To: MTB Distribution

From: E. J. Wallman

Date: 1978 August 05

Subject: Multicolumn output in compose.

   This MTB proposes a feature for compose to produce multi-column, running text.

   The implementation of compose was recently changed to operate in what has been called "full page make up" mode. In this mode, output is accumulated until the image of an entire output page is complete before any output actually takes place. Operation in this mode is internal to compose and totally invisible to the user, hence, an MTB and review were not thought necessary.

   There are, however, a number of advanced text formatting features that require operation in this mode before they can be implemented. These new features affect the user interface to compose and will be proposed individually with MTBs.

   There is a great body of formatted documentation that is traditionally or advantageously done in multi-column format. Examples are newspapers (12 to 18 columns per double page), telephone books (4 columns per page), and magazines and technical journals (2 to 4 columns per page). With Multics entering the field of Text Processing, it is thought appropriate to provide a multi-column formatting feature in compose.

   Support of this multi-column feature requires the extension of two existing compose controls and the addition of two new ones.

   - Extension of the block-begin-footnote control to allow specification of column format or page format for the footnote.
   - Extension of the page-define control to allow the definition of up to 20 columns on a page.
   - Addition of a page-define-columns control to allow definition of up to 20 columns on a page without disturbing the remaining page definition parameters.
   - Addition of a break-column control.

   These changes are described by the following changes to the user documentation.

_____

The following changes are to be made in the WORDPRO Manual
(AZ98, Rev 0). The change bars appearing here mark the actual
changes and may or may not appear in the final Manual.

Change "Summary Information", page 9-16, to read:

```
.bb                                     block-begin
   b                                    inline
   a  {#}                               artwork
   e  {#}                               equations
   f  {s}{,c|p}                         footnote
   i                                    inline
   k  {#}                               keep
   l  {#}                               literal
   n  {<name>}{,a|r}                    named
   p  {#}                               picture
```

Change "Summary Information", page 9-17 (at the top), to read:

```
.br                                     break
   b                                    block
   b                                    block
   c  {#}                               column
   f                                    format
   n  {#}                               need
   p  {e|o|#|<+n>}                       page
   s  {#} {<string>}                     skip
```

Change "Summary Information", page 9-17 (at the bottom), to read:

```
.pd                                     page-define
   b  {<l>}{,<w>}{,<c>}{,<g>,<c> ...}{,b|u} all
   c  {<c>}{,<g>,<c> ...},{b|u}         column
   l  {<+n>}                            length
   w  {<+n>}                            width
```

Change "Breaks", pages 4-2,3, to read:

   Breaks

        A break is an event that causes an interruption of some
   processing mode. Four different breaks are defined.

      Format Break
           This break is caused by a text control that changes or
           interrupts the current formatting mode, but does not
           end a text block.   Examples are indent, undent,
           page-define-width, and break-format. Any pending input
           text is composed into the output as a short line. The
           current text block is continued with the new formatting
           mode.

Block Break
    This break is caused by a text control that defines a
    text block. Examples are space-block, break-block, and
    break-page. The current text block is terminated (as
    appropriate) with a format break and a new text block
    of the type specified is begun.

Column Break
    This break is caused by a text control that forces a
    new column. Examples are break-column and break-need.
    The current text block is terminated with a block break
    and a new text block is begun in the next available
    column. If the current page has only one column or the
    current column is the last column on the page, a page
    break occurs. A column break ensures that no text
    following the control that caused the break is printed
    in the current column.

Page Break
    This break is caused by a text control that forces a
    new page. Examples are break-page and break-need. A
    page break ensures that no text following the control
    causing the break is printed on the current page. If
    inline text is being processed, the current page is
    closed out (with footnotes and footers as appropriate).
    Any pending text is handled according to the control
    given (see "Controls" later in this description).

Change "Controls", page 4-12, to read:

    footnote: .bbf; {s}{,c|p}, no break, no substitution

                Suspend processing of the current text block and
                begin processing a footnote. The footnote
                formatting parameters are carried forward from the
                previous footnote or from the default values if no
                previous footnote has occurred. Any formatting
                parameters set while processing footnotes carry
                forward to all subsequent footnotes. If the first
                parameter "s" (for "suppress") is given, the
                footnote reference (e.g. "(2)") is omitted and the
                footnote counter is not incremented when leaving
                footnote mode. If the second parameter is given
                as "p", then the footnote is formatted according
                to the formatting parameters of the page. If the
                second parameter is given as "c", then the
                footnote is formatted according to the formatting
                parameters of the current column. If multi-column
                mode is not in effect, a given value of "c" is
                ignored. The default for the second parameter is
                "p".

Insert into "Controls", page 4-14 (after break-block):

column: .brc; {#}, column break, no substitution

> Terminate the current text block then fill the
> remainder of the current column with white space.
> If {#} is not given, then advance to the next
> column on the current page. If {#} is given, then
> advance to that column on the current page,
> filling intervening columns with white space. If
> [#] is given as 0, then terminate all columns
> (with balancing if specified by the
> page-define-all or page-define-columns controls)
> and revert to full page formatting until the
> occurrence of the next break-column control. The
> "next column" for this temporary reversion is
> defined to be column 1 of the current page. If
> the next column (however determined) is not
> defined for the current page, then cause a page
> break and begin with column 1 on the next page.
> If no columns are defined for the page, then map
> this control into a break-page. In some
> processing modes (e.g., keep mode and picture
> mode) that do not allow termination of a text
> block, this control is mapped into a break-format
> control.

Change "Controls", page 4-22,23, to read:

page-define: .pd{m}

> Set page definition parameters according to the
> modifier given.

all: .pd; {<l>}{,<w>}{,<c>}{,<g>,<c> ...}{,b¦u}, no break,
> no substitution

> Define the page according to the ordered set of
> values   <l>,<w>,<c>,<g>,<c>...,b¦u.   See the
> individual modifiers with the same letter codes
> following for additional information. If a value
> is not given for a parameter, (i.e., its field is
> blank or null), then its default value is used.

column: .pdc; {<c>}{,<g>,<c> ...}{,b¦u}, no break,
> no substitution

> Define the text columns on the page according to
> the given columns widths, <c>, and gutter widths,
> <g>. If no parameters are given, the page will
> contain a single column. If any value for <c> is
> omitted, the indicated column will have zero width
> and will not appear on the page. If any value for
> <g> is omitted, a default value of 3 is used. If
> any values are intentionally omitted, their

separating commas must still be given. The
maximum number of columns allowed on a page is 20.
If the sum of all widths given exceeds the amount
of space available for text on the page, an error
diagnostic message will be generated. The final
optional parameter controls the balancing of
unfilled columns at a page break. If "b" is
given, all columns on the page will be balanced
(that is, will contain equal amounts of text)
within the constraints established by artwork,
keep blocks, and widowing. If "u" is given, no
balancing will be done. The default for the last
parameter is "b".

Finally, for the interested reader, the first page of this MTB
is given again, this time in bi-columnar (or, more commonly,
"2up") format. The only source difference between the next page
and the actual first page is the addition of a ".pdc 31,,31,b"
control just ahead of the first paragraph.

To: MTB Distribution

From: E. J. Wallman

Date: 1978 August 05

Subject: Multicolumn output in compose.

This MTB proposes a feature for compose to produce multi-column, running text.

The implementation of compose was recently changed to operate in what has been called "full page make up" mode. In this mode, output is accumulated until the image of an entire output page is complete before any output actually takes place. Operation in this mode is internal to compose and totally invisible to the user, hence, an MTB and review were not thought necessary.

There are, however, several advanced text formatting features that require operation in this mode before they can be implemented. These new features affect the user interface to compose and will be proposed individually with MTBs.

There is a great body of formatted documentation that is traditionally or advantageously done in multi-column format. Examples are newspapers (12 to 18 columns per double page), telephone books (4 columns per page), and magazines and technical journals (2 to 4 columns per page). With Multics entering the field of Text Processing, it is thought appropriate to provide a multi-column formatting feature in compose.

Support of this multi-column feature requires the extension of two existing compose controls and the addition of two new ones.

- Extension of the block-begin-footnote control to allow specification of column format or page format for the footnote.
- Extension of the page-define control to allow the definition of up to 20 columns on a page.
- Addition of a page-define-columns control to allow definitions of up to 20 columns on a page without disturbing the remaining page definition parameters.
- Addition of a break-column control.

These changes are described by the following changes to the user documentation.