

Mail Merge and Statement Composer

Version 7

Users Manual



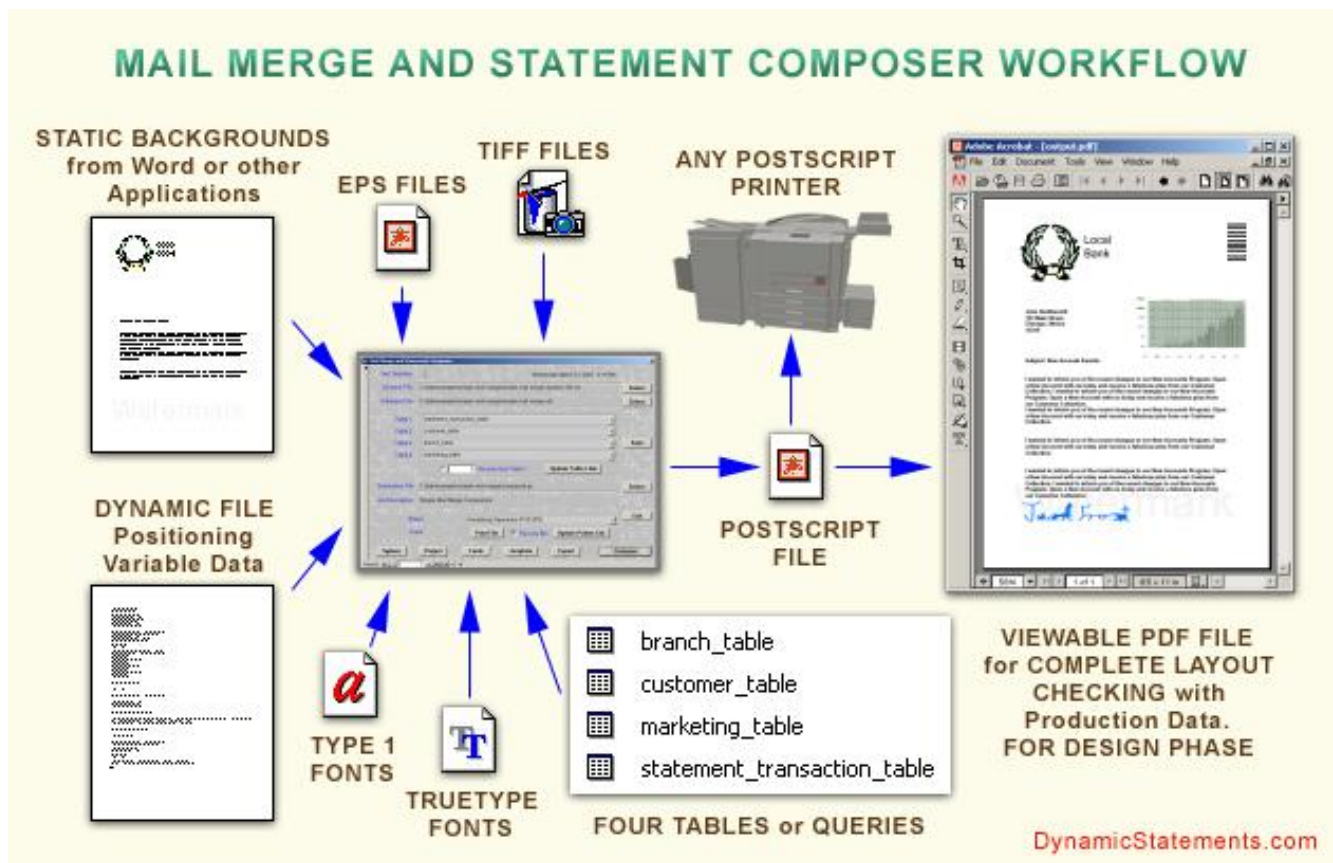
Dynamicstatements.com

BASIC OPERATION

Mail Merge and Statement Composer is a Page Composition Software Package, which provides a simple way to format data onto a page for printing to any Postscript printer device or for PDF conversion. The output Postscript files are completely self-contained with all the required fonts and graphics for an error-free RIP. Input Data is supplied from several sources.

- 1) A single database file, which is linked by a UDL (Universal Data Link) File. The database can be a variety of formats such as: TXT, DBF, MDB file, up to a full Database Server. UDL files provide an independent ability to test your database source connection.
- 2) Static EPS (Encapsulated Postscript) Backgrounds can be inserted. They are created easily from virtually any Software Application through a standard Postscript Printer Driver or Exported from various Applications.
- 3) Tiff images can be inserted which can be of any size, color depth and compression.
- 4) Any TrueType (.ttf) and Type 1 fonts (.pfb) can be used and user modified.
- 5) A single Dynamic File (simple text file) is used to format the variable data. Dynamic file Commands are explained in this document.

After all the input source files have been defined, clicking the “Compose” button will produce a single Postscript file at the destination location specified. You can then convert the Postscript file into a PDF or you can use the Print File button to send the Postscript file directly to your Postscript printer device. [Download the Free Screen Cam instructional videos](#) to view Application in action. The [easy-to-learn](#) videos will walk you through the entire Page Composition process. This Application has Full automation capability, including Composition and Schedule Tasks to help automate your workflow.



System Requirements:

Windows XP / 2003 / 2008 / Vista / 7 or Up

Access 2003 or Up, 96 MB Ram, 25 MB free disk space, Internet Access

Mail Merge and Statement Composer Commands

Categorical Order

PRINT JOB.....5

|banner_page|
|paper_color|
|paper_size|
|paper_1|
|paper_2|
|paper_3|
|paper_4|
|paper_5|
|paper_6|
|sided|

PAGE.....7

|bottom_margin|
|left_margin|
|new_page|
|right_margin|
|top_margin|

TEXT.....9

|case|
|end_text|
|font|
|line_space|
|new_line|
|no_space|
|point_size|
|spaces|
|text_alignment|
|underline|

POSITION.....12

|x|
|y|
|+x|
|+y|
|yt|
|new_origin|

DATA MANIPULATION.....14

|#1| to |#999|
|1| to |999|
|append|
|array_in|
|array_out|
|if|
|pad|
|store|

GRAPHICAL.....17

|box_border|
|box_fill|
|graphic_fetch| - [Optional](#)
|graphic_form|
|graphic_height|
|graphic_raw|
|graphic_width|
|graphic|
|line|
|rotate|
|reset_graphic_state|
|scale|

COLOR.....21

|color|

SYSTEM.....22

|| (comment)
|bookmark|
|date|
|goto|
|log|
|new_start|
|postscript_new_line|
|postscript_pass_through|
|restore_format|
|store_format|

|system_move_next_record|
|system_new_pass|
|system_set_limit|
|system_stop|
|time|

EVENT.....26

|end_line_event|
|end_old_page_event|
|start_new_page_event|

MATH.....27

|absolute|
|add|
|dec_ascii|
|dec_hex|
|dec_oct|
|decimal_places|
|divide|
|multiply|
|remainder|
|sign_change|
|subtract|

DATABASE.....30

|field|
|field_exists|
|move_first_lookup|
|move_next_lookup|
|multiple_record_field|
|next_record|
|store_lookup|
|timeout|

Alphabetical Order

1. #1 to #999	14	46. new_line 	10
2. 	22	47. new_origin	13
3. +x	12	48. new_page	7
4. +y	12	49. new_start	23
5. 1 to 999 	14	50. next_record	31
6. absolute	27	51. no_space	10
7. add	27	52. pad	15
8. append	14	53. paper_1	6
9. array_in	14	54. paper_2	6
10. array_out 	14	55. paper_3	6
11. banner_page	5	56. paper_4	6
12. bookmark	22	57. paper_5	6
13. bottom_margin	7	58. paper_6	6
14. box_border	17	59. paper_color	5
15. box_fill 	17	60. paper_size	5
16. case	9	61. point_size	10
17. color	21	62. postscript_new_line 	23
18. date 	22	63. postscript_pass_through	23
19. dec_ascii	27	64. remainder 	29
20. dec_hex 	27	65. reset_graphic_state	19
21. dec_oct 	28	66. restore_format 	24
22. decimal_places	28	67. right_margin	7
23. divide 	28	68. rotate 	19
24. end_line_event	26	69. scale 	20
25. end_old_page_event	26	70. sided	6
26. end_text	9	71. sign_change	29
27. field_exists 	30	72. spaces 	10
28. field 	30	73. start_new_page_event 	26
29. font	9	74. store_format	24
30. goto	22	75. store_lookup 	30
31. graphic_fetch 	17	76. store	15
32. graphic_form	17	77. subtract 	29
33. graphic_height 	18	78. system_move_next_record	24
34. graphic_raw	18	79. system_new_pass 	24
35. graphic_width	18	80. system_set_limit	24
36. graphic	18	81. system_stop	24
37. if 	15	82. text_alignment 	10
38. left_margin	7	83. time	25
39. line_space	9	84. timeout 	32
40. line 	19	85. top_margin	8
41. log 	23	86. underline 	11
42. move_first_lookup 	31	87. x 	12
43. move_next_lookup	31	88. y 	12
44. multiple_record_field	31	89. yt	13
45. multiply 	29		

Please note: All measurements are in Postscript point units. 72 points per inch.

Print Job Commands

|banner_page| 1 2 3 include banner page at the front of the output

Parameters:

- | | | |
|---|---|--------------------|
| 1 | any value or numeric variable or field name | banner page width |
| 2 | any value or numeric variable or field name | banner page height |
| 3 | any value or paragraph variable or field name | banner page color |

Example:

```
|banner_page| 612 792 white  
|banner_page| 612 1004 yellow  
|banner_page| 1224 792 yellow  
|banner_page| #44| #45| 123|  
|banner_page| |field| mysize_x |field| mysize_y |field| mycolor
```

|paper_color| 1 set paper color value

Parameters:

- | | | |
|---|---|-------------|
| 1 | any value or paragraph variable or field name | paper color |
|---|---|-------------|

Example:

```
|paper_color| yellow  
|paper_color| letterhead  
|paper_color| |field| mypaper
```

|paper_size| 1 2 set paper size values (72 points per inch)

Parameters:

- | | | |
|---|---|--------------|
| 1 | any value or numeric variable or field name | paper width |
| 2 | any value or numeric variable or field name | paper height |

Example:

```
|paper_size| 612 792 letter portrait (8.5" X 11")  
|paper_size| 792 612 letter landscape  
|paper_size| 612 1004 legal portrait  
|paper_size| 792 1224 tabloid landscape  
|paper_size| |field| mysize_x |field| mysize_y
```

|paper_1| 1 2 3 4 5 6 set paper stock 1 values

Parameters:

1	any value	paper media name
2	any value	paper width
3	any value	paper height
4	any value	paper weight
5	any value	paper color
6	any value	paper type

Example:

|paper_1| Plain 612 792 0 white () e.g. letter size on plain white paper
|paper_1| Letterhead 612 792 75 buff ()
|paper_1| Light 792 612 244 blue (CorpLogo)

|paper_2| and |paper_3| and |paper_4| and |paper_5| and |paper_6| 1 2 3 4 5 6

same parameters as |paper_1|

Parameters:

1	any value	paper media name
2	any value	paper width
3	any value	paper height
4	any value	paper weight
5	any value	paper color
6	any value	paper type

Example:

|paper_1| Plain 612 792 0 white () e.g. letter stock on plain white
|paper_2| Letterhead_marketing 612 792 5 buff ()
|paper_3| ExtraLight 792 612 244 blue (CorpLogo)
|paper_4| Plain 612 792 0 white ()
|paper_5| MyLetterhead 612 792 75 buff ()
|paper_6| Heavy 792 612 244 pink (Logo)

|sided| 1 set sided value

Parameters:

1 single or double or double_tumble or paragraph variable or field name

Example:

|sided| single
|sided| double
|sided| double_tumble
|sided| |field| mysided

Page Commands

|bottom_margin| 1 set bottom margin value

Parameters:

1 any value or numeric variable or field name bottom margin

Example:

|bottom_margin| 72 1 inch from bottom edge of letter page
|bottom_margin| [#347]
|bottom_margin| |field| my_bottom_margin_setting

|left_margin| 1 set left margin value

Parameters:

1 any value or numeric variable or field name left margin

Example:

|left_margin| 72 1 inch from left edge of letter page
|left_margin| [#44]
|left_margin| |field| my_left_margin_setting

|new_page| insert new page

Example:

|new_page|

|right_margin| 1 set right margin value

Parameters:

1 any value or numeric variable or field name right margin

Example:

|right_margin| 540 1 inch from right edge of letter page
|right_margin| [#44]
|right_margin| |field| my_right_margin_setting

|top_margin| 1 set top margin value

Parameters:

1 any value or numeric variable or field name top margin

Example:

|top_margin| 720 1 inch from top edge of letter page

|top_margin| #47

|top_margin| |field| my_right_margin_setting

Text Commands

|case| 1 2 3 change case from paragraph variable 2 into paragraph variable 3

Parameters:

- 1 upper or lower or title or paragraph variable or field name
- 2 field name or paragraph variable (501 – 899)
- 3 paragraph variable (501 – 899)

Example:

```
|case| upper |503| |503|          UPPER CASE
|case| lower |643| |603|         lower case
|case| title |field| salute |763| Title Case
|case| |field| case |622| |622|
```

|end_text| places last line of text block. Used for center or right alignments

Example:

```
|end_text|
```

|font| 1 set font value

Parameters:

- 1 any value or paragraph variable or field name font name

Example:

```
|font| Courier_mm
|font| ArialMT_mm
|font| |23|
|font| |field| my_font
```

|line_space| 1 set line space value

Parameters:

- 1 any value or numeric variable or field name line space

Example:

```
|line_space| 7
|line_space| 11.003
|line_space| |#55|
|line_space| |field| my_line_spacing_value
```

|new_line| insert new line

Example:

|new_line|

|no_space| insert no space

Example:

|no_space|

|point_size| 1 set point size value

Parameters:

1 any value or numeric variable or field name point size

Example:

|point_size| 9
|point_size| 11.123
|point_size| 36
|point_size| |#55|
|point_size| |field| my_point_size

|spaces| 1 add spaces - no line wrap

Parameters:

1 any value or numeric variable or field name spaces

Example:

|spaces| 13
|spaces| |#12|
|spaces| |field| my_spacing

|text_alignment| 1 set text alignment value

Parameters:

1 left or center or right or decimal or justify or paragraph variable or field name

Example:

|text_alignment| justify
|text_alignment| left
|text_alignment| |38|
|text_alignment| |field| text_alignment

|underline| 1 turn underlining on or off

Parameters:

1 on or off or paragraph variable or field name

Example:

|underline| on

|underline| |3|

|underline| |field| underlining

Position Commands

|x| 1 set current x axis value

Parameters:

1 any value or numeric variable or field name x axis

Example:

```
|x| 72
|x| 200.005
|x| [#12]
|x| [field] my_new_x
```

|y| 1 set current y axis value

Parameters:

1 any value or numeric variable or field name y axis

Example:

```
|y| 645
|y| 1112.02
|y| [#12]
|y| [field] my_new_y
```

|+x| 1 add value to current value of x

Parameters:

1 any value or numeric variable or field name number

Example:

```
|+x| 5
|+x| -123.485
|+x| [#12]
|+x| [field] add_this_x
```

|+y| 1 add value to current value of y

Parameters:

1 any value or numeric variable or field name number

Example:

|+y| -1
|+y| 12.034
|+y| [#12]
|+y| [field] add_this_y

|yt| 1 set current y axis value from the top of the page defined by #942

Parameters:

1 any value or numeric variable or field name or number

Example:

|store| [#942] 792
|yt| 92 - y = 700

|yt| [#12]
|yt| [field] from_top_y

|new_origin| 1 2 set a new page origin for x axis and y axis

Parameters:

1 any value or numeric variable or field name x axis
2 any value or numeric variable or field name y axis

Example:

|new_origin| 100 200
|new_origin| 156.374 1058.049
|new_origin| [#11] [#12]
|new_origin| [#11] [field] new_y

Data Manipulation Commands

|#1| to [#999| print contain of numeric variable

Example:

```
|#1|  
|#234|
```

|1| to [999| print contain of paragraph variable

Example:

```
|2|  
|1000|
```

|append| 1 2 3 append paragraph variables 1 and 2 into paragraph variable 3

Parameters:

- | | | |
|---|--------------------------------------|--------------------|
| 1 | any paragraph variable or field name | paragraph variable |
| 2 | any paragraph variable or field name | paragraph variable |
| 3 | any paragraph variable or field name | paragraph variable |

Example:

```
|append_paragraph| |23| |34| |57|  
|append_paragraph| |field| first_name |field| last_name |field| full_name
```

|array_in| 1 2 store value 2 into array at element 1.

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name

Example:

```
|array_in| 1 7 store 7 at element 1  
|array_in| [#100| [#101|  
|array_in| |field| element |field| my_value
```

|array_out| 1 2 restore value from array at element 1 into numeric variable 2

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name

Example:

```
|array_out| 1|#7| store first element into|#7|  
|array_out||#100|#101|  
|array_out| |field| element |field| my_paragraph
```

|if| 1 2 3 4 5 if statement with true and false result

Parameters:

- 1 any value or numeric variable or paragraph variable or field name
- 2 =,<,>,<>,<=,>= or any paragraph variable or field name
- 3 any value or numeric variable or paragraph variable or field name
- 4 any paragraph variable or field name processed if true
- 5 any paragraph variable or field name processed if false

Example:

```
|if||#44|<45|#100|#0| |0| means, to do nothing  
|if||#836|=Courier_mm|#100|#101|  
|if||#121|<>empty|#0|#256| empty is a testable state  
|if||#121|<>|#111|#field|true|#field|or_false
```

|pad| 1 2 3 pad a value in 1 with string in 2 and stores new value in 3

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or paragraph variable or field name
- 3 any paragraph variable

Example:

```
|pad|123456FFFFFFFF|#120| |120|=FF123456  
|pad||#120|00000000|#120|  
|pad| |field| my_value FFFFFFFFFFFFFFFFFF|#620|
```

|store| 1 2 store variable into 1 from 2. See System Reserved Numeric and Paragraph Variables on page 35. Paragraph 1 to 499 are static fixed sized. Paragraph 500 to 1000 are variable sized. Use these if your paragraph needs to grow it's word size during Composition.

Parameters:

- 1 any value or numeric variable or paragraph variable or field name
- 2 any value or numeric variable or paragraph variable or field name

Example:

```
|store||#2|378  
|store||#2|#23|
```

|store| |2| fixed size, will not grow

|store| |502| Dear |field| name |no_space| , |new_line| |field| very_big_paragraphs

Graphical Commands

|box_border| 1 2 3 insert box border

Parameters:

- | | | |
|---|---|--------------------|
| 1 | any value or numeric variable or field name | box width |
| 2 | any value or numeric variable or field name | box height |
| 3 | any value or numeric variable or field name | box line thickness |

Example:

```
|box_border| 30 50 2  
|box_border| 1 |#21| |#111|  
|box_border| |field| my_width |field| my_height |field| my_line_thickness
```

|box_fill| 1 2 insert box with fill

Parameters:

- | | | |
|---|---|------------|
| 1 | any value or numeric variable or field name | box width |
| 2 | any value or numeric variable or field name | box height |

Example:

```
|box_fill| 300 450  
|box_fill| |#12| |#222|  
|box_fill| |field| my_width |field| my_height
```

|graphic_fetch| 1 insert Tiff graphic at current x and y position – hard insertion of file on every call. Use this command for single calls within the composition run. (This command is an Optional Feature)

Parameters:

- | | | |
|---|---|----------------------------------|
| 1 | any value or paragraph variable or field name | Complete graphic path, no spaces |
|---|---|----------------------------------|

Example:

```
|graphic_fetch| S:\our_images\tiff\cheque_image_2345654.tif  
|graphic_fetch| |54|
```

|graphic_form| 1 insert graphic at current x and y position. Use this command for multiple calls within the composition run. File is embedded only once.

Parameters:

- | | | |
|---|---|---------------------|
| 1 | any value or paragraph variable or field name | EPS filename prefix |
|---|---|---------------------|

Example:

```
|graphic_form| page_background  
|graphic_form| |54|  
|graphic_form| |field| signature
```

|graphic_height| 1 2 store height value of graphic 1 into numeric variable 2

Parameters:

- 1 any value or paragraph variable or field name EPS filename prefix
- 2 any numeric variable

Example:

```
|graphic_height| manager_photo |#14| | |
|graphic_height| |35| |#14|  
|graphic_height| |field| photo |#433|
```

|graphic_raw| 1 insert graphic at bottom left corner of page – hard insertion of file on every call. Use this command for single calls within the composition run

Parameters:

- 1 any value or paragraph variable or field name EPS filename prefix

Example:

```
|graphic_raw| company_signature  
|graphic_raw| |54|  
|graphic_raw| |field| signature
```

|graphic_width| 1 2 store width value of graphic 1 into numeric variable 2

Parameters:

- 1 any value or paragraph variable or field name EPS filename prefix
- 2 any numeric variable

Example:

```
|graphic_width| photo |#14| | |
|graphic_width| |88| |#14|  
|graphic_width| |field| my_photo |#134|
```

|graphic| 1 insert graphic at current x and y position – hard insertion of file on every call. Use this command for single calls within the composition run

Parameters:

1 any value or paragraph variable or field name

EPS filename prefix

Example:

```
|graphic| company_signature  
|graphic| [54]  
|graphic| [field] signature
```

|line| 1 2 3 4 5

insert line

Parameters:

1 any value or numeric variable or field name
2 any value or numeric variable or field name
3 any value or numeric variable or field name
4 any value or numeric variable or field name
5 any value or numeric variable or field name

line length
line thickness
line angle
line style
1 - solid, 2 - dashed, 3 - dotted
move x and y pointer to line
endpoint
0 - no, 1 - yes

Example:

```
|line| 30 2 45 3 1  
|line| [#44] 1.003 -225 1 0  
|line| [#44] [#45] [#46] [#47] [#48]  
|line| [field] my_width [field] my_line_thickness [field] my_angle [field] my_style 1
```

|rotate| 1

rotate degrees

Parameters:

1 0...360 or -360 or numeric variable or field name angle degrees

Example:

```
|rotate| 45.5  
|rotate| -135.01  
|rotate| [#347]  
|rotate| [field] process_rotation
```

|reset_graphic_state|

reset scale to 1 1, rotation to 0 and origin to bottom left corner of page

Example:

```
|reset_graphic_state|
```

|scale| 1 2

scale x y axis

Parameters:

- 1 any value or numeric variable or field name x axis
- 1 any value or numeric variable or field name y axis

Example:

```
|scale| 2 5  
|scale| 0.5 2.321  
|scale| |#347| |#348|  
|scale| |field| process_scale_x |field| process_scale_y
```

Color Commands

|color| 1 2 3 4 5 set toner / ink color

Parameters:

- 1 any value or numeric variable or field name 1 - transparent, 2 - gray, 3 - rgb, 4 - cmyk
- 2 any value or numeric variable or field name
- 3 any value or numeric variable or field name
- 4 any value or numeric variable or field name
- 5 any value or numeric variable or field name

if parameter 1 = 1, pad rest with 0 0 0 0, (padded values will be ignored)

if parameter 1 = 2, one parameter 0 - white, 50 - gray, 100 - black
pad rest with 0 0 0, (padded values will be ignored)

if parameter 1 = 3, three parameters
0 ...100 red - unsaturated 0, saturated 100
0 ...100 green - unsaturated 0, saturated 100
0 ...100 blue - unsaturated 0, saturated 100
pad last with 0, (padded value will be ignored)

if parameter 1 = 4, four parameters
0 ...100 cyan - unsaturated 0, saturated 100
0 ...100 magenta - unsaturated 0, saturated 100
0 ...100 yellow - unsaturated 0, saturated 100
0 ...100 black - unsaturated 0, saturated 100

Example:

color 1 0 0 0 0	transparent
color 2 100 0 0 0	black and white – black
color 2 10 0 0 0	black and white – light gray
color 2 55 0 0 0	black and white – medium gray
color 3 100 0 0 0	rgb - red
color 3 0 100 0 0	rgb - green
color 3 0 0 100 0	rgb - blue
color 3 #77 77 #87 0	rgb
color 4 100 0 0 0	cmyk – cyan (bluish green)
color 4 0 100 0 0	cmyk – yellow
color 4 0 0 100 0	cmyk – magenta (bluish violet)
color 4 0 0 0 100	cmyk – black
color 4 #77 #77 #87 #97	cmyk
color field color_#1 field color_#2 field color_#3 field color_#4 field color_#5	

System Commands

|| start or end of comment section. User can select their own delimiter for each Project

Example:

```
|x||#111|  
|| comment - set x position to value in numeric variable #111 ||
```

|bookmark| 1 destination label for goto command

Parameters:

1 any value or numeric variable or field name 1 to 100 bookmarks allowed

Example:

```
|bookmark| 22  
|bookmark||#345|  
|bookmark| |121|  
|bookmark| |field| mybookmark_number
```

|date| 1 replaced with system date

Parameters:

1 1 or 2 or 3 date format

Example:

```
|date| 1 Tuesday, March 05, 2002  
|date| 2 Mar 05 2002  
|date| 3 3/5/02
```

|goto| 1 standard goto command to designated bookmark number

Parameters:

1 any value or numeric variable or paragraph variable or field name

Example:

```
|goto||#100|  
|goto| 3  
|goto| |545|  
|goto| |field| go_back
```

|log| 1 add Tab delimited info to Log.txt File in your Data folder under Project

Parameters:

1 any value or numeric variable or paragraph variable or field name

Example:

```
|log| [#100|
|log| |345|
|log| new_line          insert carriage return
```

|new_start| create a new looping starting point in Dynamic File

Example:

```
|store| [#100| 0
|new_start|
|add| 1 |#100| [#100|          set value in numeric variable 100 to 0, which will never resets
                                again. Used for OMR and 3 of 9 wrap sequence
```

|postscript_new_line| insert new line into postscript output file

Example:

```
|postscript_pass_through| on
%% Here are my Comments |postscript_new_line|
300 400 100 0 270 arc 10 setlinewidth stroke
|postscript_pass_through| off
```

|postscript_pass_through| 1 pass postscript code through application

Parameters:

1 any paragraph variable or field name on or off

Example:

```
|postscript_pass_through| on
300 400 100 0 270 arc 10 setlinewidth stroke
|postscript_pass_through| off
```

```
|postscript_pass_through| on
300 400 100 0 270 arc 10 setlinewidth stroke
|postscript_pass_through| off
```

```
|postscript_pass_through| on
|700| |800| |801|          any paragraph variable between 500 and 900
|postscript_pass_through| off
```

|restore_format| restore all text variable values

Example:

|restore_format|

|store_format| store all text variable values

Example:

|store_format|

|system_set_limit| 1 maximum number of words and commands per set.
Default value 40000

Parameters:

1 any value or numeric variable or field name array size as integer

Example:

|system_set limit| 1000000

|system_new_pass| configures system for additional Table / Query 1 data pass. Resets cursor position back to the first record. This is used for multi-pass mode.

Example:

|system_new_pass|

|system_move_next_record| moves Table / Query 1 record cursor to next record. This is used for multi-pass mode.

Example:

|system_move_next_record|

|system_stop| invokes hard Composition stop. This is used for multi-pass mode.

Example:

|system_stop|

|time| 1

replaced with system time

Parameters:

1 any value or numeric variable or field name

1 or 2 or 3

Example:

|time| 1

1:11:55 PM

|time| 2

1:11 PM

|time| 3

13:11

Event Commands

|end_old_page_event| last event invoked on every page. Commands must be on one line (no carriage return).

Example:

```
|end_old_page_event| |x| 300 |y| 60 Page |#234| of |#543|
```

|start_new_page_event| first event invoked at start of every new page. Commands must be on one line (no carriage return).

Example:

```
|start_new_page_event| |x| 0 |y| 792 |graphic_form| page_background |701| |755|
```

|end_line_event| event invoked at the end of every line. Commands must be on one line (no carriage return).

Example:

```
|end_line_event| |new_line|
```

Math Commands

|absolute| 1 2 absolute value in 1 into 2

Parameters:

- 1 any value or numeric variable or field name
- 2 any numeric variable

Example:

```
|absolute| #72| #72| | |
|absolute| -2|#362|
|absolute| |field| my_number|#2|
```

|add| 1 2 3 add 1 and 2 and store the sum value into 3

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name
- 3 any numeric variable

Example:

```
|add| 1 4|#72|          |#72|= 5
|add| #23|#658|#199|
|add| |field| my_number |field| my_number2|#299|
```

|dec_ascii| 1 2 converts the decimal number 1 into the ANSI character value 2

Parameters:

- 1 any value or numeric variable or field name
- 2 any paragraph variable

Example:

```
|dec_ascii| #120|120| | |
|dec_ascii| |field| ansi|620|
|dec_ascii| 65|148|
```

|dec_hex| 1 2 converts the decimal number 1 into a hexadecimal value 2

Parameters:

- 1 any value or numeric variable or field name
- 2 any numeric variable

Example:

```
|dec_hex| #120| #120| | |
|dec_hex| |field| my_base_ten |#620|  
|dec_hex| 23 |#148|
```

|dec_oct| 1 2 converts a decimal number 1 into an octal number value 2

Parameters:

- 1 any value or numeric variable or field name
- 2 any numeric variable

Example:

```
|dec_oct| #120| #120| | |
|dec_oct| |field| my_base_ten |#620|  
|dec_oct| 34299 |#148|
```

|decimal_places| 1 2 3 set the number of decimal places in 3

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name 0 to 5 decimal places
- 3 any numeric variable

Example:

```
|store| #72| 3  
|decimal_places| #72| 0 |#82| #82| = 3  
|decimal_places| #72| 1 |#82| #82| = 3.0  
|decimal_places| 384.3732 1 |#82| #82| = 384.4  
|decimal_places| |field| my_number 5 |#2|
```

|divide| 1 2 3 divide 1 by 2 and store the quotient value into 3

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name
- 3 any numeric variable

Example:

```
|divide| 12 4 |#72| #72| = 3  
|divide| #23| #658| #1|  
|divide| |field| my_number |field| my_number2 |#299|
```

|multiply| 1 2 3

multiply 1 and 2 and store the product value into 3

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name
- 3 any numeric variable

Example:

```
|multiply| 12 4 |#72|          |#72| = 48  
|multiply| |#23| |#658| |#11|  
|multiply| |field| my_number |field| my_number2 |#299|
```

|remainder| 1 2 3

divide 1 by 2 and stores the remainder value into 3

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name
- 3 any numeric variable

Example:

```
|remainder| 5 2 |#1|          |#1| = 1  
|remainder| |#1| |#2| |#67|  
|remainder| |field| my_number |#33| |#1|
```

|sign_change| 1 2

change the sign of value in 1 from + to - or - to + into 2

Parameters:

- 1 any numeric variable
- 2 any numeric variable

Example:

```
|sign_change| |#72| |#72|
```

|subtract| 1 2 3

subtract 2 from 1 and store the difference value into 3

Parameters:

- 1 any value or numeric variable or field name
- 2 any value or numeric variable or field name
- 3 any numeric variable

Example:

```
|subtract| 11 4 |#72|          |#72| = 7  
|subtract| |#23| |#658| |#199|  
|subtract| |field| my_number |field| my_number2 |#299|
```

Database Commands

|field| 1 print value in field. spaces not allowed

Parameters:

1 any value field name

Example:

```
|field| firstname  
|field| last_name
```

|field_exists| 1 2 3 check to see if field name 1 exists with true and false result

Parameters:

1 field name
2 any paragraph variable or field name processed if true
3 any paragraph variable or field name processed if false

Example:

```
|field| zip_mark |100| |200|  
|field| select_feeder_3 |600| |0|  
|field| select_feeder_6 |field| if_true |field| if_false
```

|store_lookup| 1 2 3 4 5 lookup data in table 1 where “in” field name 2 is equal to value 3, then slide over to “out” field name 4 and copy that cell value into paragraph variable 5

Parameters:

1 any value or paragraph variable or field name table name
2 any value or paragraph variable or field name “in” field name
3 any value or paragraph variable or field name value
4 any value or paragraph variable or field name “out” slide field name
5 any paragraph variable or field name paragraph variable

Example:

```
|store_lookup| 2 account_number |field| account_number first_name |27| | |
|store_lookup| |field| sec_table ID |field| record_number total |44|  
|store_lookup| 3 record_id 1239283 other_data |27|
```

|multiple_record_field| 1 initialize multiple record field and store flag field variable and value in that field. No nested limit.

Parameters:

1 any value flag field name or grouping name

Example:

|multiple_record_field| acct_number

|next_record| move to the next record and compare all the values with all the corresponding values stored in the flag fields. If the next record has the same values in all the flag fields, loop back to last multiple record field call, else close last multiple_record_field and pass through.

Example:

|next_record|

|move_first_lookup| 1 move pointer to first record in lookup Table / Query #

Parameters:

1 any value or numeric variable or paragraph variable or field name

Example:

|move_first_lookup| 2
|move_first_lookup| 4
|move_first_lookup| |2|
|move_first_lookup| |#33|
|move_first_lookup| |field| myquery

|move_next_lookup| 1 move pointer to next record in lookup Table / Query #

Parameters:

1 any value or numeric variable or paragraph variable or field name

Example:

|move_next_lookup| 2
|move_next_lookup| 3
|move_next_lookup| |2|
|move_next_lookup| |#33|
|move_next_lookup| |field| mytable

|timeout| 1

database connection timeout. Default value is 10

Parameters:

1 any numeric value

Example:

|timeout| 10

|timeout| 50

Automation Commands

The Application can now auto-compose any existing project based on a overwriteable script file.

Script File Path: (your windows folder)\mmsc_auto_compose.scp

Commands:

compose – 3 parameters required

- 1) Comment text of the project to indicate which project to compose (first instance found)
- 2) Destination Postscript file
- 3) Record count from table/query 1 (number or all)

task – 2 parameters required

- 1) Name of the program to run via scheduled task
- 2) Seconds from now to begin task (number -120 minimum)

Sample Script File:

```
compose
Billing Statements comment
C:\ds_data\data_parser\automation\postscript\Billing_7_1_2005.ps
all
task
C:\ds_data\data_parser\automation\configure\auto_parser.bat
120
compose
Accounts Statements
C:\ds_data\data_parser\automation\postscript\Accounts_2005.ps
300
task
C:\ds_data\data_parser\automation\configure\Accounts.bat
180
```

System Reserved Variables

#900	Top Margin	#950	**System Page Count
#901	Left Margin	#960	System Pass
#902	Right Margin	#965	**Lookup Indicator
#903	Bottom Margin	#970	Postscript_Pass_Through
#909	Color	#975	**Highest Element in Array
#910	Gray	#980	Record Number Table / Query 1
#911	Cyan	#981	**Record Count Table / Query 1
#912	Yellow	#982	Record Number Table / Query 2
#913	Magenta	#983	**Record Count Table / Query 2
#914	Black	#984	Record Number Table / Query 3
#915	Red	#985	**Record Count Table / Query 3
#916	Green	#986	Record Number Table / Query 4
#917	Blue	#987	**Record Count Table / Query 4
#920	Font	491	Paper 1 Default Paper Stock
#921	Point Size	492	Paper 2
#923	Line Space	493	Paper 3
#924	Underline	494	Paper 4
#925	Space Width	495	Paper 5
#926	Rotate	496	Paper 6
#927	Scale X axis	497	End_Line_Event
#928	Scale Y axis	498	End_Old_Page_Event
#929	First word on a new line flag	499	Start_New_Page_Event
#930	Orphan	993	Paper Size
#940	X Position	994	Paper Color
#941	Y Position	995	Font Name
#942	yt command Position	996	Text Alignment

** indicates read-only in normal usage.

System Reserved Commands

|system_end_static|
|system_close|
|system_next_record|
|system_np|
|system_sp|
|system_return|
|system_static|

These Commands should not be used in your Dynamic File.

Mail Merge and Statement Composer Commands

Quick Reference

PRINT JOB COMMANDS	5
PAGE COMMANDS	7
TEXT COMMANDS	9
POSITION COMMANDS	12
DATA MANIPULATION COMMANDS	14
GRAPHICAL COMMANDS	17
COLOR COMMANDS	21
SYSTEM COMMANDS	22
EVENT COMMANDS	26
MATH COMMANDS	27
DATABASE COMMANDS	30
AUTOMATION COMMANDS	33
SYSTEM RESERVED VARIABLES	34
SYSTEM RESERVED COMMANDS	35