

Scientific
Time Sharing
Corporation

ΔFMT
REPORT FORMATTING
in

APL PLUS

REPORT FORMATTING IN APL PLUS

INTRODUCTION

The ΔFMT facility gives the APL PLUS user a simple, economical, and adaptable means of producing reports involving tabular formatting of numeric and character data. Tabular output produced using ΔFMT takes about one-twentieth the CPU time required using other methods. Furthermore, ΔFMT needs almost no workspace storage. ΔFMT has many of the characteristics of an APL function; it may be grouped, saved, copied, or erased. A copy of ΔFMT may be obtained from workspace PLOTFORMAT of Public Library 1.

USING ΔFMT

ΔFMT employs its left argument to control printing of its right argument in one or more columns. The syntax of ΔFMT is

```
.FS  $\Delta FMT$  V1  
or  
FS  $\Delta FMT$  (V1;V2;V3;...;VN)
```

Right Argument

V1, V2, V3, ..., and VN are APL variables or expressions. They may be any combination of character or numeric scalars, vectors or matrices. A matrix is printed in one or more columns; a vector or scalar is printed in a single column.

Left Argument

FS is a character vector holding one or more format phrases separated by commas. Each phrase controls the printing of one or more columns. Allowable format phrases are:

mAw		character editing	
mIw	or	mqIw	integer editing
mFw.d	or	mqFw.d	fixed-point numeric editing
mFw.s			floating-point numeric editing
mXw			blank insertion
m□text□			text insertion

Here m, w, d, and s represent integer constants. q represents optional qualifiers and decorations, which are discussed in a later section.

(Note: the I editing phrase of the original ΔFMT is still supported, but has been superseded by the Z qualifier. See the section on Qualifiers and Decorations.)

m, which is optional, indicates how many times a phrase is to apply. If m is not present, it is taken to be 1. Thus, 3A1 is equivalent to A1,A1,A1; 2I4 is equivalent to I4,I4.

w is the field width, indicating the number of print positions occupied by the edited value (for A, I, F, and E) or by inserted blanks (for X).

For fixed-point editing, d is the number of digits to appear to the right of the decimal point. d must be less than w. For floating-point editing, s is the number of significant digits to be printed. s must be less than w-4.

The A phrase is used only to edit character values; the I, F, and E phrases are used only to edit numeric values.

Characters enclosed between \square symbols are inserted directly into the edited line. The field width of the inserted text is the same as the number of text characters between the \square symbols.

Here is a typical left argument for ΔFMT :

'X5,3A1,I4, \square US/CAN \square ,2F8.2'

The Result

A single edited result line holds one row from every matrix in the right argument of ΔFMT , and one element from every vector. The editing of successive fields on a line is controlled by successive phrases from *PS*, the left argument of ΔFMT . Each A, E, F, or I phrase from *PS*, or a repetition of such a phrase, processes one vector (or one column of a matrix) from the right argument. If ΔFMT runs out of format phrases before reaching the end of its right argument, it repeats from the left end of *PS* as necessary. For instance, the format string 'I4' can be used to print any number of four-column integer fields.

When the full field width is not needed, the edited value is right adjusted in the field and preceded by blanks. The E phrase is an exception. To maintain proper alignment of the decimal point, three print positions are always maintained on the right for the exponent. One or two of these may be blank, depending on the value being formatted.

The number of edited lines ΔFMT produces is equal to the length of the longest column in the right argument. Values with shorter columns have their columns extended by blanks.

Regardless of the `)DIGITS` setting, up to sixteen significant digits may be printed, depending on the value and the editing phrase. A format phrase requesting more than sixteen significant digits causes trailing digit positions to the left of the decimal point to be filled with underbars, and trailing digit positions to the right of the decimal point to be filled with blanks. A format phrase that does not give enough space for all leading significant digits (and sign, if negative) causes the entire field to be filled with asterisks.

Although this discussion has spoken of `ΔFMT`'s result as being printed, in fact `ΔFMT` can be used within a larger expression like any other APL function. Its result is always a character matrix. For instance,

```
T← 1 0 1 0 1 \, 'I3' ΔFMT 987
```

makes `T` a character vector whose value is `'9 8 7'`.

```
('F16.2' ΔFMT SHPMNTS) FAPPEND 20
```

puts the edited result into a file, perhaps for printing on the high-speed printer. If `ΔFMT` is not used within a larger expression, it prints its result as it goes rather than returning a matrix value. Thus, `ΔFMT` may operate without difficulty in situations where a `WS FULL` error might have been expected.

The length of an edited line is limited only by the workspace size; the `)WIDTH` setting has no effect on `ΔFMT`. Of course, the `)WIDTH` setting still determines how many characters are printed across the page before the paper is advanced.

ERROR REPORTS

Incorrect use of ΔFMT can result in these error reports:

RANK ERROR

The left argument of ΔFMT is not a vector; or some value in the right argument is not a scalar, vector, or matrix.

DOMAIN ERROR

The left argument has a numeric rather than a character value, or contains no phrases for editing the right argument, or contains an inconsistent format phrase -- for instance, a phrase specifying a field width of zero.

SYNTAX ERROR

A format phrase, or a value in the right argument, is not well formed.

WS FULL

ΔFMT 's result is too big for the active workspace to hold.

EXAMPLES

)COPY 1 PLOTFORMAT AFMT
SAVED 4.46.33 05/01/70

Formatting a Numeric Scalar

'I4' AFMT -34
-34
'F9.2' AFMT -631.487
-631.49 (2nd decimal place rounded)

X+1E20
'F24.1' AFMT X
10000000000000000_____ (More than 16 digits)

'F24.10' AFMT X
***** (Field width too small)

Formatting a Vector into a Column

'E10.4' AFMT X,+X
1.000E20
1.000E-20

N+2.7 .004 12 -6.31
'E8.2' AFMT N
2.7E0
4.0E-3
1.2E1
-6.3E0 (Compare positioning in this
and next example)

'F8.2' AFMT N
2.70
0.00 (0.004 rounded to 0)
12.00
-6.31

'F5.3' AFMT N
2.700
0.004

Formatting a Vector into a Column (contd.)

```

      'A1' ΔFMT 'ABCD'
A
B
C
D
      'A3' ΔFMT 'ABCD'
A
B
C
D

```

Formatting a Vector on a Single Line

```

      N+2.7 .004 12 -6.31
      , 'E8.2' ΔFMT N (Ravel the matrix result)
2.7E0 4.0E-3 1.2E1 -6.3E0

```

```

      'E8.2' ΔFMT (1,ρN)ρN (Make right argument
2.7E0 4.0E-3 1.2E1 -6.3E0 a 1×N matrix)

```

```

      'F8.2' ΔFMT (1,ρN)ρN
2.70 0.00 12.00 -6.31

```

```

      , 'A3' ΔFMT 'ABCD'
A B C D

```

```

      'F6.1,X2,E6.1,X2,I2' ΔFMT (1,ρN)ρN
2.7 4E-3 12 -6.3

```

In the example above, the F6.1 phrase is applied to both 2.7 and -6.3 .

```

      'A1,A2,A3,A4' ΔFMT 1 4ρ 'ABCD'
A B C D

```

Formatting a Matrix

MN+3 4p12
MA+3 3p 'CATDOGMAN'

'F6.0' ΔFMT MN
1. 2. 3. 4.
5. 6. 7. 8.
9. 10. 11. 12.

'2F6.1,I4,E10.2' ΔFMT MN
1.0 2.0 3 4.0E0
5.0 6.0 7 8.0E0
9.0 10.0 11 1.2E1

'X4,3A1' ΔFMT MA
CAT
DOG
MAN

'X4,3A2' ΔFMT MA
C A T
D O G
M A N

Formatting Multiple Values

A+'XY'
B+27.3
C+1 2 3 4
D+2 2p .1 .7 3.1 5

'A1,F7.1,I4,2F6.1' ΔFMT (A;B;C;D)
X 27.3 1 0.1 0.7
Y 2 3.1 5.0
3
4

Note that a right argument containing more than one value must be enclosed in parentheses.

N+4 6p 'NUTS BOLTS SCREWS TOTAL '
P+76 142 37
C+.05 .10 .06
Q+8 4 2
E+C×Q

'6A1,□| □,I3,F7.2,I5,F10.2' ΔFMT (N;P;C;Q;E,+/E)
NUTS | 76 0.05 8 0.40
BOLTS | 142 0.10 4 0.40
SCREWS | 37 0.06 2 0.12
TOTAL | 0.92

Note that the last value in the right argument is an expression, namely E,+/E .

QUALIFIERS AND DECORATIONS

Qualifier codes and decoration codes may be used to position and decorate the results of *I* and *F* editing, in the manner described below. Any number of qualifiers and decorations may be applied to an *F* or *I* phrase. Their order is immaterial -- `2M-CI10` has the same effect as `2CM-I10`.

Qualifier Codes

- B* Leaves the result field blank if the edited value is zero.
- C* Inserts commas between each group of three digits in the integer part of the edited value.
- L* Left-adjusts the value in the result field.
- Z* Fills unused leading positions in the result field with zeros (and inserted commas, if the *C* qualifier is used) instead of blanks.

Decoration Codes

- `M`text Places the text on the left of an edited negative value. The default value for the *M* decoration is the negative sign, '-'.
`N`text Places the text on the right of an edited negative value. The default value is null.
- `P`text Places the text on the left of an edited positive or zero value. The default value is null.
- `Q`text Places the text on the right of an edited positive or zero value. The default value is null.
- `R`text Fills the result field with the text, prior to editing the value into the field. The text will appear in all positions of the result field not filled by the edited value. The text is used as many times (replicated) as needed to fill the field. The default value is blank.

The field width for an *I* or *F* phrase must be large enough to hold the value plus any commas (from the *C* qualifier) and sign decorations (from the *P* and *Q* or *M* and *N* decorations, depending on the value's sign). An *N* or *Q* decoration causes a value of the corresponding sign to be moved left in the result field to make room for the decoration text. Thus the phrase `N DB I9` would offset a negative value three positions to the left of a positive value. If this is not wanted, `N DB Q I9` would align all values. (A similar statement holds for *M* and *P* decorations combined with an *L* qualifer.)

Examples of Qualifiers and Decorations

X+65423.45 -10 -.4 0 100

'BF10.1' ΔFMT X

-65423.5
-10.0
-0.4.

(0 is blanked)

100.0

'BI9' ΔFMT X

-65423
-10

(-0.4 is blanked)
(0 is blanked)

100

'CI9' ΔFMT X

-65,423
-10
0
0

100

'LI9' ΔFMT X

-65423
-10
0
0
100

'ZF10.1' ΔFMT X

-0065423.5
-0000010.0
-0000000.4
00000000.0
00000100.0

```

'M'-'F10.1' ΔFMT X
-65423.5
-10.0      (The M-text replaces '-' by '-')
-0.4
0.0
100.0

```

```

'P'+F10.1' ΔFMT X
-65423.5
-10.0
0.4
+0.0
+100.0

```

```

'Q' +I10' ΔFMT X
-65423
-10
0 +
0 +
100 +

```

```

'R'• I10' ΔFMT X
• • -65423
• • • -10
• • • • 0
• • • • 0
• • • • 100

```

Useful Applications

Here are examples of commercial formatting produced by combinations of qualifers and decorations.

Floating currency symbol:

```

'M'-'$P'F10.2' ΔFMT X
-$65423.45
-$10.00
-$0.40
$0.00
$100.00

```

Placing negative numbers into a separate column:

```

'N'          CI20' ΔFMT X
-65,423
-10
0
0
100

```

Check protection:

```
'R*P$CF10.2' AFMT 123.45 99999.95 .10
***$123.45
$99,999.95
*****$0.10
```

Replacing zeros by a literal indication:

```
'R NONEBF10.2' AFMT X
-65423.45
-10.00
-0.40
NONE
100.00
```

Placing negative results in parentheses:

```
'M(NN)QBF10.2' AFMT X
(65423.45)
(10.00)
(0.40)
0.00
100.00
```

Left adjust, with trailing zeros:

```
'LR0PBF10.1' AFMT X
-65423.500
-10.000000
-0.4000000
0.0000000
100.00000
```

(The P-text prevents left-adjustment of non-negative values into the leftmost position)

Credit and debit symbols:

```
'N DBCRMBBF14.2' AFMT X
65423.45 DB
10.00 DB
0.40 DB
100.00 CR
```

Scientific Time Sharing Corporation

2101 S Street, N.W., Washington, D.C. 20008 • (202) 462-4620

Offices in principal cities

REPORT FORMATTING ΔFMT Workspace 1 PLOTFORMAT

'fp,fp,....,fp' ΔFMT (expr;expr;...;expr)

expr scalar, vector, or matrix

fp format phrase:

mAw	character data
mEw.d	floating-point
mqFw.d	fixed-point
mqIw	integer
mXw	skip
m□text□	literal text

m repetition factor (optional)

w field width

d number of decimal positions

q qualifiers (optional):

B	blank if zero
C	comma insertion
L	left justify
Z	zero fill
M□text□	left of negative result
N□text□	right of negative result
P□text□	left of zero or positive result
Q□text□	right of zero or positive result
R□text□	background for result