their records last written date or program name. This, for example, would provide a direct means for archiving records that were last written prior to a given date:

```
AR TO=archive.dss LW<20JAN92
```

2. The condensed catalog has been changed so that records that differ only by the date will occupy one line, even if there are missing records within the time span. The D (Date Span) option will override this and show only complete data sets (with regards to time) as a single line. For example, if there is 50 years of hourly records for a data set, and one or more records within that span are missing, that data will show up as two separate lines in the condensed catalog, if the "D" option is specified. This option must be used in combination with the "C" and the "N" options.

(This page intentionally left blank)

Chapter 3

General Commands

3.1 Archive

Use:

```
AR.option, TO=archive-file, parameters
```

The archive command copies records from the current DSS file to the archive file, then deletes those records from the current file. The archive file name, which is specified by "TO=" followed by the file name, only needs to be given on the first archive command (the same archive file will be used on subsequent archive commands). If the archive file is not given, the user will be prompted for it. The archive command operates on a record basis; any time-window set is ignored.

For time series data the "D" (Date) part of the pathname may reference an offset from the current date. For example, to archive hourly data that is three months old, the following command may be given:

```
AR TO=DSSOBS D=M-3M E=1HOUR
```

This command will archive all records with a pathname E part of "1HOUR", and a D part containing the current month minus 3 months. For example, if today is February 22, 1970, the "D" part would be "01NOV1969".

Options:

1. S (Squeeze) - This option squeezes the current file after the archive has been completed. This option should only be used on the last archive command for a file.
2. P (Preserve) - Records with the same pathname in the file being copied to will not be overwritten.
3. C (Compress) - Time series data is compressed according to the file compression method specified in the copy file (a time window does not need to be set). If a method is not set to match that pathname, it will be uncompressed.

Parameters:

1. None - Uses the pathname set in memory.
2. Pathname - If a pathname follows the command (or archive file name), that record is archived.
3. Tag(s) - Uses the record(s) corresponding to the tag(s) specified for archiving.
4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file for archiving.
5. Date Reference (D=...) - Uses the current catalog and selects records to archive based upon a "D" part relative to the current date. The "D=" must be followed either by a D (for the current day), a M (for the current month), or a Y (for the current year), a minus (or plus) sign, a number, then another D, M, or Y. For example:
D=M-2M (The current month minus two months)
D=D-60D (The current day minus 60 days)
The time interval (E part) is usually specified along with this parameter.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to archive. For more information, refer to the selective catalog section.

Examples:

```
AR TO=DSSARC A=BIG BASIN, D=M-3M, E=1HOUR
AR NATP-F, NATP-P
AR.S A=AFOS, D=Y-2Y
AR TO=ARCHIVE.DSS LW<01JAN92
AR PROG=DSSTS
```

3.2 Catalog

Use:

CA.options, parameters

The catalog command displays either the catalog, an abbreviated catalog or the condensed catalog. The catalog is a list of the record pathnames in the DSS file, along with their last written date and time and the name of the program that wrote that record. The catalog is usually sorted alphabetically by pathname parts. Each pathname has a record tag and a reference number, either of which may be used in place of the pathname in most of the commands. The tag(s) are semi-permanent but not necessarily unique. The reference numbers are unique, and are not permanently associated with the pathnames - they may change each time a new catalog is generated.

The name of the catalog file is the name of the DSS file with an extension of ".dsc". If the catalog file does not exist, it will be created. If the DSS file has changed, a new catalog must be created (.N option) to obtain the current list of pathnames in the DSS file.

The abbreviated catalog contains only the list of pathnames, their tag(s) and reference numbers. It can be displayed from a regular catalog (CA.A), or the catalog file itself can be made abbreviated (CA.AN).

The condensed catalog is a special catalog designed for time series data. The parts of the pathname (instead of the full pathname) are listed along with the date span for records where only the "D" (date) part varies. Repeating parts are replaced by dashes for easier reading. The condensed catalog is placed in a separate file named similar to the catalog, except it has an extension of ".dsd". The condensed catalog can only be created (or updated) when a sorted catalog is generated with the default sort order.

Options:

1. None - The catalog file is displayed one screen at a time for an interactive session. If DSSUTL is being run from a batch job, or a macro, the complete catalog is displayed.
2. A (Abbreviated) - This option will cause only the pathnames and their tag(s) and reference numbers to be displayed from the catalog (similar to the Display Pathnames command). This option is recommended for 80 column screens. If a new catalog is created, then the catalog file itself will be abbreviated.
3. C (Condensed) - This option will display the condensed catalog. If the condensed catalog does not exist, it will be created.

4. D (Date Span) - Normally, for a condensed catalog, if a record set has an incomplete data span (missing records in time), that catalog line is marked with an asterisk. This option will instead cause a separate catalog line for each time span. This option must be used with the "C" and "N" options.
5. F (Full) - In an interactive session, this option displays the full catalog without pausing after each screen. The full catalog is automatically displayed in a batch job.
6. M (Map File) - When generating a new catalog, a "map" file will be created using an extension of ".dsm". A map file contains only pathnames, which is useful for creating input files for programs that use pathnames. The selective catalog feature may be used to limit which pathnames are listed.
7. N (New) - This option generates a new catalog. This option should be used after records have been added, deleted or renamed in the DSS file.
8. P (Printer) - This options sends the catalog file (or condensed catalog) to the line printer. On MS DOS personal computers, the printer should be ready before this option is used. The catalog file is also displayed on the screen, unless the ".S" option is also used.
9. U (Unsorted) - This option will cause the catalog not to be sorted when generating a new catalog. An unsorted catalog can be created much faster than a sorted one.
10. S (Suppress) - This option will cause the catalog file not to be displayed on the screen.

Parameters:

1. None - If a new catalog is created, the default sort order is used. This sort is alphabetical by pathname parts in the order A,B,C,F,E,D.
2. Order - The pathname part sort order may be specified when creating a new catalog. This is done by giving an "O=", followed by the part letter order. The part letters must not be separated, and those parts not specified are filled in using the default order. For example:

```
CA.N O=FB
CA.N O=FBACED
```

would generate the same sort order. Do not specify a sort order when generating a condensed catalog.

Remarks:

On MS-DOS computers, the catalog may be sorted by either the DOS sort utility, or an external sort program, named "GNUSORT.EXE", which uses extended memory. GNUSORT.EXE uses a memory extender named DOS4GW.EXE. Both of the extended memory programs must reside in the directory \HECEXE. Approximately 550 Kbytes of free conventional memory are required for the extended memory sorting. (The MS-DOS memmaker program will often help recover this memory.)

Examples:

CA	(Displays the catalog file)
CA . N	(Creates a new catalog, then displays it)
CA . CN	(Creates and displays the condensed catalog file)
CA . A	(Displays pathnames, their tag(s) and reference numbers only)
CA . NPS	(Generates a new catalog and sends it to the printer - the catalog is not displayed on the screen)
CA . N O=FB	(Creates a new catalog, sorting the F parts of the pathnames first, then the B parts of the pathnames)

3.3 Check

Use:

CH, parameters

The Check command determines if record(s) exists in the DSS file. If the record exists, various information about that record is displayed. If the record has been compressed, the amount of spaced saved is shown.

Parameters:

1. None - Uses the pathname set in memory.
2. Pathname - If a pathname follows the command, that record is checked.
3. Tag(s) - Checks the record(s) corresponding to the tag(s) given.
4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2 and N3 through N4 from the current catalog file to check.
5. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to check. For more information, refer to the selective catalog section.
6. ALL - Checks all records in the current catalog file.

Examples:

CH	(Checks the current pathname in memory)
CH NATP-F	(Checks the record with a tag of "NATP-F")
CH 2, 4, 6-9	(Checks records for pathname numbers 2, 4, 6, 7, 8, and 9 from the catalog file)

3.4 Copy

Use:

`CO.options, TO=copy-file, parameters`

The copy command copies records from the DSS file opened to the DSS file whose name is designated by "TO=copy-file" following the command, or it will be prompted for if not supplied. If the DSS copy-file does not exist, it will be created. Once the copy-file name has been given, it does not need to be given for other copy commands in the same session.

For time series data, if a time window is set, then only data within that time window will be copied. If a time window is not set, the "D" (Date) part of the pathname may reference an offset from the current date. For example, to copy hourly data that is three months old, the following command may be given:

```
CO TO=DSSOBS D=M-3M E=1HOUR
```

The above command will copy all records with a pathname E part of "1HOUR", and a D part containing the current month minus 3 months. For example, if today is February 22, 1970, the "D" part would be "01NOV1969".

Options:

1. None - Any records with the same pathname in the file being copied to will be overwritten. For regular-interval time series data, unless a time window is set, the record's data compression is not changed.
2. P (Preserve) - Records with the same pathname in the file being copied to will not be overwritten.
3. C (Compress) - Time series data is compressed according to the file compression method specified in the copy file (a time window does not need to be set). (If a method is not set to match that pathname, it will be uncompressed).

Options for Regular-interval Time series Data:

Note: A time window must have been specified to use these options. The options are mutually exclusive and are only useful when copying data to another file with the same pathnames. They are infrequently used.

4. M (Missing) - Replace Missing data flags only. Only -901's (or non-existent records) will be replaced in the copy file.
5. U (Update) - Will not allow missing data flags (-901's) in the originating file to replace valid values in the copy file.

6. R (Regardless) - Copies all data within the time window given, including non-existent records or all missing data. Normally, if the data to be stored for one record is all missing (-901), that record is not written. The R option bypasses this, and will allow a record of all -901's to be copied.

Option for Irregular-Interval Time Series Data:

7. M (Merge) - If data for the same record already exists in the copy file, the M option will merge the two sets together. For example, if data in the copy file exists for Noon, 2 and 4 p.m., and data in the originating file is for 1, 3, and 5 p.m., the M option will cause the final data set to be for Noon, 1, 2, 3, 4, and 5 p.m.. Normally data is replaced.

Parameters:

1. None - Uses the pathname set in memory.
2. Pathname - If a pathname follows the command (or copy file name), that record is copied.
3. Tag(s) - Copies record(s) corresponding to the tag(s) given.
4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file for copying.
5. Date Reference (D=...) - Uses the current catalog and selects records to copy based upon a "D" part relative to the current date. The "D=" must be followed either by a D (for the current day), a M (for the current month), or a Y (for the current year), a minus (or plus) sign, a number, then another D, M, or Y. For example:
 D=M-2M (The current month minus two months)
 D=D-60D (The current day minus 60 days)
 The time interval (E part) is typically specified along with this parameter.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to copy. For more information, refer to the selective catalog section.
7. ALL - Copies all records in the file (does not use the catalog file). This is useful in merging two or more DSS file together. Any time window specified is ignored.

Examples:

```
CO TO=LHDSS D=Y-1Y, E=1YEAR, F=OBS
CO NATP-F, NATP-P
CO C=FLOW
CO LW>20JUN92
CO PROG=HEC1
CO TO=NEWFILE ALL
```

3.5 Data Compression

Use:

DC, parameters

The data compression command will display or set a file's default data compression methods for regular-interval time series data. This mode of setting data compression parameters is based on matching pathname parts for new data to be stored in the file. Data already stored may be re-compressed by the "C" option in the squeeze command. Compression methods may also be designated by the program that stores the data. Methods for as many pathname part specifiers as needed may be set in a DSS file.

Three methods of data compression are available. They are: 1) The repeat method, which flags repeated values; 2) The delta method, which stores the offset for each value from a base value, and; 3) The significant digits method, which stores only three significant digits for each data value. In addition, the repeat method and delta method can be used together, and the repeat method and the significant digits method can be used together. A complete description of data compression may be found in the DSS overview description.

The delta method requires a precision exponent parameter indicating the accuracy of the data. If the data to be stored is measured to the nearest hundredth (0.01) (e.g., precipitation), the precision exponent would be -2, to the nearest thousandth, the exponent would be -3.

In addition, a "base value" and "data size" parameter may be specified for the delta method. These parameters are typically only used with "real-time data", data that is being updated frequently, and only to increase the efficiency of storing future data. The base value is the expected minimum value that the data will obtain for that record. For example, the base value for incremental precipitation would be 0.0. The data size parameter indicates whether one or two bytes should be pre-allocated for each data value. One byte allocates a difference of 256 units, two bytes allocates a difference of 65,536 units. Typically, hourly precipitation would pre-allocate only one byte (up to 2.56 inches per hour), whereas reservoir elevations would pre-allocate two bytes (up to 65.536 feet difference). If the data changes so that either of the selected values are invalid, the software will automatically select new values (and re-compress the data). If these parameters are not specified, the software will automatically select values based upon the data.

If required parameters are not given, the user will be prompted for them. To remove a compression method, specify "NONE" as the method.

Parameters:

1. ? - The current default file data compression methods are displayed.

2. No parameters - The file's current default file data compression methods are displayed, then pathname part and data compression parameters are prompted for.
3. Pathname parts - One or more pathname parts to compare against the pathnames of new data for this type of compression. The part to match is specified by the part letter, followed by an equal sign then the part. Usually the "C" part (data type) is specified. The "at sign" (@) can be used as a wild character at the end (only) of a portion of a part, so that only the first piece is matched. For example, "C=FLOW@" would match pathnames with a C part of "FLOW", "FLOW-RES OUT", and "FLOW-NATURAL".

a) Method - The compression method is defined by "METHOD=" (or the abbreviation "ME="), then the method (which can be abbreviated to the first three characters) or the method number. The possible methods and their numbers are:

0. NONE (removes compression method)
1. REPEAT
2. DELTA
3. REPEAT+DELTA
4. SIG. DIGITS
5. REPEAT+SIG.

b) Precision Exponent (required for the delta method) - The precision exponent for the delta method is identified by "PREC=", then the exponent. For example to set the precision to thousandths, this parameter would be "PREC=-3". If this parameter is not included, it will be prompted for.

c) Base (optional, only applies to the delta method) - The base value is indicated by "BASE=" followed by the base number.

d) Size (optional, only applies to the delta method) - The pre-allocation data size is specified by "SIZE=" followed a "1" to allocate one byte per value, or a "2" to allocate two bytes per value.

Examples:

```
DC ?
DC C=PRECIP@, METH=REPEAT+DELTA, PREC=-2
DC C=TEMP-WATER METH=5 BASE=32.0
DC C=FLOW@, F=OBS METH=3
DC C=STAGE, METH=DELTA, BASE=0.0, SIZE=2
DC C=ELEVATION, METHOD=NONE
```

3.6 Delete

Use:

DE, parameters

The delete command eliminates records from the DSS file. A record is not physically removed from the file until the "squeeze" command is given. Because of this feature, deleted records may be recovered by use of the "undelete" command. Refer to documentation on the undelete command for further information.

Parameters:

1. None - Uses the pathname set in memory.
2. Pathname - If a pathname follows the command, that record is deleted.
3. Tag(s) - Deletes record(s) corresponding to the tag(s) given.
4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file to delete.
5. Date Reference (D=...) - Uses the current catalog and selects records to delete based upon a "D" part relative to the current date. The "D=" must be followed either by a D (for the current day), a M (for the current month), or a Y (for the current year), a minus (or plus) sign, a number, then another D, M, or Y. For example:
D=M-2M (The current month minus two months)
D=D-60D (The current day minus 60 days)
The time interval (E part) is typically specified along with this parameter.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to delete. For more information, refer to the selective catalog section.

Examples:

```
DE 2-5, 7, 9
DE NATP-F, NATP-P
DE F=COMPUTED
DE PROG=HEC1
DE LW<08FEB94
```

3.7 Display Pathnames

Use:

DP.options, parameters

The display pathname command lists pathnames, their tag(s) and reference numbers from the catalog file (similar to an abbreviated catalog). If the catalog file does not exist, it will be created.

Options:

1. None - Displays one screen of pathnames at a time. At the end of the screen, the user may enter a new command, or press a carriage return to continue the display. All the pathnames in the catalog are displayed in a batch or macro mode.
2. F (Full) - In an interactive session, this option displays all pathnames in the catalog without pausing after each screen.
3. N (New) - This option generates a new catalog before displaying the pathnames.
4. U (Unsorted) - This option will cause the catalog not to be sorted when generating a new catalog. An unsorted catalog can be created much faster than a sorted one (the default is to sort).

Parameters:

1. None - The pathnames are displayed, beginning with the first one.
2. N (N is a number) - Displays pathnames beginning with reference number N. If N is greater than the number of pathnames, then the last twenty pathnames are displayed.
3. N1-N2 - Displays pathnames with reference numbers N1 through N2.
4. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities to display pathnames based on their parts. For more information, refer to the selective catalog section.

Examples:

```
DP 9999          (displays the last 20 pathnames)
DP.F C=FLOW
DP LW>20MAR92
```

3.8 Duplicate

Use:

DU, parameters

The duplicate command duplicates records, giving the new records different pathnames. After giving the duplicate command and the associated parameters, DSSUTL will display a list of the pathnames to be duplicated, then prompt the user for new pathname part(s) or a new pathname for the new record. New parts are specified by entering the part letter followed by an equal sign then the part. If you change your mind, and you decide not to duplicate the data, the word "QUIT" may be entered to return to the main program. A time window may be specified to duplicate time series data.

Option:

1. P (Preserve) - Records with the same pathname in the file will not be overwritten.

Parameters:

1. None - Uses the pathname set in memory.
2. Pathname - If a pathname follows the command, that record is duplicated.
3. Tag(s) - Duplicates record(s) corresponding to the tag(s) given.
4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file to duplicate.
5. Date Reference (D=...) - Uses the current catalog and selects records to duplicate based upon a "D" part relative to the current date. The "D=" must be followed either by a D (for the current day), a M (for the current month), or a Y (for the current year), a minus (or plus) sign, a number, then another D, M, or Y. For example:
D=M-2M (The current month minus two months)
D=D-60D (The current day minus 60 days)
The time interval (E part) is typically specified along with this parameter.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to duplicate. For more information, refer to the selective catalog section.

Example:

```
DU C=FLOW, F=OBS  
F=OBS-RAW
```

(Duplicate all observed flows giving the records a new "F" part.)

3.9 Edit

Use:

ED.options, parameters

The edit command provides a means of editing data in the DSS file. The data is written in a formatted form to a temporary scratch file, edited by an external editor, then read back in and stored in the DSS file. Several records may be edited at one time.

DSSUTL will use the COED program as the editor, unless the "A" option is specified. Normal COED commands are used to edit the data. When complete, the user enters FILE to save the revisions (or QUIT to disregard the revisions), which transfers control back to DSSUTL. DSSUTL then reads the data from the edit file and stores it in the DSS file. If the file was not modified (i.e., the user entered QUIT), the data will not be read back in. The data in the edit file is read in a free format style.

For regular-interval time series data, the first part of each line is used for the date and time of the first piece of data in that line. This date and time is for informational purposes only, and is **ignored** when the data is read in. The starting date and time printed in the header (just below the pathname) is used to determine the date and time of the first piece of data. The date and time of all other data is implied by its position from the first piece of data (you cannot just add or delete data anywhere). To delete data, replace the data value with the letter "M" or a -901. (including at the end of the data set). Missing data at the beginning and end of the data set are suppressed, unless either a time-window was specified (via the TIME command), or the "C" option is given.

Irregular-interval time series data may be deleted anywhere within the data set, including at the beginning or end. If data is to be added or deleted from a paired data set, the number of ordinates shown in the header must also be modified to reflect any changes.

If no format was specified (using the FORMAT command), the program will select one based on the magnitude of the data.

Options:

1. None - COED is used as the editor.
2. A (Alternative editor) - When this option is specified, the user is transferred into the computer's operating system. At this point, the selected editor should be used to edit the file specified by DSSUTL. At the end of the edit session, control should return to DSSUTL. On MS DOS personal computers, the user must type the word "EXIT" to return to DSSUTL.
3. N (No Quality Flags) - If the record has data quality flags associated with it, this will cause the flags not to be edited (otherwise they will be shown on a separate line in a hex format).

Options for Regular-Interval Time series Data:

4. C (Complete) - All data within the record, including leading and trailing missing data (-901's), are written to the edit file. This option ignores any time window set.
5. D (Data only) - If a time window has been set, then this option causes missing data at the beginning and ending of the window not to be edited. If no time window is given, this option is automatically used.

Parameters:

1. None - Uses the pathname set in memory. For time series data, if a time window has been set, the data within that time window are edited
2. Pathname - If a pathname follows the command, that record is edited.
3. Tag(s) - Edits data for the record(s) corresponding to the tag(s) given.
4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file for editing. Any time window set will be adhered to if the data is time series.
5. Date Reference (D=...) - Uses the current catalog and selects records to edit based upon a "D" part relative to the current date. The "D=" must be followed either by a D (for the current day), a M (for the current month), or a Y (for the current year), a minus (or plus) sign, a number, then another D, M, or Y. For example:
D=M-2M (The current month minus two months)
D=D-60D (The current day minus 60 days)
The time interval (E part) is typically specified along with this parameter.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to edit. For more information, refer to the selective catalog section.

Examples:

```
ED 2, 3, 5-9
ED NATP-F, NATP-P
ED.C C=FLOW
ED.A B=NORTH SHORE
```

3.10 Exchange Format

Use:

`EF.options, format`

The exchange format command defines the format to use for both the export and import commands. The format line indicates which and how exchange variables are to be imported or exported, their dates and times, and pathname parts. The format set will be displayed if a question mark follows the command (i.e., "EF ?").

The exchange format may be up to 256 characters in length. Because long lines are difficult to edit, two forward slashes (//) at the end of the format line indicate that the next line is a continuation of the current format line. DSSUTL will delete the two slashes before appending the next line (which should begin without a command). As many continuation lines may be used as needed, as long as the exchange format does not exceed 256 characters.

Although the exchange format is used for both the import and export commands, the use of the format may be different for these uses. Additional information concerning the exchange format may be found in Chapter 6.

Options:

1. None - The primary format for both the export and import command is defined (unless a question mark follows the command).
2. H - The header (title) format for the export command is defined, or instructions on how to skip a header in the import file is given.
3. M - Indicates how missing data values should be printed in the export file, or what string constitutes missing data in the import file.

Export:

DSSUTL essentially copies the exchange format to the export file, replacing exchange variables (enclosed in square brackets "[]") with data and any reserved variable with the appropriate information. All characters that are not recognized are copied "as is". Thus, placing quotes around a pathname part reserved variable (on the outside of the brackets), and putting commas between each field will produce an ASCII delimited file. Spaces and all other character are copied as is to the export file.

Export Exchange Variables

Exchange variables, which may be located anywhere on the exchange format line, must be enclosed within square brackets "[]" (note that the variable names are not enclosed in brackets when defined with the EV command). An example of such an exchange variable might be "[STG]". If a certain output format is required (e.g., the number of decimal places needed), that format may be specified within the brackets by following the variable name with a colon (:) and the required format. The format for floating point numbers is *columns.precision*, where *columns* is the total number of columns the number will occupy, and *precision* is the number of digits to the right of the decimal to show. Decimal numbers may be printed with a format of *Icolumns*. Examples of valid formats are:

[STG:.3]	Stage with a precision of 3 digits to the right of the decimal will be printed.
[STG:12.3]	The number will occupy 12 columns, with 3 digits to the right of the decimal printed (blanks will occupy unused columns to the left). This format will cause numbers to line up.
[STG:I6]	Print the number as an integer value, occupying 6 columns.

If no format is given, two digits to the right of the decimal are printed.

Export Reserved Variables

The date, time, and pathname parts for data values may be printed in the export file by use of a reserved variable. Reserved variables must be enclosed in square brackets ([]). The reserved variables for exporting are:

```
[APART]
[BPART]
[CPART]
[DPART]
[EPART]
[FPART]
[DATE]
[TIME]
[T:n]      (where n is a column number to tab to)
```

The pathname parts to be printed in the export file are for the data value following that variable. For example, in the export format

```
[APART] -- [BPART], [STG1]; [BPART], [STG2]; [BPART], [STG3]
```

"[APART]" will be replaced with the A part of the pathname for the exchange variable STG1, and the "[BPART]" will be replaced with the B part of the pathname for the variable following that reserved variable.

A pathname part can be explicitly printed by following the part identifier with a colon and the exchange variable name for that part, all within the square brackets. For example:

```
[APART:STG2] -- [BPART:STG1], [STG1]; [BPART], [STG2]; [BPART], [STG3]
```

The string "[APART:STG2]" will be replaced with the A part of the pathname for the exchange variable STG2.

The "[DATE]" and "[TIME]" reserved variables will be replaced with the date and time for the data on that line in the export file. The default date style used is a nine character military style date (e.g.,07JAN1991). The default time style is a 24 hour clock time (e.g., 1630), with midnight reported as "2400" (not "0000").

A date and time style may be defined by following the word DATE or TIME with a colon and a "picture" of the date or time. The picture consists of letters that are replaced with parts of the date or time. For example, the variables

```
[DATE:Www, Mmm D, YY] [TIME:H:MM am/pm]
```

might be replaced with

```
Mon, Jan 7, 91 2:00 pm
```

The valid strings for the date picture are:

<u>String</u>	<u>Replaced with</u>
D	The day of the month, composed of one or two digits (e.g. "7")
DD	The day of the month, composed of two digits (e.g., "07")
M	The month number, expressed as one or two digits (e.g., "5" for May)
MM	The month number, expressed as two digits (e.g., "05")
MMM	The three character abbreviation of the month (e.g., "JAN"). Lower case and upper case characters are respected (e.g., Mmm will be translated to "Jan")
YY	The year, expressed as two digits (e.g., "91")
YYYY	The year with the century included (e.g., "1991")
W	The day of the week, expressed as a single digit (with 1 as Sunday)
WWW	The three character abbreviation of the day of the week (e.g., "TUE"). Lower case and upper case characters are respected (e.g., Www will be translated to "Tue")
JJ	The julian day of the year (e.g., " 32" is February 1)
JJJJ	The julian day since Jan 1, 1900 (e.g., "33269" represents Feb. 1, 1991)

The valid time strings are:

<u>String</u>	<u>Replaced with</u>
H	The hour of the data, composed of one or two digits.
HH	The hour of the data, composed of two digits.
MM	The minutes portion of the time, composed of two digits.
SS	The seconds of the data (set to zero)
AM/PM	If this string is present, the hours are reported in twelve hour clock time, and the appropriate characters are printed. If this string is not present, the hours are printed in 24 hour time. The AM/PM string may be in lower case, and may contain periods. (e.g. "a.m./p.m.")

The date and time pictures may contain other characters as desired. These other characters often include commas, blanks, colons, slashes, etc..

Moving to an explicit column is accomplished by using the tab variable. The tab variable is "[T:n]", where n is the column number to tab to. The tab variable will cause the character following the specifier to be in the column given (regardless if that character is part of a data value, a blank, or any other character). (A tab character will be used if a real tab character is inserted in the format.)

Export Header Format

A single header (or title) line can be written to the export file by providing a header format. The header format follows the same conventions as the regular exchange format, and is specified by using the "H" option with the EF command (e.g., "EF.H"). Pathname part, date, time and tab reserved variables may all be given as discussed previously. However, the pathname part reserved variables should explicitly identify the exchange variable (e.g., "[BPART:STG1]"). A date or time reserved variable will use the date and time of the first data value. Exchange variables should NOT be given in the header format. An example header format command might be:

```
EF.H [T:5]Lake Report for [BPART:ELEV], on [DATE:Mmm d, YY] //
[T:40][CPART:ELEV] [T:50][CPART:INFLOW] [T:60][CPART:OUTFLOW]
```

Export Missing Data Definition

If the "M" option is used (e.g., "EF.M"), the character string following the command will be printed for missing data (instead of the number -901). The string will be centered within the space normally occupied by the number. Examples of missing data definitions are:

```
EF.M -
EF.M Miss
```

Import

Lines read with the import command are interpreted in fields, delimited by commas and (or) blanks. Pathname parts that might contain blanks should be enclosed in quotes (single or double) in the import file. If the parts do not contain blanks, quotes are not necessary.

The items that may be specified on the Exchange format line for importing data are:

```
[exchange variable]
[APART]
[BPART]
[DATE]
[TIME]
[SKIP]
```

Exchange variables do not have format styles associated with them, as used in the export command.

The A part and the B part may be read from the import file. Only one A part and B part may be specified in the format line. These parts will override the parts given in exchange variable pathname. Reading A and B parts from the import file is useful when importing a file with sections of data from different areas (e.g., an import file containing all the lake reports in a basin). If an import file has different B parts on the same line, the B parts must be skipped, and the data identified with different exchange variables (that have pathnames with the correct B part).

If desired, the date and time of the data can be read from the import file, if the file contains both the date and time. If the import file does not contain dates and times (or you do not wish to use those), and the data is regular-interval time series data, the date and time of the first data value must be identified with the DSSUTL TIME command. If the date field contains blanks (e.g., "Jan 7, 1990"), it should be enclosed in quotes. A date and time field must be given on each import line for data to be stored as irregular-interval time series data. (The E part of the pathname identified by the Exchange Variable command identifies whether the data is regular interval or irregular-interval).

All other information in the import file that will not be imported, must be identified by the [SKIP] variable. The [SKIP] variable indicates that the field is unused, and should be ignored by DSSUTL. A field is a series of characters, separated by a comma and/or blank. If quotes surround several strings, all the characters within the quotes should be treated as a single field. Several fields can be skipped with a single [SKIP] by following the "P" with a colon and the number of fields to skip (e.g., [SKIP:n], where n is the number of fields).

Example:

If the following line is typical in a file to be imported:

```
COE "SOUTH LAKE" AM REPORT: 20MAY90, 0800; ELEV: 342.123, PRECIP: 0.15
```

The following exchange format might be given to import that file:

```
EF [SKIP] [BPART] [SKIP:2] [DATE] [TIME] [SKIP] [ELEV] [SKIP] [PREC]
```

Skipping Import Header Lines

If the import file contains lines that are not to be imported, the header exchange format can be used to indicate which lines to ignore. To search the import file for the line to begin importing data on, follow the header exchange format with the word LOCATE, then the string to search for exactly as it appears in the file. For the example above, the following command can be used:

```
EF.H LOCATE AM REPORT
```

(note that there are no quote marks around the string AM REPORT).

If a known number of lines at the top of the import file are to be ignored, follow the exchange format header command with the word SKIP, then the number of lines to skip. For example, to skip 12 lines at the top of the file, use the command:

```
EF.H SKIP 12
```

Also, lines with a specific character string will be ignored if that string follows the word SKIP. For example, to ignore all lines that contain the string "Daily Status" (assuming that this is a title in the import file), use:

```
EF.H SKIP Daily Status
```

(again, note that there are no quotes around Daily Status.)

Defining Missing Import Data

If the import file contains a flag for missing data (such as a string of dashes), the missing data exchange format can be used to define that flag so that the DSS missing data flag (-901) will be used for that value. The exact string that flags missing data should follow the EF.M command. For example:

```
EF.M ---  
EF.M -9999
```

In both of these example, a -901 value will be used instead of dashes, as in the first example, or instead of the number -9999 as in the second example. Only one missing data flag can be defined for an import file.

3.11 Exchange Variable

Use:

```
EV.option name=pathname, UNITS=units, TYPE=type
```

The exchange variable command sets a user defined short name to be equivalent to a pathname for exporting or importing data. The variable name may then be given in the exchange format (EF command), enclosed in square brackets ([name]). The exchange variable is nothing more than an abbreviation for a pathname.

The variable name can be up to eight characters long, and cannot contain any commas or blanks. An equal sign (=) must follow the variable name, then the pathname (or pathname reference) that is assigned to that name. If data is to be imported, the string "UNITS=units, TYPE=type" should follow the pathname with the appropriate units and type of the data. If data is exported, the units and type are ignored.

Up to 50 exchange variables can be declared at any time, and an exchange variable may be redefined. However, memory is equally divided among all exchange variables, so it may be prudent to clear variables (with the .R "reset" option) if a new set of variables will be defined.

The exchange variables and their associated pathnames defined may be displayed by entering a question mark following the EV command (e.g., "EV, ?"). Additional discussion on exchange variables may be found in Chapter 6.

Option:

1. R - Resets and clears all exchange variables

Parameters:

1. ? - Displays a list of all exchange variables defined.
2. EV *variable-name=pathname-reference* [UNITS=*units*, TYPE=*type*]

Valid pathname references include:

- a. The pathname itself.
- b. Pathname parts identified by the part letter and an equal sign (e.g., B=NATP, C=FLOW). The other pathname parts are obtained from the last pathname referenced (which could be in a previous exchange variable).
- c. The record's tag.

d. The pathname's catalog reference number.

Because the set of commands to import and export data are often defined in an input file or PREAD macro file, the first two procedures are preferred over the last two. The "E part" of the pathname determines whether the data is regular-interval or irregular-interval time series data.

Examples:

For exporting data:

```
EV FLOW=/ALLEGHENY/NATP/FLOW/01JAN1990/1HOUR/OBS/  
EV STG=C=STAGE  
EV PREC= C=PRECIP-INC, E=IR-MONTH  
EV DO=24 (gets pathname 24 from the catalog file)
```

For importing data:

```
EV STG1=/ALLEGHENY/NATP/FLOW/01JAN1990/IR-MONTH/OBS/ UNITS=FEET TYPE=INST-VAL  
EV STG2= B=PITT, E=1HOUR, UNITS=FEET TYPE=INST-VAL  
EV STG3= B=FRKP, E=1DAY, UNITS=FEET, TYPE=PER-AVER
```

3.12 Export

Use:

`EX.option, export-file`

The export command causes data to be exported to a file whose name follows the export command. If the export file does not exist, it will be created. If the file name is not supplied, DSSUTL will prompt the user for it.

The export command uses the exchange format and exchange variables previously defined. The data is read from the DSS file when the export command is issued (not when the exchange variable command is given). Typically, a time window is designated prior to the export command. This time window applies to all pathnames given. For more information on exporting data, refer to Chapter 6.

Option:

1. None - Exported data is written to the beginning of the file.
2. A - (Append) Exported data is written starting at the end of the file.

Parameters:

1. None - The export file name will be asked for. If data was just exported (in the same session), then this export will append to the end of the file.
2. file-name - If a file name is given, the data will be exported to that file. If the file does not exist, it will be created.
3. * - If an asterisk is used in place of a file name, the data will be printed on the screen (standard out). This is a good way to check the export format before exporting the data to a file.

Examples:

```
EX walter.dat           (exports data to file "walter.dat")
EX.A lehigh.dat        (appends data to file "lehigh.dat")
EX *
```

3.13 File Check

Use:

FC

The file check command searches the entire DSS file currently opened for any errors in the internal address tables. Any errors or inconsistencies in the internal tables are reported. Because an exhaustive search of the file is made, this procedure will require a significant amount of time (about the same amount of time required for a squeeze).

The DSS software has been designed with a high regard for file integrity. Generally a DSS file should not be subject to damage, even from a system crash or power outage. However, some of today's operating systems buffering schemes to improve computer performance can defeat some of the DSS safe guards in the event of a crash. If you experience a crash, check any DSS files in use at that time with this command as soon as possible. Most all data can be recovered from a damaged file by the SQUEEZE command.

If you find any errors in a DSS file that were not the result of a system crash or similar failure (on DOS use CHKDSK to search for lost links), make a backup copy of that file. Then notify HEC of the error and any conditions relevant to use of that file (e.g., what programs wrote to the file, is it on a network drive, etc.). The SQUEEZE command can be used to recover data from the file once a backup copy has been made.

3.14 Finish

Use:

FI

The Finish command terminates the execution of DSSUTL.

3.15 File Query

Use:

FQ

The file query command displays attributes of the current DSS files opened, including the file name, DSS version, number of records in the file and the file size.

3.16 Format

Use:

```
FO, (format)
```

The format command sets an output format to use for the write data , edit, or tabulate commands. The format follows the command name and must be a valid FORTRAN format of up to 40 characters. The format can be used to indicate how many data values should be printed on a line, the number of significant digits of the data, or a way of adding other alphanumeric information to a data line. If no format has been set, DSSUTL will chose a format based on the magnitude of the data. To clear a format (and cause DSSUTL to select one), enter just the format command (with no format following).

The format command may be used to produce data lines to be used as input for a program that does not have DSS capabilities. For example, if "IN" records are needed, the following format may be appropriate:

```
FO ( 'IN' ,F6.1 ,9F8.1 )
```

The Write Data command is used to write the data to an output file. This file must be edited to remove the pathname, header and END DATA lines.

When regular-interval time series data is tabulated or edited (not the Write Data command), there may be only one format specified (e.g., (8F10.4), not (2F12.2,4F10.1)). The number of data values per line may be reduced by the program in order to make them fit within the 80 column limit.

Examples:

FO (F12.2)	(One data value per line)
FO (6F12.4)	(Increases the number of significant digits)
FO (T20,6F8.0)	(Tabulate to column 20, then print the data)
FO ('FLOWS' ,2X,6F10.1)	(insert the word "FLOWS" at the beginning of each line)
FO	(Clears previously set format. The format used is selected by DSSUTL based on the magnitude of the data.)

3.17 Help

Use:

`HE.option, parameters`

The help command causes information about a command to be displayed. Entering just HELP provides a list of commands.

Options:

1. None - A short description of the command and its options are displayed.
2. A (All) - Displays all help information without paging. Normally, help information is displayed on the screen one page at a time.
3. E (Extended) - A comprehensive description of the command and its options are displayed.

Parameters:

1. None - A list of the available commands is displayed.
2. Command - A description of that command is displayed.
3. USAGE - A description of how to use help and commands is displayed.
4. SC (Selective Catalog) - Documentation on the selective catalog is displayed.
5. EX-IM - Describes the export-import capability.

Examples:

<code>HE</code>	(A list of commands is shown)
<code>HE .E ED</code>	(A complete description of the edit command is displayed)

3.18 Import

Use:

IM, import-file

The import command causes data to be read in from the file whose name follows the command, and stored in the DSS file opened. The entire import file is read in and data is stored in the DSS file according to the previously defined exchange format and exchange variables. If the data to be imported is regular-interval time series data, and the data's dates and times are not given in the import file, then the starting time of the data must be specified using the time (TI) command. Because the end of the time window is implied by the beginning time and the number of data, an ending time is not needed. Irregular-interval time series data must have a date and time for each line in the import file.

For more information on importing data, refer to Chapter 6 Export-Import.

3.19 Open

Use:

OP, filename, parameters

The open command opens the DSS file whose name follows the command. If no file name follows the command, a list of DSS files within the current directory is displayed. By moving the cursor to a file name using the space bar (or on DOS the cursor keys), then pressing the return key, that file will be opened.

File size parameters may be set for new files to be opened. These are intended for use where the size and type of DSS file to be generated is known.

Parameters (New Files Only):

1. Table Type - A "stable" hash (address) table type file will be generated if the parameter "TABLE=STABLE" follows the file name. This type of addressing table is primarily intended for data bases that do not change in size frequently (e.g., a master data base file). A "dynamic" hash table is the default.
2. Size - The hash table size can be controlled to optimize storing and retrieval of data according to the expected number of records to be stored in the file. (This is automatically done during a "squeeze".) The size parameter is specified by "SIZE=", then the expected number of records in the file, or one of following sizes:

<u>Size Name</u>	<u>Target Number</u>	<u>Target Range</u>
TINY	20	1-50
EXTRA-SMALL	50	1-200
SMALL	200	100-1,000
MEDIUM (default)	1,000	200-5,000
LARGE	4,000	1,000-10,000
EXTRA-LARGE	10,000	2,000-20,000
HUGE	20,000	5,000-50,000
EXTRA-HUGE	50,000	>25,000

Examples:

OP DATAB SIZE=EXTRA-LARGE

(opens a new dss file with a parameter size of extra large)

OP DSSDATA SIZE=LARGE TABLE=STABLE

(opens a new dss file with a stable type table and parameters set to large)

3.21 Read Data

Use:

RE.options, filename

The Read Data command reads data from a formatted ASCII file (whose name follows the command) that was previously created by the Write Data command in DSSUTL, and stores that data in the DSS file currently opened. This command is intended to be used when transferring DSS data from one computer to another. (See exporting and importing for reading data from another program.)

Options:

1. P (Preserve) - Records with the same pathname in the file will not be overwritten.

The following options apply only to time series data (and are infrequently used).

Regular-Interval Time series Data:

2. C (Compression) - Ignore any data compression settings in the read data file, and use the file compression methods set in the DSS file.
3. M (Missing) - Replace missing data flags only. Only -901's (or non-existent records) will be replaced in the DSS file.
4. U (Update) - Will not allow missing data flags (-901's) in the originating file to replace valid values.
5. R (Regardless) - Normally, if the data to be stored for one record is all missing (-901), that record is not written. The R option bypasses this, and will allow a record of all -901's to be stored.

Irregular-Interval Time series Data:

6. M (Merge) - If data for the same record already exists in the DSS file, the M option will merge the two sets together. For example, if data in the file exists for Noon, 2 and 4 p.m., and data in the read file is for 1, 3, and 5 p.m., the M option will cause the final data set to be for Noon, 1, 2, 3, 4, and 5 p.m.. Normally all data is replaced.

3.22 Record Tags

Use:

`RT.option, parameters`

The record tags command controls the tag names given to records. It can set or display the file tag scheme, re-tag all records within the file using the current scheme (or catalog reference number), or set specific record(s) to a given tag.

A record tag is a one to eight character semi-permanent record identifier, that is not necessarily unique. It must begin with a non-numeric character. It can be set by the user, or can be set according to a scheme base on the parts of the pathname. The default record tag is the letter "T" followed by the sequence number (number of records in the file). Tag(s) may be used in place of pathnames in several DSS programs. If more than one pathname has the same tag, only the first one found will be used.

Tag Schemes:

A file tag scheme generates tag(s) based on letters from the pathname. A typical tag might be the location name followed by a portion of the data type. For example, the observed flow at location NATP might have a tag of NATP-FO; the barometric pressure at FLD might be FLD-BP. Each character in the tag scheme is set by specifying the pathname part letter (A, B, C, D, E, or F) followed by the character position number in that part. This is followed by a comma, then another tag character specifier. When a character or symbol without a character position is used, that character is inserted into the tag. The tag "NATP-FO" given above was generated by the following scheme:

```
B1 , B2 , B3 , B4 , - , C1 , F1
```

This generates a tag using the first through fourth characters of the B part, a dash, the first character of the C part, then the first character of the F part. If no character corresponds to the position given, that character is ignored.

It is also possible to use characters from the second word of a part by preceding the part letter with an underscore "_". The tag "FLD-BP", from a pathname with a C part of "BAROMETRIC PRESSURE", was created using the following scheme:

```
B1 , B2 , B3 , - , C1 , _C1
```

Note that the underscore causes the first character position of the second word to start counting at one. Pathname part words are delimited by any of the following characters "-@_+.;:". (If the above C part were "FLOW-NATURAL", the tag would be "FLD-FN".) If there were no second word for that part, that character would be ignored.

Option:

1. S (Set Scheme) - Sets the tag scheme to that designated in the parameters. If the parameter is "NONE", the current scheme is removed, and the default tag(s) are used ("T" followed by the sequence number).

Parameters:

1. ? - The file tag scheme is displayed.
2. Scheme (used with .S option) - Sets the file tag scheme. (This overwrites the scheme already set.)
3. NONE (the word "NONE", used with .S option) - Removes the current file tag scheme.
4. RETAG - Causes all records within the file to be re-tagged.
5. TAG=new-tag, pathname (or other reference) - A new tag for individual record(s) is set by specifying the parameter "TAG=" followed by the new tag, followed by the pathname or other record identifier(s) (such as reference number(s) or selective catalog pathname parts). Because tag(s) may be non-unique, several records may be set to have the same tag with this command.
6. SEQUENCE - A new beginning sequence number may be set by the parameter "SEQUENCE=" (or abbreviated to "SEQ=") followed by a integer number. When a new default tag is created (e.g., T14), this sequence number is used for the numeric portion of the tag.
7. REFERENCE - When the parameter is the word "REFERENCE" (which may be abbreviated to "REF"), all records are re-tagged according to their catalog reference number (a "T" followed by the number). This parameter requires that the catalog must already exist.

Examples:

```
RT ? (displays the current file tag scheme)
RT .S B1 , B2 , B3 , - , F1 , C1 , _C1 (sets the file tag scheme)
RT RETAG (Re-tags all records within the file with the
current scheme or new sequence numbers.)

RT TAG=SAC-FLOW /SACRAMENTO/I ST/FLOW/01JAN1980/1HOUR/OBS/
RT TAG=FOL-OUT B=FOLSOM, C=FLOW-RES OUT
RT TAG=SRC-TEMP 33-38, 44, 46
RT TAG=SRC-PRE T241
RT REFERENCE (Re-tag all records according to their catalog
reference number)

RT SEQUENCE=100 (Set the current sequence number to 100)
```

3.23 Rename

Use:

RN, parameters

The rename command renames record pathnames. One or more of the pathname parts may be changed, or the entire pathname may be changed. The rename command does not alter the record tag.

After giving the rename command and the associated parameters, DSSUTL will display a list of the pathnames to be changed, then prompt the user for new pathname part(s) or a new pathname. To change a part, enter the part letter followed by an equal sign and the new part. Alternatively, a entire new pathname may be entered if only one record is being renamed. If the user decides not to rename any pathnames, the word "QUIT" may be entered to return to the main program.

Parameters:

1. None - Uses the pathname set in memory.
2. Pathname - If a pathname follows the command, that record is renamed.
4. Tag(s) - Renames the record(s) corresponding to the tag(s) given.
5. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file to rename.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to rename. For more information, refer to the selective catalog section.
7. ALL - All pathnames in the catalog will be renamed.

Example:

To rename all records with a B part of "STH BEND" to be "SOUTH BEND" the following command is given:

```
RN B=STH BEND
```

The pathnames to be renamed are listed, then the prompt "Enter New Pathname Part(s)" is given. In response, the user types:

```
B=SOUTH BEND
```

More than just one part may be renamed at the same time, if desired. For example:

```
RN B=STH BEND, C=FLOW  
B=SOUTH BEND, F=COMPUTED
```

3.24 Squeeze

Use:

`SQ.options, parameters`

The squeeze command removes inactive space that may have accumulated from actions

Examples:

```
SQ.A  
SQ.T SIZE=2000  
SQ TABLE=STABLE  
SQ SIZE=EXTRA-LARGE
```

(no adjustments are made to the file)

3.25 Status

Use:

ST, parameters

If no parameters are given, the status command displays the current memory settings, including the name of the DSS file(s) opened, the pathname in memory and the time window set. If command(s) are entered as parameters, the settings for these commands are displayed. For example, if "ST TIME" is entered, the current time window set will be displayed.

Parameters:

1. None - The status of the current memory settings are displayed.
2. AR - The name of the DSS archive file is displayed.
3. CA - A message is printed indicating whether a catalog exists for the DSS file opened.
4. CO - The name of the DSS copy file is displayed.
5. DC - The file's default data compression methods are displayed.
6. DE - A list of the record pathnames just deleted (and available to undelete) is displayed. (The command UD with no parameters will undelete these records.)
7. EF - Displays the exchange format.
8. EV - Displays a list of the exchange variables, and what pathnames they are set to.
9. FO - The format currently set is displayed.
10. OP - The name of the DSS file opened is displayed.
11. PA - The pathname set in memory is displayed.
12. RT - The file's record tag scheme is displayed.
13. TI - The current time window is displayed.
14. UD - A list of all the record pathnames in the file that are available to undelete is displayed.
15. WR - The name of the file last used for writing data to is displayed.

3.26 Tabulate

Use:

TA.options, parameters

The tabulate command displays data (one screen at a time) from the specified records. Unless an output format has been previously set using the FORMAT command, DSSUTL will select an appropriate format based on the magnitude of the data.

For regular-interval time series data, the data is displayed in columns of 6 or less, along with the date and time corresponding to the data in the first column. A single column of data, which will have a date and time for each data value, may be produced by setting the FORMAT with one column (e.g., "FO (1F12.4)"). Irregular-interval time series data is always displayed in a single column form.

Options:

1. A (All) - Tabulates all data to the screen without paging. (This option is automatically invoked if the tabulation is written to a file.)
2. F (File) - The data is written to a file instead of to the screen. The name of the file to write to is asked for.
3. P (Printer) - The tabulated data is sent to the printer instead of the screen. The printer should be on-line before the command is given.

Options for Time series Data:

4. N (No Quality Flags) - If the record has data quality flags associated with it, this will cause the flags not to be displayed (otherwise they will be printed on a separate line in a hex format).
5. C (Complete) - All data within the record, including leading and trailing missing data (-901's), are tabulated. This option ignores any time window set.
6. D (Data only) - If a time window has been set, then this option causes missing data at the beginning and ending of the window not to be tabulated. If no time window is set, this option is automatically used.

Parameters:

1. None - Uses the pathname set in memory. For time series data, if a time window has been set, the data within that time window is tabulated.
2. Pathname - If a pathname follows the command, that record is tabulated.
3. Tag(s) - Tabulates the record(s) corresponding to the tag(s) given.

4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file for tabulating. Any time window set will be adhered to if the data is time series.
5. Date Reference (D=...) - Uses the current catalog and selects records to tabulate based upon a "D" part relative to the current date. The "D=" must be followed either by a D (for the current day), a M (for the current month), or a Y (for the current year), a minus (or plus) sign, a number, then another D, M, or Y. For example:
 D=M-2M (The current month minus two months)
 D=D-60D (The current day minus 60 days)
 The time interval (E part) is typically specified along with this parameter.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to tabulate. For more information, refer to the selective catalog section.
7. ALL - Tabulates data for all pathnames in the catalog file.

Examples:

TA 2	(Tabulate the record with pathname reference number 2)
TA.P C=FLOW	(Print a tabulation of all records which have a C part of "FLOW" in the catalog)
TA PROG=HEC1	(Tabulates all data that was written by program HEC1.)

3.27 Time

Use:

TI, starting date, time, ending date, time

The time command sets the time window used with time series data. If no time window is set, data for the entire record is used. To set a time window, follow the time command by the starting date and time then the ending date and time. The time or date may be in either order, as long as the starting date and time precede the ending date and time.

A time must be a four digit number, given in 24 hour clock time. A date can be one of several styles, but must not contain any spaces within it. (A 7 or 9 character military style date is typically used.)

A time window can be set relative to the system time by the single character "T", optionally followed by a minus (or plus) sign, a number, then a "H" for hours, or a "D" for days, or a "Y" for years. In addition, a fixed hour for the current day may be specified by using "T" as the date portion, then specifying the time in the next field. See the examples below.

A time offset may be specified from what the time window is currently set to by giving just a plus or minus sign followed by a number then a "H" for hours, or a "D" for days, or a "Y" for years. The ending date/time may be changed without affecting the beginning date/time by leaving empty fields (identified by commas) for the beginning of the time window.

To clear the time window, just enter the time command (with no times).

Examples:

TI 01MAR72, 2400, 14JAN73, 1200	
TI 2400, 01MAR72, 1200, 14JAN73	
TI T-4H, T	(current date/time - 4 hours, current date/time)
TI T, 0200, T, 1600	(today at 2 a.m., today at 4 p.m.)
TI T-5Y, T-40D	(today - 5 years, today - 40 days)
TI -2D +8H	(subtract 2 days from the starting date/time, add 8 hours to the ending date/time)
TI , , , +12H	(add 12 hours to the ending date/time)
TI , , , 20SEP72	(Change the ending date to 20 September 1972)
TI 1200	(Change the starting time to 1200)
TI	(Clear the time window)

3.28 Undelete

Use:

UD, parameters

The undelete command restores records that have been previously deleted, but not yet physically removed by the squeeze command. If some records were deleted in the previous command, they may be undeleted just by entering the undelete command with no parameters (DSSUTL retains a list of the pathnames from the last delete command).

Any or all records may be undeleted until the file is squeezed. Once the file has been squeezed, all deleted records are physically removed and cannot be recovered. If the catalog has not been updated since the desired records have been deleted, it may be used to specify the records to undelete.

Parameters:

1. None - If records have just been deleted, DSSUTL will use an internal list of pathnames from the last delete command to undelete. If not, the pathname set in memory will be undeleted.
2. ? - A list of the record pathnames that are available to undelete is displayed.
3. Pathname - If a pathname follows the command, that record is undeleted.
4. Tag(s) - Undeletes the record(s) corresponding to the tag(s) given.
5. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file to undelete (assuming that the catalog has not been updated since those records were deleted).
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to undelete (assuming that the catalog has not been updated since those records were deleted). For more information, refer to the selective catalog section.
7. ALL - Undeletes all records in the DSS file that have been deleted since the last squeeze. This parameter does not use the catalog.

Examples:

DE F=COMPUTED	
UD	(this will undelete those records that were deleted in the previous command)
UD D=01JAN1972	(this will use the catalog file to undelete records with a D part of 01JAN1972, assuming that it has not been updated since the delete)
UD ALL	(undeletes all records in the DSS file)

3.29 Write Data

Use:

```
WR.options, TO=outfile, parameters
```

The Write Data command writes data to an ASCII file in a formatted form. This file can then be transmitted to another computer, where it may be read into a DSS file using the Read Data command. The command can also be used to aid in creating a data input file for a program that does not have DSS capabilities.

The name of the file to write the data to (outfile) may be specified by "TO=" followed by the file name. It only needs to be specified once in a session. If no file name is given, the file name will be asked for.

Unless an output format has been previously set using the FORMAT command, DSSUTL will select an appropriate format based on the magnitude of the data. If the data has many significant digits, it is wise to specify a format, as the data will be truncated to the number of significant digits shown in the file when read back in.

Options for Time series Data:

1. T (Time) - This option will cause a date and time identifier to be written at the beginning of each line, similar to the tabulate command. The date and time correspond to the data value in the first column. These are for informative purposes only as the date and time in the header are the only values that are actually used.
2. C (Complete) - All data within the record, including leading and trailing missing data (-901's), are written. This option ignores any time window set.
3. D (Data only) - If a time window has been set, then this option causes missing data at the beginning and ending of the window not to be written. If no time window was set, this option is automatically used.
4. N (No Quality Flags) - If the record has data quality flags associated with it, this will cause the flags not to be printed (otherwise they will be printed on a separate line in a hex format).

Parameters:

1. None - Uses the pathname set in memory. For time series data, if a time window has been set, the data within that time window is written.
2. Pathname - If a pathname follows the command, that record is written.
3. Tag(s) - Uses the record(s) corresponding to the tag(s) given.
4. N1, N2, N3-N4 - Obtains pathnames with reference numbers N1, N2, and N3 through N4 from the current catalog file to use. Any time window set will be adhered to if the data is time series.

5. Date Reference (D=...) - Uses the current catalog and selects records to write based upon a "D" part relative to the current date. The "D=" must be followed either by a D (for the current day), a M (for the current month), or a Y (for the current year), a minus (or plus) sign, a number, then another D, M, or Y. For example:
 - D=M-2M (The current month minus two months)
 - D=D-60D (The current day minus 60 days)
 The time interval (E part) is typically specified along with this parameter.
6. Selective Catalog (A=..., B=...) - Uses the selective catalog capabilities for obtaining pathnames for records to write. For more information, refer to the selective catalog section.
7. ALL - Writes data for all pathnames in the catalog file.

Examples:

WR TO=DATFIL 3	(Writes data from the pathname with reference number 3 to the file "DATFIL")
WR.T C=FLOW	(Writes data for all records whose pathname in the catalog file has a C part of "FLOW". The data is appended to DATFIL. A date and time is placed at the beginning of each line.)
WR FLD=FLOW	(Writes the record with the tag "FLD-FLOW".)

Chapter 4

Secondary Commands

4.1 Close

Use:

CL, parameters

The close command closes files that have been opened. If no parameters are specified, all files are closed. Specific files may be closed by identifying the files to close.

Parameters:

1. None - All files that have been opened are closed.
2. MAIN - If one of the parameters is "MAIN", the main DSS file is closed.
3. COPY - If one of the parameters is "COPY", the DSS copy file (or secondary file) is closed.
4. TAB - If one of the parameters is "TAB", the tabulate file (used with the "F" option in tabulate) is closed.
5. WR - If one of the parameters is "WR", the write data file is closed.
6. DE - If one of the parameters is "DE", the file containing the list of pathnames of records last deleted is closed.

4.2 Debug

Use:

DB, parameters

The debug command sets the DSS message level.

Parameters:

1. Number - If a number between 1 and 15 is given, the message level is set to that number. Values between 5 and 9 are used to debug higher level subroutines, while values over 9 debug low level subroutines.
2. RESET - If the parameter "RESET" is given, the message level is set back to its default value.

4.3 Inquire

Use:

```
IN, parameter
```

The inquire command calls the DSS ZINQIR subroutine with the parameter specified, then prints out the character string returned, and the numeric value returned.

4.4 No Operation

Use:

```
NOP (or "*")
```

The no operation command is ignored by DSSUTL. This command is usually used for comment lines in an input file. An asterisk (*) may be used in place of NOP.

4.5 Set

Use:

```
SET, parameter1, parameter2
```

The set command calls the DSS ZSET subroutine to set various items. Parameter1 must be a character string, and parameter2 must be an integer number. Both parameters must always be set. If the item to be set only requires one parameter, use a dummy value for the other parameter.

4.6 Verify

Use:

```
VE ON/OFF
```

The verify command turns on or off the informative output displayed by DSSUTL. The default is on.

Informative output may be turned off for a single command by preceding that command with a period ".". For example:

```
U>.CO TO=DATFIL2 C=FLOW
```

is the same as

```
U>VE OFF
U>CO TO=DATFIL2 C=FLOW
U>VE ON
```

Chapter 5

Selective Catalog Capabilities

5.1 Introduction

The catalog routines provide a means of obtaining a sorted inventory of the record pathnames in a DSS file and pertinent information about those records (e.g., date last written). The catalog also provides an option of selecting and using pathnames based on their pathname parts. For example, a user can rename all observed flow data by the command:

```
RN C=FLOW, F=OBS
```

5.2 Sort Order

The user may define a sort sequence for the catalog based on the six pathname parts (A, B, C, D, E, or F). If the sort order is not defined, the default sequence of ABCFED is used. In this case all of the A parts are sorted alphabetically first. Then for each set of pathnames with identical A parts, the B parts are sorted, and similarly for the C, F, E and D parts.

The sort order may be defined as a parameter in the catalog command by entering the letter "O" followed by an equal sign and the sort order. Those parts not specified are filled in with the default order. For example:

```
CA O=FB  
CA O=FBACED
```

define the same sort sequence.

5.3 Selection of Pathnames Based on Last Written Date or Program

Pathnames may be selected from the catalog based on the date they were last written to, or the name of the program that wrote the record, when using a full (non-abbreviated) catalog. Records may be selected based on their last write date before, after, or on a given date. The two character identifier "LW" (for Last Write), or "PR" (for program) are used to identify the date or program name to select on. This identifier is followed by an equal (or < or > for last written date), then the appropriate date or program name. This identifier may be used wherever, or in addition to, selective pathname parts are given. A summary of the syntax is:

LW<date	Selects all pathnames last written before "date"
LW=date	Selects all pathnames last written on "date"
LW>date	Selects all pathnames last written after "date"
PR=name	Selects all pathnames last written by the program "name"

The date should be a 7 character military style date (e.g., 04JUL82), and the program name must be exactly as shown in the catalog file. The "LW=date" cannot be used in combination with the "LW<date" or "LW>date" identifiers. Some examples of this capability are:

```
AR TO=backup.dss LW<20JAN92
AR B=SOUTH BEND LW<25JUL92
DE PR=MYPROG
CO TO=computed.dss LW=16SEP92
CO F=PLAN 2 LW>16SEP92
```

5.4 Selection of Pathnames Based on Pathname Parts

The user may select which records to operate on (or catalog) based on pathname parts. The routines compare the requested parts with the actual parts of each pathname to decide which records to operate on. The user may select records based on the comparison of one or several of the pathname parts. This is done by entering the part identifier (A, B, C, D, E, or F) followed by an equal sign and the part specification. The eight types of part specifications that can be used are:

1. NAME - NAME must be identical to the part.
2. NAME@ - The part must begin with NAME.
3. @NAME - The part must end with NAME.
4. @NAME@ - The part must have the character segment NAME within it.
5. #NAME - The part must not be NAME.
6. #NAME@ - The part must not begin with NAME.
7. #@NAME - The part must not end with NAME.
8. #@NAME@ - The part must not have the character segment NAME within it.

Note that the at sign (@) and the pound sign (#) are special characters. The "@" indicates that any characters prior (or subsequent) to it are not compared. The "#" at the beginning of the part negates the specification. For example:

```
B=SOUTH MARINA, C=@FLOW@, D=@197@, E=1DAY, F=#PLAN 2
```

This line will obtain all pathnames in catalog with the following characteristics: the data is some type of daily flow for the 1970's at South Marina, except for data under plan 2.

Chapter 6

Export - Import

6.1 Introduction

A capability of DSSUTL is the exportation or importation of time series data for exchange between DSS and spreadsheet and data base programs, such as Lotus 123 and dBase. This is accomplished by defining data sets to be exchanged and an exchange format. When exporting data from DSS, the data is written to an ASCII (text) file with the defined format. The user then exits DSSUTL and executes the PC program and imports that ASCII file. Importing data to DSS essentially follows the reverse procedure. This capability can also serve as a means for a "user-defined" tabulation format.

Exporting or importing data with DSSUTL requires the use of several commands to perform one function. The "exchange format" command is inherently more complex than other commands. Because of this, users typically write these commands in an input file (refer to the introduction at the beginning of this document), or in the PREAD macro file "utlmac". The primary commands used for exporting and importing data are:

Exchange Format: This identifies the format (e.g., the third field contains the time of the data, the fourth field contain the flow, etc.) of the import or export file.

Exchange Variable: This provides a short abbreviation of the DSS pathname of the data to export or import for the exchange format (instead of the full pathname).

Export: This command tells DSSUTL to write data to a given file.

Import: This tells DSSUTL to read data from a given file.

6.2 Procedure

For each data set (DSS pathname record), an "exchange variable" must be provided to identify the DSS data in the exchange format. This essentially provides an abbreviated means of specifying a DSS pathname. The "EV" (exchange variable) command specifies a user defined name that is from one to eight characters long, and a pathname reference. For example, such an exchange variable might be:

```
STG1=/ALLEGHENY/NATP/STAGE/01JAN1991/1HOUR/OBS/
```

Up to 50 exchange variables may be specified for importing or exporting 50 data sets (or values per line).

An "exchange format" must be provided using the "EF" command. The exchange format tells DSSUTL how the data is to be written out or read in. Exchange variables specified in the format are enclosed in square brackets (for example, the above variable would be given as "[STG1]"). The date and time of the data may be designated by the reserved variables [DATE]

and [TIME]. When exporting data, the A, B, C, D, E, or F parts of the pathname can also be exported by the reserved variable [APART], [BPART], etc.. (Only one A and B part may be read from an import file.) An example of an exchange format for exporting data might be:

```
COE [DATE], [TIME]: [BPART] [STG1] [PRECIP]; [BPART] [STG2] [TEMP]
```

In the above format, the date, time and B part correspond to the data value following (e.g., the first [BPART] corresponds to [STG1], while the second corresponds to [STG2]).

Characters that are not defined variables within square brackets are copied just as they are to the export file. In this example, "COE", commas, spaces, the colon, etc. are copied to the export file.

In addition, the style of the date and time can be specified, as well as the format of the data. Tabs may be defined to position data or other items to specific columns in the export file. A single header line (which can contain items like the A, B, C parts) to be written at the top of the export file can be designated by the EF.H command option.

The format to identify data to be imported uses the same EF format. Imported data is identified by fields (data or characters separated by blanks and/or a comma), and exchange variable. Data exported using the above format may be imported by the following format:

```
[SKIP] [DATE] [TIME] [SKIP] [STG1] [PRECIP] [SKIP] [STG2] [TEMP]
```

The "[SKIP]" identifier tells DSSUTL to skip that field. Since data is read in by fields, the comma, colon and semi-colon are not given. Refer to the exchange format documentation for specifics on this command.

Once the exchange format and exchange variables have been entered, data may be exported with the EXPORT command or imported with the IMPORT command. A time window must be used when exporting data, and may be used when importing (if the import file does not contain the date and time of the data). The file name to export to, or import from, follows the command name.

6.3 Examples

The commands used in exporting and importing data are more complex than other DSSUTL commands. Because of this, it is wise to enter the commands in an input file to DSSUTL, or in the PREAD macro file "UTLMAC". The PREAD macro file can contain many sets of commands that are identified by a macro name. DSSUTL will automatically connect to the macro file in the same directory. To execute those commands, the user enters "!RUN macro-name" at the DSSUTL prompt (where macro-name is the name of the macro to execute). The macro capability is ideal to set up several of the export - import commands. For example, the user might run a macro to set all the exchange variables and the exchange format. Then the user can manually enter a time window and the export command. Thus, the same macro could be used on a daily basis without any modification.

Example 1. Export irregular interval time series data

DSSUTL commands:

```
EV SWS=/SACRAMENTO/SPO/WIND SPEED/07MAY1991/IR-DAY/OBS/
EV SWD=C=WIND DIRECTION
EV SBP=C=BAR PRESSURE
EV SRH=C=REL HUMIDITY
EV ST=C=TEMPERATURE
EF.H [T:3][DATE:Mmm D, YY] [T:16][CPART:ST] [T:30][CPART:SRH] //
  [T:45][CPART:SBP] [T:60][CPART:SWS] [T:75][CPART:SWD]
EF [T:8][TIME:H:MM am/pm] [T:20][ST:I3] [T:34][SRH:I3] //
  [T:45][SBP:10.2] [T:65][SWS:I3] [T:80][SWD:I3]
TIME 09MAY91 1130 09MAY91 1230
EXP EXP.1
```

Export file (Exp. 1):

May	9, 91	TEMPERATURE	REL HUMIDITY	BAR PRESSURE	WIND SPEED	WIND DIRECTION
	11:31 am	-	36	-	-	-
	11:33 am	-	-	-	8	2
	11:35 am	65	-	-	-	-
	11:38 am	-	-	-	7	278
	11:39 am	-	33	-	-	-
	11:43 am	66	-	-	-	-
	11:45 am	-	-	-	6	250
	11:46 am	-	33	-	-	-
	11:50 am	67	32	-	-	-

Example 2. Export regular interval time series data

DSSUTL commands:

```
EV ELEV=/AMERICAN/FOLSOM/ELEV/01JAN1991/1DAY//
EV IN=C=FLOW-RES IN
EV OUT=C=FLOW-RES OUT
EV STOR=C=STOR-RES EOP
EF [T:3][DATE:Mmm D, YY] [CPART]: [STOR:I7] [CPART]: [ELEV:9.3] //
  [CPART]: [IN:I4] [CPART]: [OUT:I4]
TIME 01APR91 2400 30APR91 2400
EXP EXP.2
```

Export file (Exp. 2):

Apr	1, 91	STOR-RES EOP:	427903	ELEV:	404.840	FLOW-RES IN:	3654	FLOW-RES OUT:	428
Apr	2, 91	STOR-RES EOP:	435059	ELEV:	405.810	FLOW-RES IN:	3954	FLOW-RES OUT:	318
Apr	3, 91	STOR-RES EOP:	440945	ELEV:	406.600	FLOW-RES IN:	3365	FLOW-RES OUT:	361
Apr	4, 91	STOR-RES EOP:	446802	ELEV:	407.380	FLOW-RES IN:	3379	FLOW-RES OUT:	382
Apr	5, 91	STOR-RES EOP:	453619	ELEV:	408.280	FLOW-RES IN:	3907	FLOW-RES OUT:	431
Apr	6, 91	STOR-RES EOP:	461425	ELEV:	409.300	FLOW-RES IN:	4652	FLOW-RES OUT:	682
Apr	7, 91	STOR-RES EOP:	471649	ELEV:	410.620	FLOW-RES IN:	5557	FLOW-RES OUT:	382

Example 3. Import regular interval time series data

DSSUTL commands:

```
EV FLOW=/ALLEGHENY//FLOW//1HOUR/OBS/ UNITS=CFS TYPE=INST-VAL
EV STAGE=C=STAGE UNITS=FEET TYPE=INST-VAL
EV PRECIP=C=PRECIP-INC UNITS=INCHES TYPE=PER-CUM
EF [SKIP] [BPART] [DATE] [TIME] [FLOW] [STAGE] [PRECIP]
IMP FILE.IMP
```

Import file (File.imp):

COE	"NATP"	08MAY91	1700	23177.00	12.67	0.00
COE	"NATP"	08MAY91	1800	21371.00	12.49	0.00
COE	"NATP"	08MAY91	1900	20678.00	12.42	0.00
COE	"NATP"	08MAY91	2000	20876.00	12.44	0.00
COE	"NATP"	08MAY91	2100	21074.00	12.46	0.00