(BSSR)

68010

BUREAU OF SOCIAL SCIENCE RESEARCH, INC.

WASHINGTON, D. C.

DOCFMT: A Format Style Macro File

by

Richard C. Roistacher

This Work Funded By

LEAA Grant 78-55-AX-0028

U. S. Department of Justice

December 14, 1979

TABLE OF CONTENTS

.DOCFMT: A Format Style Macro File

Richard C. Roistacher
Bureau of Social Science Research
Washington, DC 20036

December 14, 1979

What Is A Macro?

DOCFMT is a file of macro definitions and control statements for *FORMAT. A "macro" as used in computing, is a piece of text with a name. The language processing program, (e.g., *FORMAT), has facilities for replacing the name with the text to which the name is attached. Consider the author line on this document. It consists of three centered lines of typing which appears on many documents. In order to simplify the production of documents, the author line is included in a macro, which is defined in the control section of the document. The definition of the macro for producing the author line is:

```
/v/
define RCR
¦ /12m/ Richard C. Roistacher
¦ /1/ Bureau of Social Science Research
¦ /1/Washington, DC 20036 /ml2/
go
```

The macro is defined in the control section of the document. (The vertical bars in the definition indicate to format that the line contains macro text rather than control phrases.)

DOCFMT contains a predefined control segment which includes a number of macros (including this one). The macro is invoked by mentioning its name in the input text of the document. [1] A name is marked as a macro by being prefixed with a vertical bar, "¦". Thus, "¦RCR", written into the input file expands into the author line at the beginning of this document.

----

1. Although the names of macros can be written in either upper or lower case letters, all macro names quoted in this document are in upper case to make reading easier.

1

Macros With Inserted Text

Often, it is desirable to have a macro which contains some fixed and some variable text. The section headings in this document are examples of such macros, containing common formatting commands, but differing printed text. Consider a second level heading, which is centered, capitalized, and underlined, with a following paragraph indentation. The second level heading macro is defined:

```
/v/
define H2
|/12w6muf/ |par.1|
|  |contents(|par.1|) /fump/
go
```

The macro is invoked by writing

```
|H2('Introduction')
```

into the source text, which produces

Introduction

in the formatted document. The special word "|par.1|" stands for "parameter 1", which is the parenthesized text in the macro invocation. Parameter 1, which has the value "Introduction" is written out surrounded by the proper format commands to yield a properly formatted second level heading.

The second line of the macro prints nothing, but enters the text, "Introduction", and the current page number into the contents, which is printed after the entire document has been formatted. (Notice that this macro gives two copies of the same text, one with one in the document and one in the contents.)

Invoking DOCFMT

Suppose that you are producing a document whose input text is in a file called "Paper.m," and whose formatted text will be placed in a file called "Paper.x." The input text or "manuscript" file should begin with the following:

```
$empty paper.x ok
$run *format sprint=paper.x
include ccid:docfmt
```

In the above example, "ccid" represents the user ID on which DOCFMT is located. At the University of Michigan, the proper reference is to "SGDA:DOCFMT," while at Wayne State, the reference should be to "XCGG:DOCFMT".

DOCFMT ends with a "go" command, and should be followed immediately by the text of the document. If more control statements are to be inserted at the beginning of the document, .then the "include ccid:docfmt" line should be followed with a "/v/".

Document layout. When used at the head of a document, DOCFMT produces a finished document which fits on an 8-1/2" x 11" page with a 1-1/2" half left margin and 1 inch right top and bottom margin. DOCFMT places the page number of a chapter or major section at the bottom center of the page; the page numbers of succeeding pages at the upper right corner, with a title in the top margin of pages after the first. If desired, DOCFMT will produce titles and page numbers alternated in a form suitable for reproduction on facing pages.

DOCFMT sets tab stops every five positions, beginning with position 5. It also sets indentation points of (4,0), (9,0), etc., which match the tab stops. Note that many of the features of DOCFMT require the use of the pre-set tab stops and indents. If you wish to reset the tabs or indents, be sure to save the current values by using *FORMAT's STACK function. (See MTS Volume 15.)

The DOCFMT File

The following is the entire text of DOCFMT. The file loads functions from a load library called FMTLIB, and macros from a macro library called FMTMACLIB.

```
no control listing
no capitalization automatically
lowercase input
comment A footer is placed at the bottom of page 1
com in order to make space for a page number.
com A call to TITLE deletes this footer
lfooter
/j2e/
rfooter
/j2e/
page number 1 in line 60 offset 30 from the margin
tabs 5 10 15 20 25 30 35 40 45.50 55 60 65
indent (4,0)(4,4)(9,0)(9,4)(14,0)(14,4)(19,0)(19,4)
text starts on line 4, print position 11
sentences have 2 blanks
load page
load oddpage xcgg:fmtlib
load just
load contents
go
|library(xcgg:fmtmaclib)
|notnbox
|contents(PAD,3,.)
```

The DOCFMT Macros

The "Title" Macro. This macro puts a centered title at the beginning of a document or chapter and then places a running head and number at the top of each succeeding page. The macro takes two parameters; the first and second lines of the title. An example of this macro is

    |TITLE ('The Safety Pin','In the Roman Empire')

If the title is to have only one line, then the second parameter can be omitted. (The titles for this document were constructed with the |TITLE macro.)

Sometimes it is desirable to use a running head which is not exactly the same as the title at the beginning of the document. In this case, a third parameter can be used to suppress the printing of text at the point of the invocation of |TITLE . The |TITLE macro will set up the running heads and move the page number from the bottom to the top of the page, while the full title is placed at the front of the document as ordinary text. Consider the example:

    /j3m A Short History of The Safety Pin's
    /1/ Role in The Roman Empire /m12/
    |TITLE ('The Safety Pin','In the Roman Empire',x)

The value of the third parameter is of no consequence, its presence alone is what suppresses the printing.

American Psychological Association Headings

DOCFMT produces four levels of internal headings which follow the style of the American Psychological Association's Publication Manual. These macros are called |H1 , |H2 , |H3 , and |H4 .   Each of them takes a single parameter, the text of the heading.

|H1 IS A CENTERED UPPER CASE HEADING

|H2 Is Centered, Underlined, And Capitalized

|H3 Is Capitalized, Underlined, And Flush

|H4 is an indented paragraph heading. The paragraph heading is run directly into the text without a line break.

The text of each heading should have an initial capital letter but should otherwise be uncapitalized except in the case of proper nouns. The heading macros will automatically capitalize at levels 1, 2, and 3. The level 4 heading will add a "." to the end of the heading. Headings 1, 2, and 3, cause an automatic paragraph indentation in the following text, while heading 4 should have no paragraph break. Therefore, invocations of the heading macros should not be followed by "/p/" commands, as paragraphing will be done automatically. The macros are designed so that all that is necessary to change the level of a heading is to change the number in the macro name.

In addition to producing headings, these macros also produce entries in a table of contents, which can be printed at the end of the document.

Numbered Paragraphs

FORMAT has facilities for the manipulation of up to 100 counters which can be displayed in a variety of formats. DOCFMT has a set of macros which allow the production of numbered paragraphs.

The initialization macro. The beginning of a set of numbered paragraphs is marked with " |BEGINNP ." This macro, which usually has no parameters, is used to initialize the numbered paragraph counter and set an indentation. |BEGINNP takes two optional parameters. If no parameters are given, the numbered paragraph section will begin with "1." in arabic numerals. Otherwise, the number of the first paragraph in the section may be given as the first parameter. The second optional parameter is the format of the paragraph counter, the default for which is "integer." However, "alpha" and "roman" will cause paragraphs to be numbered with lower-case letters or roman numerals.

The numbering macro. The beginning of each numbered paragraph in a section is marked with " |NP ," which takes no parameters.

The numbered paragraph ending macro. The |ENDNP macro, which takes no parameters is used to end a numbered paragraph section. It restores the previous tabs, indentations, and margins.

An example.   The numbered paragraph macros give the
following results.

```
¦BEGINNP
¦NP Here is the first of several numbered paragraphs,
which are automatically indented by DOCFMT.
¦NP Here is a second numbered paragraph.
¦ENDNP
```

This input produces:

1. Here is the first of several numbered paragraphs, which are
    automatically indented by DOCFMT.

2. Here is a second numbered paragraph.

The same text headed with ¦BEGINNP (4,ALPHA) would print
as:

d. Here is the first of several numbered paragraphs, which are
    automatically indented by DOCFMT.

e. Here is a second numbered paragraph.

When formatting a section of numbered paragraphs, the
¦BEGINNP and ¦ENDNP macros should be on separate lines, while
the ¦NP macros should be on the same line as the beginning text
of each paragraph.   This arrangement has no effect on the
formatted document, but makes it easier to identify   the
beginning of each numbered paragraph and to change the order of
numbered paragraphs if desired.

Decimal Numbered Paragraphs

An   alternate   formatting   style   from   the   American
Psychological Association heading is the use of decimal numbered
paragraphs.

DOCFMT includes a set of macros for automatically producing
paragraph numbered 2.1.3., 2.5.4., etc.

1. ¦P1 produces a first-level numbered paragraph, i.e, numbered
    "1.", "2.", etc.

2. ¦P2 produces a second-level numbered paragraph, numbered
    "1.1.", "1.2.", etc.

3. ¦P3 produces a third-level paragraph, numbered "2.1.3.",
    etc.

These macros will automatically reset higher level counters when invoked and usually need no intervention by the user. The proper tabs are set and the counters initialized with a call to ·|BEGINP . An example of an input file with decimal numbered paragraphs is:

```
|BEGINP
|P1 ('A first level heading')
|P2 ('Second level heading')
|P2 Second level text.
|P3 Third level text.
|P1 ('Another first level heading')
|P2 More second level text.
|ENDNP
```

The formatted output from this text is:

1.   A first level heading.

   1.1   Second level heading.

   1.2   Second level text.

      1.2.1   Third level text.

2.   Another first level heading.

   2.1   More second level text.

These macros take two optional parameters. The first parameter is the text of the heading, which will be underlined but not capitalized. Use of this parameter will result in an entry being made into the table of contents. However, as the example shows, it is not necessary to give any parameters to the numbered paragraph macros in order to have a formatted outline.

The second, third, fourth and fifth parameters are values of the counters if it is desired to have the numbered paragraphs begin with a particular value. For example,

   |P3 ('Introduction',1,4,3)

would print as:

   1.4.3 Introduction.

Producing A Table Of Contents

Format has extensive facilities for producing a table of contents. The heading macros not only format section headings, but also make entries of the headings into a Table of Contents. The table of contents is built up during the formatting of the document by storing each section heading and the current page

number. At the end of the document, the stack of headings and
page numbers is popped and formatted into a printed table of
contents.

The ¦TOCINIT macro placed at the end of the document will
produce a table of contents on an unnumbered and untitled page.
To use the macro, just type ¦TOCINIT at the end of your input
file.

Other Useful Macros

Footnotes.   Footnotes can be formed by placing a ¦FN macro
at the beginning of the footnote text and an ¦EF macro at the
end of the text. The macros will provide a footnote counter in
the text and a similarly numbered footnote at the bottom of the
page.   ¦FN takes two optional parameters. The first optional
parameter can be used to reset the value of the footnote
counter.   If footnotes are to be renumbered at the beginning of
each chapter, the first footnote reference in the chapter should
be marked with " ¦FN (1)". The second optional parameter can be
any character.   Its presence suppresses the printing of a
footnote counter in the text.

Dates. The ¦DT macro causes the current date to be printed
centered and set off two lines from previous and following text.
The date at the beginning of this document was printed with the
¦DT macro. ¦DTNC prints a similar date, but without centering
or skipping lines.

Page skipping.   ¦ODDPAGE is a function which takes two
parameters. The first parameter is returned if the function is
called on an odd-numbered page.   The second parameter is
returned if the function is called on an even-numbered page.
The most common use of ¦ODDPAGE is to make sure that a new
chapter begins on an odd-numbered page (i.e., the front side of
the sheet.) This skip is accomplished by writing

    ¦ODDPAGE (' /s/ ' /s/',' /s/ ')

into the input text.   If the current page is odd, the first
string will be returned. The first "/s/" will skip to an even
page, the printing of the non trivial blank, "~" will make the
page non-empty, and the second "/s/" will cause a skip to the
top of the next, odd-numbered, page. If the function is called
on an even-numbered page, then the return of the second
parameter causes a simple skip to the next page. For the sake
of convenience, this call to obtain a skip to the next odd page
is defined in the ¦OPS macro, which needs no parameters. ¦EPS
will cause a conditional skip to the top of .the next even-
numbered page.

  <u>Listing and numbering.</u> DOCFMT turns off the listing of control phrases and input text. |CLN will turn on Control listing, Listing, and Numbering.

  <u>Turning off unnecessary titles and footers.</u> In order to put the number at the bottom of the first page, DOCFMT creates a footer. The footer, which has no further use after the first page of a document or chapter, causes the printing of extra blank lines in order to fill the page. The |KILLFOOTER macro, placed in the text somewhere after the beginning of the second page of output, will reduce the number of lines printed without altering the format of the document. |KILLFOOTER takes no parameters. The |KILLTITLE macro does the same thing for unwanted page titles.

  <u>Special box characters.</u> *FORMAT was designed to print on IBM's TN print train, which contains a number of special characters not found on the usual ASCII daisywheel terminal. The |NOTNBOX macro will translate these box characters into roughly equivalent ASCII characters, thus allowing the printing of boxes on the terminal. Since most DOCFMT users do not have access to a TN print train, DOCFMT includes an invocation of this macro.

  If you are going to print on a TN print train (or have a terminal which prints the box characters, include |TNBOX after the "include CCID:DOCFMT" line of your input file, and the TN characters will be printed.

INDEX

END