

Data

Merger

Copyright Notice

Data - Merger is Copyright (c) 1990, by Keith J. Alphonso and is licenced to Alpha Software Technologies. The unauthorized duplication of this software or the accompanying documentation is PUNISHABLE UNDER FEDERAL LAW. Anyone in violation of this copyright agreement will be prosecuted to the full extent of the law.

The only authorized copies of this software are those made for backup puposes only. Backup copies may only be retained by the owner of the software, and cannot be distributed in any fashion.

WARNING: Your serial number is attached to the Data - Merger executable files. This serial number cannot be changed, and can be linked directly to you in the event that an unauthorized copy is found. DO NOT DISTRIBUTE THIS SOFTWARE!

Disclaimer

This software is designed to be reliable, easy to use, and safe; however, the loss of data is always a possiblity. The maker of this software, namely Alpha Software Technologies and Keith J. Alphonso, are not responsible for the loss of data caused directly, or indirectly by the use of this software.

Data - Merger

Copyright (c) 1990 By Keith J. Alphonso
Licenced to: Alpha Software Technologies
All Rights Reserved

Data – Merger

I. Introduction	1
A. What is Data – Merger	1
B. How can I use Data – Merger	1
II. Installation	2
III. Description	3
A. Set up database	3
B. Make document in Word – Processor	3
C. Process through Data – Merger	3
D. Format/Output to printer or file	4
IV. The Data – Merger Document	5
A. Configuration Commands	7
1. @DEF	7
2. @TRIM/@NOTRIM	8
3. @WRAP/@NOWRAP	9
B. Text Replacement Commands	10
1. @PUT	10
2. @TOTAL	11
3. @PROMPT	12
4. @CENTER	13
5. @EJECT	14
6. @CODE	15
7. @LINE	16
C. Control Commands	17
1. @IF/@ENDIF	17
2. @WHILE/@ENDWHILE	18
3. @NEXT/@PREV/@FIRST/@LAST	19
4. @PAUSE	20
5. @CLEAR	21
V. Running Data – Merger	22
A. Syntax	22
B. Prompts	22
1. Output Path	22
2. Keyfield	22
3. Text Expression	22

Introduction

Data - Merger is a document processing program that allows you to incorporate Data - Windows data into your word-processor documents. The document can be created with any OS9 word processor (such as Dynastar, Window-Writer, TsWord, etc.). The document can then be processed with Data - Merger to insert the database data. The Data - Merger can print out a separate copy of the document for each record in the database.

Document merging is certainly nothing new, most COCO word processors have some sort of mail-merging facility. The problem is that the data cannot be retrieved from a database package, it must be re-typed for the document. With Data - Merger you will be able to have all data in one place, and use that data in documents.

Another use for Data - Merger is the creation of invoices. An invoice can be created and printed for each customer in the database or for just a single customer. This gives quite a lot of flexibility in the output of Data - Windows data.

The Data - Merger set of commands is also flexible enough to use Data - Merger as a complex report generator. The commands include commands for skipping between records and totalling fields.

Installation

Data – Merger installation is relatively simple. You should make a backup, then copy the Data – Merger executable file to the commands directory of your Data – Windows disk.

To make a backup, follow this procedure:

Boot up OS9

type:backup /d0 /d0 #56k

follow the prompts.

After you have a backup, you can copy the Data – Merger executable file to the CMDS directory of your Data – Windows disk.

Single Drive Users:

Place DMerger disk in drive 0

type:copy -s /d0/cmds/DMerger /d0/cmds/DMerger #56k

Dual Drive Users:

Place DMerger disk in drive 0

Place Data - Windows disk in drive 1

type:Copy /d0/cmds/DMerger /d1/cmds/DMerger #56k

Hard Drive Users:

Place DMerger disk in drive 0

type:Copy /d0/cmds/DMerger /h0/cmds/DMerger #56k

Note:It is not required that Data – Merger reside on the Data – Windows disks, this just makes it more convenient to use. You can use DMerger from the CMDS directory of any disk, even the distribution disks.

Description

To use Data – Merger you must follow a four step procedure, create the database, create the document, Data – Merge the document, and finally, format and print the document.

Creating the database is done with Data – Windows. You can refer to the instructions in the Data – Windows manual for more information on creating and using databases. One thing to remember is that anything that can be put in a Data – Windows output (PUT) field, can be put in a Data – Merger document. This gives you all the power of using expressions for data output.

Creating the Data – Merger document is done with your favorite word-processor. A Data – Merger document is no different from any other document, except that the document may contain embedded Data – Merger commands. The Data – Merger commands will be removed from the document by Data – Merger, and will be replaced by the appropriate action. The appropriate action could be placing data into the document (such as a name or amount).

After the document has been created, it can be "Data – Merged." The process of executing Data – Merger will be explained later. When completed, you will have a file that contains a copy of the document for each record in your database. Any data fields that you specified to be placed in the document will be filled with the actual data.

Once the document has been merged, you can format and print the document to your printer. The process of formatting and printing the final document are different for different word-processors. Basically there are two types, all-in-one word processors (like DeskMate and WindowWriter), and combination editor/text formatters (like TsWord and DynaStar). For all-in-one word processors, simply load the newly created merge file and print. For editor/text formatter combinations, simply perform the text formatting process on the new merged file.

An interesting note is that the reformatting step may not be required. Data – Merger will ignore any special print codes that may be embedded in the file, and will pass them directly to the output. Because of this it may be possible to PRINT the document to disk, then Data – Merge this "print" file and send it directly to the printer.

The only time this can cause problems is when the size of the output

data can corrupt the formatting process. This occurs in situations such as text centering and word wrapping (because the word-processor does not know the size of the data, it cannot format it properly). Special Data – Merger formatting commands were included to compensate for this problem. Data – Merger does NOT purport to be a text formatter, only a mail-merger, these extra commands were added for user convenience.

The Data – Merger Document

You create the Data – Merger document from within your favorite word-processor or editor. In this document you put Data – Merger commands that will NOT show up on the output (printer), but will cause Data – Merger to perform some action. Data – Merger commands will not interfere with any printing codes also embedded in the document.

The following section describes each Data – Merger command and gives examples of how they will be used. The commands are organized into Configuration Commands, Text Replacement Commands, and Control Commands.

At first, the number of commands may make you think that Data – Merger is difficult to use...it is not. The truth is that you may never need many of the commands provided, but if you do need them they are there!

The commands that will be used the most are @DEF and @PUT, these are the two basic define data and place data commands. Special formatting etc. can be performed with other commands, but this may not be necessary, as you can perform this formatting in your word-processor. You probably will not need the configuration commands (except for @DEF) as the defaults are usually what will be needed.

Quick List

Configuration Commands

@DEF–Define an output data element
@TRIM/@NOTRIM–Enable/Disable field blank space padding
@WRAP/@NOWRAP–Enable/Disable word wrap handling.

Text Replacement Commands

@PUT–Insert a data element at this position
@TOTAL–Insert a data element total at this position
@PROMPT–Prompt the user for text to be inserted here
@CENTER–Center a data element on the current line
@EJECT–Eject a page at this point in the text
@CODE–Insert a character code at this position
@LINE–Insert an end of line at this position

Control Commands

@IF/@ENDIF—Define a conditional print block
@WHILE/ENDWHILE—Define a repeating print block
@NEXT—Go to next data record
@PREV—Go to previous data record
@FIRST—Go to first data record
@LAST—Go to last data record
@PAUSE—Pause output/printing at this point
@CLEAR—Clear a data element total

Configuration Commands

Configuration commands allow you to configure some part of the Data – Merger system. These commands are stripped from the input and do not show any result on the output, but change some state of the Data – Merger program.

@DEF – Define Data

The @DEF command is used to define a data element that will be used later in the document. You use the @DEF command once for each database expression that you want to place in the document. Each @DEF command specifies a number to be used when referencing that expression later in the document. This command is used in conjunction with the @PUT, @TOTAL, and @BREAK commands. Note that the @DEF command should be on a line by itself so as not to interfere with any other text.

The syntax for the @DEF command is as follows:

@DEF(n,<expression>)

n – Expression reference number (from 0 – 199)

<expression> – Any valid Data – Windows output expression

Here are some examples:

```
@DEF(1,first_name + " " + last_name)
@DEF(2,address1)
@DEF(3,address2)
@DEF(4,address3)
. . . later in the document . . .
From:Alpha Software Technologies
      210 Bluefield Dr.
      Slide11, LA. 70458

To:  @PUT(1,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
      @PUT(2,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
      @PUT(3,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
      @PUT(4,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")

Dear:@PUT(1,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
```

@TRIM/@NOTRIM – Enable/Disable picture trimming

The @TRIM and @NOTRIM commands allow you to enable and disable the Data – Merger field trim feature. Field trimming is the stripping of spaces from the end of an output expression. If you disable trimming (@NOTRIM), all output expressions will be the EXACT length of the picture string defined in the output command. If trimming is enabled (the default), all trailing blanks will be stripped from the field.

The syntax for @TRIM and @NOTRIM are as follows:

@TRIM

@NOTRIM

Here are some examples:

```
@DEF(0,first_name + " " + last_name)
@DEF(1,amount_due1)
@DEF(2,amount_due2)
@DEF(3,amount_due3)
. . . Later in the document
```

If you please,@PUT(0,"XXXXXXXXXXXXXXXXXXXX"), the following report will show you how much you owe us.

@NOTRIM

```
Time 1      Time 2      Time 3
@PUT(1,"999999.99")@PUT(2,"999999.99")@PUT(3,"999999.99"
)
```

@TRIM

@WRAP/@NOWRAP – Enable/Disable Data – Merger word wrapping

The @WRAP and @NOWRAP commands can be used to force Data – Windows to handle document word wrapping. This is useful as Data – Windows knows the size of the data and can handle the word wrapping. This may save the extra step of formatting the Data – Merger output before printing.

Note that it is not required that you use the Data – Merger word-wrapping, you may want to have the word-processor hand word wrapping AFTER Data – Merging has been performed.

The syntax for @WRAP and @NOWRAP is as follows:

@WRAP(<left>, <right>)

<left>– Character position of left margin

<right>– Character position of right margin

@NOWRAP

Here are some examples:

@WRAP (10, 70)

This text will be wrapped such that the left margin will be placed at character position 10 (1 inch) from the side of the page, and no word will exceed character position (1 inch from the right of the page).

@NOWRAP

Text Replacement Commands

Text Replacement Commands are commands that will be stripped from the input, and replaced with some new information on the output. This information can be information from the database, prompted from the user, or given as a constant.

@PUT – Output Expression

The @PUT command is used to output a previously defined expression. You give the @PUT command an expression number that was defined with @DEF and a picture definition to use for the output. Note that the output picture is used for data formatting and for a MAXIMUM element size, not a minimum. The output expression will NOT be padded with excess spaces unless the @NOTRIM command is used to force this. The @PUT command will be replaced by the defined expression on the output.

The syntax for the @PUT command is as follows:

@PUT(n,<picture>)

n – A previously defined expression number.

<picture>– Any valid Data – Windows picture string.

Here are some examples:

```
@DEF(0,first_name + " " + last_name)
@DEF(1,amount_due1)
@DEF(2,amount_due2)
@DEF(3,amount_due3)
```

. . . later in the document . . .

```
From:Alpha Software Technologies
      210 Bluefield Dr.
      Slidell, LA. 70458
```

```
To: @PUT(1,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
     @PUT(2,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
     @PUT(3,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
     @PUT(4,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
```

```
Dear: @PUT(1,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX")
```

@TOTAL – Output a total expression

The @TOTAL command is used to output the current value of the running total for the given expression number. Data – Merger keeps a running total for all numeric expressions that you define. This is useful when current totals are required. The total for each expression can be cleared with the @CLEAR command (described later). The picture string formatting for @TOTAL is exactly the same as that for @PUT.

The syntax for @TOTAL is as follows:

@TOTAL(n, <picture>)

n -- A previously defined expression number.

<picture>-- Any valid Data – Windows picture string.

Here are some examples:

```
@DEF (1, quantity)
@DEF (2, unitcost)
@DEF (3, quantity * unitcost)
```

. . . later in the document . . .

```
                Quantity      Unit Cost      Total
Your Values @PUT (1, "9999.99") @PUT (2, "9999.99") @PUT (3, "9999.99")
```

```
CurrentTot.@TOTAL (1, "9999.99")@TOTAL (2, "9999.99")@TOTAL (3, "9999.9"
)
```

@PROMPT – Prompt the user for text

The @PROMPT command is used to prompt the user to input his own text at some part of the document. This can be useful when the information needed to be put into some space is not known until the document is being printed (is not stored in the database). The picture string definition for this command will be used both to limit the user's input string to a specific length and format, as well as to format the output.

The syntax for the @PROMPT command is as follows:

@PROMPT(<prompt>, <picture>)

<prompt> –A prompt to display to the user (can contain any valid Data – Window expression).

<picture> –Any valid Data – Windows picture string

Here are some examples:

```
. . . Somewhere in the document . . .  
We appreciate the @PROMPT("Enter gift from " +  
first_name + " " + last_name,"XXXXXXXXXXXXXXXX") it will  
come in quite handy.  
. . .
```

@CENTER – Center An Expression On The Page

The @CENTER command can be used to center output data on the page. It can be useful for Data – Merger to center the data instead of the word-processor because Data – Merger already knows the size of the data. This may help you avoid the final step of formatting the data before printing.

The syntax for the @CENTER command is as follows:

@CENTER(n, <picture>, <width>)

n– Expression number to output

<picture>– A valid Data – Windows picture

<width>– The width of the page in which to center

Here are some examples:

```
@DEF(0, first_name + " " + last_name)
. . . Later in the document . . .
@CENTER(0, "XXXXXXXXXXXXXXXXXXXX", 80)
```


@EJECT – Eject A Page

The @EJECT command will insert a character 12 (form feed) into the document. This will cause the printer to eject the current page. This can be used to force a page break at any point in the document.

The syntax for the @EJECT command is as follows:

@EJECT

Here are some examples:

. . . Somewhere in the document . . .

Page 1

@EJECT

This is the beginning of page 2.

@CODE – Output A Special Code

The @CODE command can be used to embed any character code into the document. This is useful when you need to send a printer code or special character.

The syntax for @CODE is as follows:

@CODE(<code>)

<code>– The character code (in decimal) to send.

Here is an example:

```
. . . Somewhere in the document . . .  
This should be @CODE(27)@CODE(31)BOLDED on a DMP  
printer@CODE(27)@CODE(32)  
This should not.
```

@LINE – Output An End-of-line

The @LINE command can be used to force a line feed in the middle of your document. This is only useful between an @WRAP and @NOWRAP command, as while WRAP is in effect, normal end-of-line characters are ignored for word-wrapping.

The syntax for @LINE is as follows:

@LINE

Here is an example:

```
. . . Somewhere in the document. . .  
This is line 1@LINE  
This is line 2@LINE
```

Control Commands

Control commands are used to control the flow of the Data – Merger document. This can include skipping blocks of text, repeating blocks of text, pausing the printing, even changing the current state of the database.

@IF/@ENDIF – Conditional Printing

The @IF command, in conjunction with the @ENDIF command can be used to create conditional print blocks. Conditional print blocks allow you to determine, based on some database criteria, whether or not to print certain blocks of text.

The syntax for @IF/@ENDIF is as follows:

```
@IF(<condition>)  
... Some text ...  
@ENDIF
```

<condition>– Any valid Data – Windows test condition

Here are some examples:

```
... Somewhere in the document ...  
@IF(amount_due > 100.00)You MUST send the money  
IMMEDIATELY or some action will be taken.@ENDIF  
@IF(amount_due <= 100.00 and amount_due > 0)We would  
appreciate it if you would send the money as soon as  
possible.@ENDIF  
@IF(amount_due <= 0)Thank you for your payment.@ENDIF
```

@WHILE/@ENDWHILE – Repeating Print Blocks

The @WHILE and @ENDWHILE commands can be used to create repeating print blocks. The blocks of text will continue to print as long as the criteria given in the while statement is true. Somewhere inside the text block some action must be performed to cause the condition to fail. Usually this action will be some record changing command (such as @NEXT).

The syntax for @WHILE/@ENDWHILE is as follows:

```
@WHILE(<condition>)  
... Some Text ...  
@ENDWHILE
```

Here is an example:

```
. . . Somewhere in the text . . .  
@WHILE(amount_due > 0)Please send your payment  
of:@PUT(1,"99999.99")  
@NEXT  
@ENDWHILE
```

@NEXT/@PREV/@FIRST/@LAST – Database Record Selectors

The @NEXT/@PREV/@FIRST/@LAST commands allow you to reposition the current database record pointer. This is useful when you have a database in which each customer may have several records. These commands will allow you to output data from multiple records in a single document. NOTE: Total calculations will only be performed on the record that is currently active when the output data element is defined (@DEF statement).

The syntax for these commands is as follows:

```
@NEXT  
@PREV  
@FIRST  
@LAST
```

Here is an example:

```
@DEF(1,amt_due)  
. . . Somewhere in the document . . .  
Your 4 account balances are as follows:  
@PUT(1,"99999.99")@NEXT  
@PUT(1,"99999.99")@NEXT  
@PUT(1,"99999.99")@NEXT  
@PUT(1,"99999.99")@NEXT
```

@PAUSE – Pause the Data – Merger

The @PAUSE command can be used to halt the execution of Data – Merger within a document. This is useful for inserting a different type of paper in the printer, or for having the user perform some required action. The @PAUSE command can be embedded in an @IF/@ENDIF block for conditional pausing.

The syntax of @PAUSE is as follows:

```
@PAUSE(<message>)
```

<message> –A message to display to the user (can contain any valid Data – Window expression).

Here is an example:

... Somewhere in the document ...

```
@PAUSE("Processing " + first_name + " " + last_name)
```

@CLEAR – Clear a Data Element Total

The @CLEAR command can be used to reset the total for a pre-defined field. Data – Merger keeps a running total of all defined data element fields (@DEFs) that are numeric. It may be necessary to reset this total from within a document.

The syntax for @CLEAR is as follows:

@CLEAR(n)

n --The predefined data element number (@DEF number) to clear.

Here is an example:

```
. . . Somewhere in the document . . .
@WHILE(record_type = "S")
@DEF(0,amount_due)
Amount due: @PUT(0,"99999.99")
@ENDWHILE
Total: @TOTAL(0,"99999.99")
@CLEAR(0)
```


Running Data – Merger

Executing Data – Merger is quite simple. If you are using Multi-View, simply double-click on the Data – Merger ICON. If you are not using Multi-View, execute Data – Merger with the following syntax:

DMerger [<mergefile>] [<database>]

<mergefile> –The Data – Merger document file (created with a word-processor and containing embedded commands).

<database> –The Data – Windows database file to use for merging.

Note that both of these parameters are optional. If you omit one or both of the parameters you will be prompted for them with a standard Data – Windows filename window.

After it is run, Data – Merger will prompt you for the output path to use, a key field, and a test expression.

The output path is the name of the file or device to which you would like the merge file to be printed. If this field is left blank, the documents are printed to the standard output.

The key field is the name of a valid, pre-defined (in Data – Windows) key field to use for sorting. Leave this field blank if you do not want the documents to be sorted.

The test expression is a valid Data – Windows expression to test before printing the merge-document. This condition will be tested for each record in the database before inserting the data into the document. The following are some example conditions:

```
amount_due > 100
```

```
amount_due >=0 and amount_due <= 100.00
```

```
last_name = "HOOLIGAN"
```

Note that test conditions can be blank, thus processing is done on ALL records.

After entering the required data, Data – Merger will begin processing records. When completed, Data – Merger will exit.

If you printed the data to a file, you can then either text-process the file, or re-load the file into the word-processor for formatting. You can then print the file.