EUROPEAN ORGANISATION FOR NUCLEAR RESEARCH

REPORT User's Guide

For the Writing of Documents

D. Bates
M. Bertier
R. Cailliau

Abstract

Report runs on the PS/CCI Nord-10 program development system
It processes normal ASCII text files written in free-format,
and produces formatted A4 (or other) size output pages, ready
for reproduction and distribution. Special "Directive" words
are used to pass instructions to the Report program.

(text formatting by REPORT program)

Contents

1. Introduction

The program REPORT serves to format text that has been typed into a Sintran III file in free-format, into sections and paragraphs, producing a layout such as the one of this note. The input file (having filekind TXT-) for Report is created via the normal QED editor. When completed, this file is passed to the Report program for processing. The file is normally read as a continuous stream of words, with end-of-lines, tabs and multiple spaces being ignored. These words are then placed into fixed length lines for output, with padding blanks inserted between words, where necessary, to right justify all lines. In other words, output then looks like this paragraph (which was formatted by Report!).

Exceptions to the above procedure occur when the user gives a Directive to Report to do otherwise. Directives are strings starting with "%" signs and ending with another "%" or the end-of-line. They obviously do not appear on the processed document. They are used to instruct Report to take new paragraphs, chapters or sections or to modify the parameters of the document, such as the page width and length. There are in fact a total of 37 directives currently implemented, to allow flexibility on the part of the writer. The writer need not be familiar with them all, as most parameters have commonly used defaults for typical documents. All directives are described in detail in Chapter 6 and an example of their use is given in Appendix I.

If the user should ever require a % sign to be accepted, as a normal character rather than indicating a directive, he can dynamically specify an alternative character to be accepted as a directive indicator (see section 6.13).

The underline character "_" has special significance. It causes all characters from the position of this _ character up until the next _ character (including spaces), to be underlined. To print a real _ character, it is necessary to put the sequence "_ _" into the TXT file. (This will print an underlined blank).

Additional special features of Report include an automatically produced table of contents listing all chapter and section headings, with the page number on which they occur, and an index page, listing alphabetically all words or phrases specified in the index (X) directive. Conversions from lowercase to uppercase characters are automatic for the first character of all words in chapter and section headings longer than 3 characters and for first characters of sentences. Tables can be conveniently output by use of Absolute Mode.

The advantages of Report in generating documents are firstly that modifications never require the retyping of the whole document, but simply a brief editing of the TXT file and reprocessing by the program (or even the processed output file may be edited) and secondly that the document is always available on-line.

Novice users of Report are advised to read and utilise only the basic information in chapters 7-9, initially, since this will already enable them to produce adequately formatted documents.

2. Concept of a Page

Report thinks in terms of pages. Each page has the same length and width and left border which are set by default to 65 lines, 80 characters and 10 spaces respectively. These default values may be modified via appropriate directives before any text is given, but cannot be modified again within the document. However, the user is able to dynamically manipulate left and right margins and top and bottom borders within the confines of this page:

Initial values of parameters are:

```
PAGELENGTH    -   60 lines
PAGEWIDTH     -   80 characters
LEFTBORDER    -   10 blanks
TOPBORDER     -    4 lines
BOTTOMBORDER  -    0 lines
LEFTMARGIN    -    0 characters
RIGHTMARGIN   -    0 characters
```

In addition to manipulation by the user, the Leftmargin is also modified automatically by Report to cause indentation of sub-sections.

3. Chapters and Sections

A text can be subdivided into sections. Sections are numbered consecutively, but there may be different levels. To illustrate this, look at the table of contents at the beginning of this document: the first six sections are on the uppermost level. Section 5 (Different Modes of Operation) has two subsections, namely 5.1 and 5.2, and in addition 5.1 (Margin Mode) has four subsections on yet a lower level.

It would be a great nuisance if the writer had to number sections himself, since then he would have to change all numbers when adding a section as a later modification. Report, therefore, numbers the sections automatically; the writer must only change levels at the appropriate places. To indicate these level changes, there are four directives:

1) Section, which stays on the current level

2) Section-down, which moves one level down

3) Section-up, which moves one level up

4) Chapter, which moves right up to the top level, irrespective of the current level (is thus the same as Section, if already at the top level, except that in addition a Chapter always begins on a new page).

All of these directives take heading-strings as parameters. However in the case of Section-up, if no string is given, Report alters its section numbering, but prints nothing. This is for the case when the writer wishes to move up more than one level before printing the next section heading.

Chapter and Section headings are automatically underlined, entered into the Table of Contents, and followed by blank lines in the main text.

3.1. Section and Paragraph Footings

Obviously one does not normally wish that a section heading be printed at the bottom of a page, with no space remaining on that page for one or two lines of the body of that section. Rather, in that case, the section header should come at the top of the next page. Also a new paragraph looks untidy if only one line of it can be printed at the bottom of a page. Hence before taking new sections or paragraphs, a test is first made to see if secfooting or parfooting printable lines remain on the page. If not, then a new page is taken.

Secfooting is initialised to 6, i.e. 2 lines for Secspacing, 1 line for Secheading, 1 blank line after heading, leaving minimum of 2 lines for the start of the body of that section. Parfooting is initialised to 3, i.e. 1 line for parspacing and at least 2 lines for the new paragraph.

4. Cover Page

Since a major use of Report is to write PS/CCI Notes, a standard format has been defined which only requires that certain holes be filled in (see Appendix for template). These holes are the Title, Authors, Note Number and Date, all of which are optional, but if required must be specified, via the necesary directives, before any of the text of the note. If a Title is given, then a cover page is generated, otherwise not.

5. Different Modes of Operation

There are 2 distinct modes in which Report can function, namely Margin Mode and Absolute Mode. Report is initially running in Margin Mode and only switches to Absolute Modes when instructed to do so through the A directive.

5.1. Margin Mode

This is the normal mode of operation and can be returned to from Absolute Mode by use of the M directive.

In Margin Mode, blanks, tabs and end-of-lines serve only to delimit words, but are otherwise ignored. If a word begins with a % sign, it is assumed to be a directive word (unless the directive sign is changed) and Report takes the appropriate action (see below). Moreover, all other words are placed followed by one blank (2 in the case end-of-sentence words - i.e. those terminating in a . ? or ! and not recognised as abbreviations) into an output buffer. The length of this output buffer is dependent on the Pagewidth, Leftborder and current values of Left and Right Margin. When a word is read that will not fit into the remaining space in the output buffer, this word is held back. At this point, those words already in the buffer are moved to the right by the padding of extra blanks between the words, until the buffer is exactly filled. Then the buffer is output, and filling recommenced with the word that was held back.

Directives fall into 4 distinct categories:

5.1.1. Initialisation Directives

There are directives for changing some of the basic (static) parameters of the document generation and they can only be given at the start of the document before any ordinary text. Included in this category are directives for laying out of the cover page, such as Title and Author.

5.1.2. Parameter Modification

These are directives for changing the dynamic parameters of the printed page, such as margins. These directives are not acted upon immediately if the line, or perhaps the page, is partially filled. Rather, the parameter change is remembered and is actually performed either at ends-of-lines or ends-of-pages depending on whether it is referring to a line or a page parameter.

5.1.3. Directives Causing Line Termination

This class of directives includes Paragraph, Chapter, Section and Absolute directives, amongst others. They all require that the current partial line buffer is output as it is, i.e. without being right justified. Then a new line is taken and perhaps a heading, (preceeded by blank lines, in the case of Chapters and Sections).

5.1.4. Immediate Directives

Finally there are a few directives, such as Directivesign or Abbreviation, that can be acted upon immediately, since they do not cause any formatting modifications to the printed output.

5.2. Absolute Mode

This mode is entered by means of the A directive. Its purpose is for formatting of output under the control of the user - where he does not require right justification of lines, and where tab characters become meaningful. It causes the current partial line buffer to be output unadjusted and takes a new line. From then onwards, output becomes an exact copy of the input - that is to say that blanks and end-of-lines are significant, and that no right justification is done. In fact all line parameter settings are ignored, except that output begins from the left border. There is no check to see if lines are longer than the pagewidth.

Tabulator characters are recognised in Absolute Mode. They cause a number of blanks to be output, up to the next tab position. The positioning of the tab stops must be done by the user by means of the TS directive. Note that a tab character is entered into the source file, via QED, by typing (CTRL-V) (CTRL-I). If there are more tab characters on a line than tab stops defined, an error message will be given.

Underlining is also performed, as for Margin Mode, but is always terminated automatically by Report (if not by the writer) at the end of each line.

The only means of terminating this mode is by the M directive which must be at the beginning of a line by itself in the source file. No other directives are recognized in Absolute Mode.

6. Directives in Detail

Directives are always of the form:

%XY=String%

where XY is the directive mnemonic and may be 1 or 2 letters long. (They are always accepted in either upper or lower case.) The = sign is necessary if the directive takes a parameter, but does not give an error message for directives that take no parameter. A directive is terminated either with the directivesign (%) or an end-of-line. The final % sign is thus unnecessary if the remainder of the line is blank. Any number of directives can appear on the same line.

"String" is the directive parameter and may be a character string, or an integer of the form "n", "+n" or "-n". In the second case, this will imply either absolute setting, an increase or a decrease in one of the format control parameters. (See description of LM directive for an example). Note that if the "=n" is omitted, the value 0 is taken as default.

All the available directives are listed below in alphabetical order, with details of their use, (the ones that must be used before any text, and are normally used only once, are marked with an "*").

Chapter 7 lists the most frequently used directives, and shows that the large number of special purpose ones available should not lead to confusion.

6.1. Absolute Mode   %A%

Terminates Margin Mode and enters Absolute Mode. See section 5.2

6.2. Abbreviation   %AB=String%

Places "String" in the table of recognized abbreviations. This is to make known to Report any abbreviations used in the text, which terminate with a full stop (.) and which would otherwise be tagged as end-of-sentence words, with the usual consequences. A maximum of 10 abbreviations can be handled by Report, but note that any abbreviation no longer required may be removed from the table via the RA directive. Pre-defined abbreviations in Report are:

e.g.   i.e.   c.f.   viz.

6.3. Appendix   %AP=String%

Current partial linebuffer is output unadjusted.   A new page is taken and the word "Appendix", followed by "String", is printed at the top of the page.   Any desired sequence numbering of appendices should be provided by the writer as part of the String.   After each Appendix, the section numbering is re-initialised and thus appendices may have subsections if required.

6.4. Abstract*   ·%AS=String%

"String" is a line of text to be printed on the cover page.   Up to 10 such lines may be specified.   Note that they are not right-justified.

6.5. Author*   %AU=String%

"String is the author(s) name(s) to be printed on the cover page. Each use of the AU directive produces consecutively printed lines, centered within the printable width.   Maximum possible number of lines is 5.

6.6. Absolute Word   %AW=String%

"String" is a character string in which blanks are significant. That is to say that they are meaningful characters, not to be treated as word separators.   Hence "string" is to be output exactly as given, without any modification to the blanks contained within. Examples of usage might be the output of a telephone number such as: 83 61 11 or a string of words that should appear on the same line, such as "William the Conqueror" .

An absolute word will never be considered as an end-of-sentence, even if it ends in . ? or !

6.7. Bottom Border   %BB=n%

"n" modifies the number of lines that will be left blank at the bottom of each output page.   Initial value is 0.

6.8. Blank Lines   %BL=n%

Current partial linebuffer is output unadjusted, and then n blank lines are output.   (If n is greater than the number of printable lines remaining on the page, no blank lines will be carried over to the next page.) This directive has limited use, since PF and SF exist for varying the number of blank lines automatically given for paragraphs and sections.

6.9. Cookbook* %CB%

If the document is destined for a Cookbook note (PS programmer information notes) this directive should be given to modify the default pagewidth and pagelength for subsequent processing by CBFORMAT. The program CBFORMAT can only be used by the user COOKBOOK.

6.10. Centre %CE=String%

Current partial linebuffer is output unadjusted, and then "String" is output on the following line, centered between the Left and Right margins.

6.11. Chapter %CH=String%

Current partial linebuffer is output unadjusted and the "String" is output as the next chapter header at the top of the next page, automatically numbered and underlined. The left margin is decremented by 2*<the number of levels gone up>. If this should make the left margin negative, it is set to 0.

6.12. Date* %DA=String%

"String" is the date to be printed on the cover page. If not given, the cover page will have the date on which it was printed. (This will then give different dates for copies printed on different days!)

6.13. Directive Sign %DS=C%

C becomes the new character to be recognized as the delimiter of a directive. From then on, until the delimiter is further redefined, all directives should be enclosed by the character "C" rather than the "%" sign. This is obviously necessary if the writer wishes the % sign to be read as a normal character, rather than a directive indicator. Thus to print a % sign, using the "#" sign as the temporary directive sign, and, after, redefine the % sign as the directive delimiter, the following is necessary:

%DS=#%   %   #DS=%#

6.14. End French %EF%

Terminates special treatment of accent signs (see FR directive).

6.15. French   %FR%

This directive will cause special treatment of the characters ´ ^ and ` (in Margin Mode only), for the purpose of printing accents in french text. The algorithm is as follows: when ´ ^ or ` are read by Report and the preceeding character was a vowel that can gramatically receive that accent, then a backspace character is stored between the two. Thus to print the word "déjà", put the characters "de´ja`" into the TXT file. Note that this mode should only be used when necessary and switched off again, by the EF directive, when practical, to reduce the processing time by Report, but note also that each <u>output</u> line can only be all French or all non-French.

6.16. Inverted Paragraph   %IP=String%

This is the opposite to an ordinary paragraph, where the first line is indented. The current partial linebuffer is output unadjusted. Then the Left margin is temporarily moved Length-of-"String" spaces to the left and the String printed. Printing then continues as normal, and the left margin is reset to the old value at the end of the line. Thus the following layout can be created:

1) This is an inverted paragraph.   Notice that the block number is pulled to the side.

Note that since the left margin is automatically moved to the left for insertion of the block number, it is normally necessary for the writer to first move the left margin to the right (via the LM directive) to allow sufficient space. He then must reset the left margin at the end of the series of Inverted Paragraphs. Often, too, it is aesthetically more pleasing to move the right margin to the left by a roughly equal amount. Inverted paragraphs do not automatically start a new sentence, contrary to paragraphs.

6.17. Left Border*   %LB=n%

"n" modifies the width of the left border. Initial value is 10.

6.18. Left Margin   %LM=n%

"n" modifies the width of the left margin. Initial value is 0 characters. If n is unsigned then the left margin is set to n characters; if n is signed, then the left margin is modified by n.

Therefore:

%LM=5% means "set the left margin to be 5 positions wide",
%LM=+5% means "make the left margin 5 positions more than it is",
%LM=-5% means "make the left margin 5 positions less than it is".

6.19. Line Spacing %LS=n%

"n" modifies the line spacing between all printed lines. Normally, of course, this is set to single spacing, but may be set to other values, for example in a draft version, where space is to be left for comments.

6.20. Margin Mode %M%

This directive is for returning from Absolute Mode to Margin Mode. It must appear on a line by itself, starting from column one. If it is encountered while already in Margin Mode, an error message will be given.

6.21. Note Number* %NN=String%

"String" is the PS/CCI note number to be printed on the cover page.

6.22. Paragraph %P%

Current partial linebuffer is output, unadjusted. Parspacing (normally 1) blank lines are output, the following line is indented by Parindent (normally 3) spaces, and a new sentence is started.

6.23. Paragraph Footing %PF=n%

"n" modifies the minimum number of footing lines required for a new paragraph to begin on the same page. See chapter 4.

6.24. Page %PG%

Current partial linebuffer is output, unadjusted, and a new page is taken.

6.25. Paragraph Indent %PI=n%

"n" modifies the number of blanks by which paragraphs are indented. Initial value is 3.

6.26. Page Length* %PL=n%

"n" modifies the document Page Length. Initial value is 60 (4 of which are the Top Border, 2 for page numbering).

6.27. Page Width*   %PW=n%

"n" modifies the document Page Width. Initial value is 80 (10 of which are used for the Left Border).

6.28. Remove Abbreviation   %RA=String%

"String" is removed from the list of recognized abbreviations, and hence further occurrences of "String" in the text will be regarded as normal end-of-sentence words (see AB directive).

6.29. Right Margin   %RM=n%

"n" modifies the width of the right margin. Initial value is 0 characters.

6.30. Section Down   %SD=String%

Current partial linebuffer is output unadjusted, and then "String" is output as the next section header, after Secspacing blank lines, automatically numbered one level down and underlined. The left margin is incremented by 2 positions.

6.31. Section   %SE=String%

Current partial linebuffer is output unadjusted, and then "String" is output as the next section header, after Secspacing blank lines, automatically numbered, on current level, and underlined.

6.32. Section Up   %SU=String%

Current partial linebuffer is output unadjusted, and then "String" is output as the next section header, after Secspacing blank lines, automatically numbered on one level up and underlined. The left margin is decremented by 2 positions. However, note that in the special case where no String is given, no output is produced. This enables the writer to move up more than one level before printing the next section heading.

6.33. Section Footing   %SF=n%

"n" modifies the minimum number of footing lines required for a new section to begin on the same page. See chapter 4.

6.34. <u>Top Border</u>   %TB=n%

"n" modifies the number of lines that will be left blank at the top of each output page (after the two lines containing the page numbering). Initial value is 4.

6.35. <u>Title*</u>   %TI=String%

"String" is the title to be printed on the cover page. Each use of the TI directive produces consecutively printed lines, centered within the printable width. Maximum possible number of lines is 10. At least one title line must be given to force the printing of a cover page.

6.36. <u>Tabulator Set</u>   %TS=n,p,...%

Note the exceptional format here. This directive sets the tab position for use of tabs in Absolute Mode.   "p" sets the column marker of the "n"th tab position in a line. Any number of n,p pairs may be given in the one directive. Maximum possible value of n is 10. Initially no tabs are defined, and hence before any use of them, as many tabs as required per line, must be initialized.

Obvious Pn must be $> P(n-1) > ... > P1 > 0$ where n is the number of tabs defined.   Naturally errors may still occur if any text between two tab characters is greater than the distance between the two corresponding tab positions.

6.37. <u>Index</u>   %X=String%

"String" is placed into an Index which is written out at the end of the document (after any Appendices) as an alphabetically sorted list.   Note that "String" does <u>not</u> also appear as a string in the text.

- Reminder: all those directives indicated with an asterisk (*) must appear at the start of the document before any text of the document is given.

7. Summary of Most Useful Directives

The length of the previous chapter may give the impression that use of Report is rather complicated. However, most of the directives described are for creating special layouts, modifying default parameters and also for specifying the standard PS/CCI Cover page. The beginner need not bother with these extras and can manage well with the 5 basic directives briefly repeated below:

1) Chapter (%CH=String%)

   Used when a new chapter is required. "String" is the new chapter name, which will be automatically numbered.

2) Section Down (%SD=String%)

   Used when a new section is required, one level down. "String" is the new section name which will be automatically numbered and indented.

3) Section (%SE=String%)

   Used when a new section is required on same level as previous section. "String" is the new section name which will be automatically numbered and indented.

4) Section Up (%SU=String%)

   Used when a new section required, one or more levels up than previous section. The special case %SU% must be used to move up more than one level before printing a new section heading.

5) Paragraph (%P%)

   Used to specify that following text should begin a new paragraph.

Recommendations:

1) Put directives, or at least the five described above, at the start of a new line. Note that if you do not write anything else on that line, the last % sign need not be given.

2) Keep lines short in the TXT file and begin all sentences on new lines, for ease of later modification.

3) Write all text and directives in lowercase wherever possible for ease of later editing (searching) with QED.

9. Error Conditions

No pure text sequences should be able to lead Report into an error state - even if a word appears which is longer than the linelength and would hence not fit on any line, it is simply split in two at the end of the line. However, execution of directives obviously can lead to error situations, most of which, hopefully, will be detected by Report which can then take suitable action.

Perhaps one of the most common errors is when the writer has some % signs in his text and forgets to modify the directive sign character before this. Naturally Report will then expect a directive to follow, and when a recognizable one does not follow, it will print the error message "UNKNOWN DIRECTIVE".

Errors are divided into two categories, fatal and non-fatal. Both categories produce an error message as follows:

*** ON LINE NN *** <error type>

If the error was fatal, Report now tidies up its files and terminates, otherwise it continues after ignoring the faulty directive, since it is likely that the output will still be of use as a draft.

Note that if, at any time, Report is aborted by hitting the terminal escape key, then two temporary files will be left in the user's directory, namely <name>:PREP and <name>:CONT. They will automatically be deleted the next time that Report is run on the same TXT file.

Below is a copy of the source TXT file for Chapters 7 and 8 of this
document. It uses a fair selection of directives (perhaps untypically
many), whose action can be clearly seen.

%ds=@%
@ch=Summary of Most Useful Directives
The length of the previous chapter may give the impression that use of
Report is rather complicated.
However, most of the directives described are for creating special
layouts, modifying·default parameters and also for specifying the
standard PS/CCI Cover page. The beginner need not bother with these extras
and can manage well with the 5 basic directives briefly repeated below:
@lm=+7@ @rm=+7
@ip=1) @Chapter (%CH=String%)
@p@ Used when a new chapter is required. "String" is the new chapter
name, which will be automatically numbered.
@ip=2) @Section Down (%SD=String%)
@p@ Used when a new section is required, one level down. "String" is
the new section name which will be automatically numbered and indented.
@ip=3) @Section (%SE=String%)
@p@ Used when a new section is required on same level as previous section.
"String" is the new section name which will be automatically numbered
and indented.
@ip=4) @ Section Up (%SU=String%)
@p@ Used when a new section required, one or more levels up than previous
section. The special case %SU% must be used to move up more than one
level before printing a new section heading.
@ip=5) @ Paragraph (%P%)
@p@ Used to specify that following text should begin a new paragraph.
@bl=3@ @a@
            Recommendations:
@m@
@lm=+7@@rm=+7
@ip=1) @ Put directives, or at least the five described above, at the start
of a new line. note that if you do not write anything else on that line,
the last % sign need not be given.
@ip=2) @Keep lines short in the TXT file and begin all sentences on new
lines, for ease of later modification.
@ip=3) @ Write all text and directives in lowercase wherever possible for
ease of later editing (searching) with QED.
@lm=-7@@rm=-7
@ch=Running the REPORT program
When the input document, including its directives, has been typed into
a normal SINTRAN III file, via the QED editor,
the Report program itself must be called up as follows:
@ds=%@ %a%

     @_REPORT

     TXT FILE TO BE PROCESSED: _TXT-<name>
%m%
%p% user input is underlined. His filename should have the prefix TXT.
Assuming processing proceeded satisfactorily he will subsequently

Dir	Fullname	Parameter Use
=====	==========	===============
A	Absolute	-
AB	Abbreviation	Abbreviation noted
AP	Appendix	-
AS	Abstract*	-
AU	Author*	Authors' name
AW	Absolute Word	Absolute word given
BB	Bottom Border	Number of blank lines at bottom of page
BL	Blank Lines	Number of blank lines of spacing
CB	Cookbook*	Document will be a PS Cookbook Note
CE	Centre	String to be centered
CH	Chapter	Chapter title
DA	Date*	Document's date
DS	Directive Sign	New directive sign character
EF	End French	Terminate interpretation of accent signs
FR	French	Begin interpretation of accent signs
IP	Inverted Paragraph	Inverted paragraph string
LB	Left Border*	Number of columns at left of page
LM	Left Margin	Number of column adjustments
LS	Line Spacing	Physical lines between each printed line
M	Margin Mode	-
NN	Note Number*	Document's PS/CCI Note number
P	Paragraph	-
PF	Paragraph Footing	Minimum number of lines for new paragraph on page
PG	Page	-
PI	Paragraph Indent	Number of blanks to indent each paragraph
PL	Page Length*	Number of lines on a page
PW	Page Width*	Number of column on a page
RA	Remove Abbreviation	Abbreviation to be removed
RM	Right Margin	Number of column adjustments
SD	Section Down	Section title
SE	Section	Section title
SF	Section Footing	Minimum number of lines for new section on page
SU	Section Up	-
TB	Top Border	Number of blank lines at top of page
TI	Title*	Document title
TS	Tabulator Set	Tab number and position
X	Index	String to be indexed

* Must be given before any text