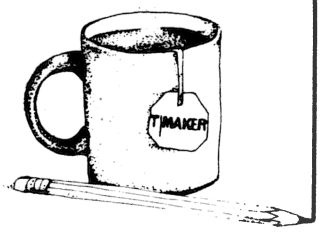
T|MUG

T/MAKER USER'S GROUP NEWSLETTER
VOLUME 4, NUMBER 1, JANUARY/FEBRUARY 1985

In This Issue:



T/Maker Users' Group Newsletter

T/MAKER NEWSFRONT

-- *Heidi Roizen*

Yet Another New Look for the T/MUG

Here we go again, folks, with another new look -- one which we hope will last for a while. The T/MUG is now being printed on the new HP LaserJet. All the typefonts you see in this issue are being controlled directly through T/Maker by using a print.utl file. Those of you with a LaserJet can send a stamped (for 2 oz) self-addressed envelope and we'll send you a print-out of the print.utl file for cartridge 92286-B -- the one used for this issue.

Europe on One/Data General a Day

Data General was kind enough to lend us one of their new Data General/One computers for a promotion trip to Europe. The machine is an IBM PC compatible, with two 3 1/2" drives, an 80 column by 25 line LCD screen, 256K, and a built-in battery pack which allows for disconnected operation up to twelve hours. The whole thing (including battery pack) weighs about twelve pounds. It was a terrific machine for the promotional tour, allowing us to give demos to anyone at any time, including the stewardesses during our flights. T/Maker is (naturally) now available for the DG/One through us directly!

Visit Us at the MacWorld Expo

T/Maker will be showing its Apple Macintosh products at the MacWorld Expo at Brooks Hall, San Francisco February 21-23. We'll not only be exhibiting the three ClickArts currently out, but also a new one called ClickArt:Effects, which adds the powers of rotation, slant, distortion and perspective to MacPaint. Come by and see us in booth #237. I'll also be speaking on a panel about the future of the Mac in the business environment, also at the show on the 21st.

T/Maker Product Notes

We have now completely exhausted our stock of T/Maker III manuals, so the full T/Maker Integrated package is the only one available. Upgrade and second system pricing to T/MUG members is listed immediately following this column.

For IBM PC/compatible users with a hard disk, if you are really desperate for a way to move through the subdirectories while within T/Maker, we have a new version which does that. This requires a change of your entire T/Maker system disks for new ones. We will swap versions if you send in your original master disks and a check for $35. This is available for current T/Maker Integrated Software for PC DOS owners only.

Thanks, InfoWorld

T/Maker Integrated Software, the fourth in the line to be covered by InfoWorld, received a terrific reveiw in the December 17th issue. Although we blew our perfect track record of all excellents, the one "good" we received was for performance -- InfoWorld said that the "lack of a documented file-sharing capability caused us to downgrade the program's performance from Excellent to Good." They said they had terrible difficulty importing data files to and from other programs. This is odd, considering that as long as the user does not save TABS with files, the files are straight ASCII and should be readable by any other system. Well, this will be documented in the next manual more clearly! In any event, the review was extremely complimentary, and we are already receiving many orders as a result of it.

T/MUG PRICES FOR SOFTWARE:

The following are prices for various T/Maker options that we often receive requests for. Orders can be placed through us and we'll take them over the phone for COD, VISA, or MasterCard payments. California residents add 6.5% sales tax. Shipping is $6.50 for UPS blue and $3.13 for UPS ground. And remember, we MUST know what kind of computer and operating system you have! Our number is (415) 962-0195.

UPGRADE FROM T/MAKER III TO T/MAKER INTEGRATED SOFTWARE MUST HAVE PROOF OF T/MAKER III SERIAL NUMBER	$175.00
T/MAKER III FOR SECOND OPERATING SYSTEM, NO DOCUMENTATION MUST HAVE PROOF OF T/MAKER III SERIAL NUMBER	$137.50
T/MAKER INTEGRATED SOFTWARE FOR SECOND OPERATING SYSTEM NO DOCUMENTATION. MUST HAVE PROOF OF T/MAKER I.S. SERIAL NUMBER.	$225.00

T/MAKER T/IPS

Never Fear the End: Did you know that the command END can't hurt you? Try it. Type END at the WHAT NEXT? prompt. You will see that T/Maker accepts it but does nothing about it.

This is useful to know when creating DO lines. Some commands, such as TALLY, require the use of END to tell T/Maker when the command has ended. Others, like SORT, do not need the END. But, since the END never does any harm, you can use it whenever in doubt at no cost.

The same is true of the colon (:) placed after a files command. Both **FILES A** and **FILES A:** will get you the same thing.

Aligning Wedge Safety Tip: This one's in the manual . . . but who reads manuals anyway! If you are using two or more different aligning wedge sets in the same document, don't just copy the appropriate wedge over the paragraph(s) in question. Rather, give the wedges nametags, like this:

```
<<n                                                    ->>
These above are my normal wedges

        <<s                                  ->>
        These above are my small wedges
```

Now whenever you want either of these wedges, you just have to identify them like this:

```
<<=s
```

or

```
<<=n
```

Not only does this save you from mistakenly copying and dumping the wrong set of wedges, it also makes changing alignment a snap. By simply redefining the first wedges, all others with the same nametag will likewise be redefined.

T/Ips from Hal Bayman: *(By the time this issue comes out, Hal will probably own one of every T/Maker trinket in existence. Keep writing, Hal!)*

First, I just wanted to nudge other users to remember how handy it is to use T/Modify to designate the Default Data Drive and to default to TABS. Remember, both of these commands can be over-ridden while running.

I wanted to have a standard letter format to follow by INSERTing a Topbar file containing print commands, aligning wedges and such. However, the INSERT command does not import tabs. Therefore, I decided instead to leave a file on disk that is a letter template. My letter includes a tab setting where I want to type the date, an .indent print design command, aligning wedges, a salutation and closing, and my address. I just call it up and over-type the thing. Then I can RENAME it and SAVE it, or just print it. This is much easier than starting from scratch every time.

One last tip: I purchased my copy of T/Maker for the Kaypro in March 1984. I had for some time been using Plu*Perfect Systems BACKGROUNDER keyboard enhancement program, its predecessor, and Plu*Perfect's ZCPR based CP/M 2.2E. After using T/Maker a while, I decided to set up some dedicated keys using the aforementioned. To save printer ribbon, let me come to the point. Anyone trying this combination should be aware that they need to change the Console Input Delay Factor from 1 to 0 using T/Modify Option 13 -- System Patches.

IN SEARCH OF: APPLICATIONS

We know you're out there. Thousands of you. You send in your $15. You faithfully renew your subscriptions. Many of you even buy the back issues.

But what are you doing with T/Maker . . . and why don't you share it with us? By contributing your applications, you get not only the fame associated with being a T/Maker master, but you will also receive one of our infamous trinkets, like the T/Maker Fancy Pen or the Keychain.

We love any and all contributions, from simple to complex. And we like them even better if they come on disk, any format. We hope to be seeing you in print . . .

Using Dates with T/Maker

-- Robert Simon

This example will show an easy way to enter and work with dates. We will calculate the due date of an invoice and also sort records by date.

The first thing to do is create a file which contains today's date. We'll call this file TODAY.DAT. For convenience sake and for later uses, this file contains today's date in two forms. The first, called the "slash" form, shows the date in the familiar order -- month, day, year. The second form shows the date in a form that orders the date components in order of most significant component, that is, year, month, day. This way we can sort or order the date as if it were a number.

Here's the TODAY.DAT file:

```
TODAY.DAT

Todays date file:

          mm dd yy
          {sdate!}
sdate = "10/23/84"

          yy mm dd
          {cdate!}
cdate = "84,10,23"
```

This particular application is going to use the date file in conjunction with an order processing system which uses T/Maker's database features. After saving the file with today's date, get the order file and type UPDATE to enter data. When entering data, if you need today's date in the date field, simply type "x" or "X." The x will be transformed into today's date after giving the TRANSFER and SET commands.

Look at the RULES below. The TRANSFER command will bring in today's date and place it into a SET equation. A subsequent SET command will place today's date in the record file when it is marked with an X.

Notice also that if you enter 30 for the due date, the SET rules will add a month to today's date and use that as the due date. If you enter COD for the due date, today's date in used as the due date.

Finally, with the date in its numeric form, with the year, month, and date separated by commas, it can be sorted in numeric order. This is extremely useful for determining Accounts Receivable.

This is the order file, Before Any Transfer or Set command.

```
Enter SELECT IT WHEN (r = " ") END ORDER dd

=============================================================================

Instructions:

--GET Today, set today's date, SAVE

--UPDATE.....use X or x for today's date

---TRANSFER Today SET SAVE


<form>
Order Form
++++++++++++++++++++++++++++++++++++++++++++++++
Company:_________{company                          <}


Date          Total      Due Date    Received
-----------------------------------------
x                        cod
x                        30

{od    >}     ${t      >} {dd    >}  ${r      >}

<end>

<rules>
..                {cdate>}
od  99,99,99 = 84,10,23  when (od  = x or od  = X)
dd  99,99,99 = od when (dd = cod or dd = COD)
dd  99,99,99 = od + 100 when (dd = 30)
<end>
```

```
<record>
       a {company                              <}
+        {od    >} {t      >} {dd    >} {r      >}
<end>

_____________________________________________________________________
          Order    Billings     Due     Receipts
          Date                  Date
ex                 999,999.99           999,999.99
zv

..<here>
       a Byte Shop
+                x    310.00       cod
       a Computer Works of Florida
+                X    310.00        30
       a Computer Sales Hawaii
+                x    310.00       cod     310.00
       a Data Domain
+        84,05,01     310.00 84,05,01     310.00

=
=====================================================================
```

After commands:: TRANSFER TODAY SET

```
_____________________________________________________________________
          Order    Billings     Due     Receipts
          Date                  Date
ex                 999,999.99           999,999.99
zv

       a Byte Shop
+        84,10,23     310.00 84,10,23
       a Computer Works of Florida
+        84,10,23     310.00 84,11,23
       a Computer Sales Hawaii
+        84,10,23     310.00 84,10,23     310.00
       a Data Domain
+        84,05,01     310.00 84,05,01     310.00

=
_____________________________________________________________________
```

After commands:: ORDER dd COMPUTE

```
          Order    Billings     Due     Receipts
          Date                  Date
ex                 999,999.99           999,999.99
zv


..<here>
        a Data Domain
+         84,05,01     310.00 84,05,01     310.00
        a Byte Shop
+         84,10,23     310.00 84,10,23
        a Computer Sales Hawaii
+         84,10,23     310.00 84,10,23     310.00
        a Computer Works of Florida
+         84,10,23     310.00 84,11,23


=        Total       1,240.00             620.00
```

Order Entry with Inventory Count

-- Peter Roizen

Below is a simple order entry system. To keep things easy, we have not used separate <FORM> and <RECORD> definitions, but decorated our <RECORD> somewhat. This is not a good practice for real applications but helpful for keeping examples short.

Let us suppose that after entering our orders for the day in this file, we would like to update an inventory count of what parts are still available.

ORDERS FILE

```
<rules>
a1 99,999.99 ' '  = q1 * p1
a2 99,999.99 ' '  = q2 * p2
a3 99,999.99 ' '  = q3 * p3
total 99,999.99 = a1 + a2 + a3
<end>
<record>
-------------------------------------------------------------------
Name:     {name                          }
-------------------------------------------------------------------
          Quantity   Part       Name            Price       Amount
-------------------------------------------------------------------
+         {>   q1}   {pr1 <   } {n1        }  {>   p1}   {      a1}
+         {>   q2}   {pr2 <   } {n2        }  {>   p2}   {      a2}
+         {>   q3}   {pr3 <   } {n3        }  {>   p3}   {      a3}
-------------------------------------------------------------------
          Total:                                          {total  }
-------------------------------------------------------------------
<end>
```

Our inventory file might look like the one below. Notice that the rules are set up so that using the SET command on the file will recalculate the "on-hand" figure based on the number of "sales" and the number "received." These rules also clear the "sales" and "received" fields.

Thus, solving our problem amounts to getting the sales figures from our file of orders into the inventory file.

INVENTORY FILE

```
<rules>
price 99,999.99 = price
onhand 999,999 = onhand - sales + received
sales = ' '
received = ' '
<end>
<record>
    Part: {part}
    Name: {name                          }
   Price: {price  >}
 On Hand: {onhand  }
Received: {received} }}
   Sales: {sales   } }}
<end>
    Part: 1
    Name: HAMMER
   Price:      14.95
 On Hand:         11
    Part: 2
    Name: PLIERS
   Price:       5.95
 On Hand:          9
```

Consider the file below. It represents another way of looking at at our file of orders. It defines only a part of each order; namely, the quantity sold and part number for the lines that contain the items sold. The last line in the file means that if a selection is made from this file, one will actually go through the orders file.

PARTS FILE

```
<record>
+         {>sales}   {part<    }
<end>
<continue> orders
```

If the file above were called PARTS, the command "SELECT PARTS END GROUP part END" would prepare a file of total sales by part number. This file could then be saved on disk under some new name, say, "TEMP."

Then, GETting the inventory file, we could use the command "COMPLETE TEMP part." This would add the sales figure to our inventory record based on a corresponding part number. Then only the SET command would need to be issued to update the on-hand inventory figures.

One More Approach to Side-by-Side Addresses

(Ed Note: You users have been tremendously creative in your many ways to place addresses side by side. Here's another approach, by Dollie Latimer. Any more?)

Here's my contribution to the side-by-side problem. First, I type the entire list of names in one long column.

Since I put it on legal sized paper, I put ".length 80" at the beginning of the page. Then I print it to the screen to see which name it breaks on. In the logical breaking place, I enter ".new " followed by ".indent 50."

When the print command stops for the new page break, I take the paper that was just printed on out of the printer, and reinsert it. the program does the rest. Isn't that simple?

(Ed Note: Yes, it is simple! Note that it can also work for lists longer than can fit on one page. Follow the same instructions, putting the ".new" and ".indent 50" commands on every other page break.)

T/Maker Accounting Systems Available

Many customers, OEMs and distributors have emphasized to us the importance of integrated accounting applications. Even a seemingly simple company -- for example, a single person selling a single product on a part-time mail order basis from the home -- could benefit from accurate cash records, billing statements, and statements of changes in financial position.

For the larger company, precise financial records can save thousands of dollars during tax and audit time. Furthermore, these records can be a tremendous aid in budgeting and planning.

We at T/Maker like to believe that T/Maker can do pretty much anything we'd want to do with a computer. It seems that two of our users/consultants have tried to prove us right. Both have designed accounting systems which use T/Maker as their base. An outline of these systems follows.

In many of your companies, you may already be using T/Maker for checkbook, billing, budgeting, and other accounting-related applications. The beauty of moving from your own systems into one of these, is that the interrelationships and reporting formats have been worked out by professionals, yet the data formats are compatible (with a little manipulation) with the data you have already collected and entered.

Though we are not endorsing or marketing these applications ourselves at this time, we feel strongly that our users could benefit from the knowledge of these systems. For pricing and other information, please contact the author of the applications. They are listed here in increasing order of sophistication and complexity.

English Only Computer Systems

English Only Computer Systems calls their system "accounting overlays for the un-accountant". The system is totally menu driven and consists of one data entry register for all checks, deposits, sales information and adjustments for a double entry accounting system.

At present, reports available are check register with running balance, sales register, detailed general ledger, Profit/Loss statement showing current period and year-to-date, and balance sheet. Available shortly will be modules for payroll, check writing, invoicing, inventory and accounts receivable reports.

for pricing and more information, contact:

Gus Korman
English Only Computer Systems
5222-1 Lindley Avenue
Encino, Ca 91316
(818) 344-2422

Strategic Business Systems, Inc.

Strategic Business Systems, Inc. has developed two systems which are of different levels of complexity. Both systems are menu-driven (including extra help screens) and can be tailored to a business' particular reporting needs.

These systems provide full accounting statements, including: statement of changes in financial position, income statement, balance sheet, cash flow , general ledgers (opening, closing) balance sheet posting detail, income statement posting detail, account analysis, accounts receivable.

The smaller system offers somewhat less flexibility than the larger, but can be run on many more, smaller computers and is simpler to install, requiring little knowledge in accounting.

The larger system is the most sophisticated application of T/Maker we have probably eve seen. It encompasses all of the smaller system, plus allows for completely personalized statements, multiple cash and sales registers, and different levels of summarization. It requires larger disk capacity, preferably a 5-MB or greater hard disk.

Companies may submit their Chart of Accounts and Opening Balances to SBS, who will then tailor the system to their needs.

For pricing and more information, contact:

R.M. Danielson
Strategic Business Systems, Inc.
P.O. Box 137
Star Prairie, Wisconsin 54026
(715) 248-3289
(715) 248-3434

Bibliographies with T/Maker's Database

-- *Elyse Sommer*

Whether you need a bibliography for your PhD dissertation or simply want to organize your library, why not use the data base to do it? You'll be able to alphabetize your entries with ORDER, create special category files with SELECT and format by inserting appropriate print commands.

The following record contains fields for the basic information, as well as for descriptive comments. The double brackets after all but the first Comment field tell T/Maker to use the space for storage only if you enter information. In short, you can give yourself enough room without wasting valuable bytes.

Here's the record definition:

```
<record>
Title:  {title                                                  }
Auth:   {auth                                        }
Pub:    {pub                                         }
Date:   {date       }
Cat:    {cat                                                    }
Comm:   {comm                                                     }
Comm2:  {comm2                                                  } }}
Comm3:  {comm3                                                  } }}
Comm4:  {comm4                                                  } }}
Comm5:  {comm5                                                  } }}
Comm6:  {comm6                                                  } }}
Comm7:  {comm7                                                  } }}
<end>
```

Here's how your information would be contained in the file once you quit the update mode: (of course with the record definition still at the top)

```
Title:  Perfect Writer Made Perfectly Clear.
Auth:   Sommer, Elyse.
Pub:    Radnor, PA:  Chilton Books.
Date:   1984
Cat:    computer/word processing.
Comm:   Step-by-step guide for popular word processing program.
```

```
Comm2:  Special features include learn by doing exercises,
Comm3:  reference charts to fit most keyboards, extensive
Comm4:  hints and input/output examples.  Quick reference
Comm5:  appendix section.  Index.
Title:  The Second Self.
Auth:   Turkle, Sherry
Pub:    New York:  Simon & Schuster.
Date:   1984
Cat:    Computer/sociology
Comm:   An M.I.T. professor's look at the new computer
Comm2:  culture from humanist viewpoint.  Six year study
Comm3:  encompasses children and computers, "hackers" and
Comm4:  professinal programmers, artificial intelligence.
Comm5:  Index.
```

Naturally, the above is not the format in which you'd want to print out your bibliography. And it needn't be. Simply create a new file -- say, biblio.frm. Enter a new record format. You don't have to retype it. Instead, CREATE the new file and INSERT the original file, then FIND <end>, CLIP BEFORE EDIT. Next, set up a macro to erase the headings with the colons and push over the bracketed field entry with the command to close up space. Notice the use of T/Maker's printing and word processing commands, such as ".ind", the aligning wedges, and the backslash (\) used to underline the title. Here's the resulting record format we'll use:

```
<record>
        <<<                                                    ->>
\
.
{title                                                      !}\
{auth                                          }
{pub                                           }
{date         }
{cat                                                        }
.ind 3
{comm                                                          }
{comm2                                                     } }}
{comm3                                                     } }}
{comm4                                                     } }}
{comm5                                                     } }}
{comm6                                                     } }}
{comm7                                                     } }}
.ind 0

<end>
..<here>
```

To transfer the data from its original file, go to the WHAT NEXT? prompt and SELECT biblio (or whatever your original file was named) END. Then give the ALIGN command to produce your finished report. You'll then see why a period was put at the end of each initial entry. Here is a printout of the results:

Perfect Writer Made Perfectly Clear. Sommer, Elyse. Radnor, PA: Chilton Books. 1984 computer/word processing.

Step-by-step guide for popular word processing program. Special features include learn by doing exercises, reference charts to fit most keyboards, extensive hints and input/output examples. Quick reference appendix section. Index.

The Second Self. Turkle, Sherry New York: Simon & Schuster. 1984 Computer/sociology

An M.I.T. professor's look at the new computer culture from humanist viewpoint. Six year study encompasses children and computers, "hackers" and professinal programmers, artificial intelligence. Index.

USING 'SET' AND 'SQUEEZED' DATA

-- Bert Zitek

My faith has been shattered -- I have found a problem with T/Maker.

Using SELECT I transferred some data to a printing file and used the SET command with RULES to convert numerical data into 'strings'. I also wanted to 'squeeze' the first name with '!' so that the last name would follow with just one space instead of a wide gap. Here's the printing record definition with the data right after SELECT:

```
---------------------------------------------------------------------
<record>
1     {fname!       } {lname<            }
2     {bowl                              }
3     {address<                          }
4     {city<             }   {#st}  {zip}
5     {a#rea}-{p}-{nu}
.new
<end>
1     Robert          Witherspoon
2     Arthur's Bowl
3     152 North State Street
4     Concord                 MA  02195
5     202-555-1324
.new
1     Jacques         Cousteau
2     Underwater Inc
3     887 Central Ave.
4     Dover                   NJ  07670
5     201-384-3345
.new
---------------------------------------------------------------------
```

The names were fine after SELECT but after I issued the SET command, I found that the space that corresponded with the space between the first and last name in the record had eliminated a letter in some of the last names, like the samples here:

The file, after the SET command was invoked:

```
<rules>
p = "   " when p = 415

<end>
<record>
1     {fname!       } {lname<            }
2     {bowl                              }
3     {address<                          }
4     {city<             }   {#st}  {zip}
5     {a#rea}-{p}-{nu}
.new
<end>
1     Robert Withers oon
2     Arthur's Bowl
3     152 North State Street
4     Concord               MA  02195
5     202-555-1324
.new
1     Jacques Couste u
2     Underwater Inc
3     887 Central Ave.
4     Dover                 NJ  07670
5     201-384-3345
.new
```

I could only assume that the SET command was 'squeezing' the first name a second time -- which left the space.

I first tried to correct the problem by bringing the data into the print file using the field name 'fname' and then REPLACE it with 'fname!'. I found that when replacing field names they must be the same number of characters. Then I tried using a first field name of 'fname<' and then brought the data into the print file with SELECT followed by REPLACE 'fname<' with 'fname!' and then used the SET command. It worked.

That's what I like most about T/Maker -- There always seems to be a solution to the few problems that do rarely occur.

VISICALC to T/MAKER: The Transition Continues

-- *Royal Farros*

Last issue, we discussed ways to make T/Maker look more like a traditional spreadsheet. We said it was perfectly legal to label columns with LETTERS rather than 999's. We also suggested typing out the full word EXAMPLE on the Example Line, thus giving you a "visual ruler" under which column equations can be placed. As a final tip, we suggested ending all spreadsheets with a "blank example line" (just EXAMPLE with no column headings on it).

We ended the article discussing JC Row Calculations. We said that it's very important to understand how T/Maker works on each row before we can confidently build large spreadsheets made up of many rows.

The JC row code will help us master two very important spreadsheet techniques: moving values around a table (STORE & FETCH) and calculating a result based on a future result (RE-CALCULATION).

To understand these and other spreadsheet features, let's first talk about the order in which T/Maker calculates tables.

ORDER OF CALCULATION

T/Maker works through a spreadsheet from the top one row at a time, from left to right -- but in an odd sort of way. COLUMN EQUATIONS are dealt with first, followed by the ROW EQUATIONS.

To give you a better picture of this, imagine T/Maker as having a separate CALCULATOR for each column in a table.

The order of calculation is as follows:

1) Each calculator looks for the first row that has a MATH SIGN on it in the first seven columns.

2) T/Maker calculates any applicable ROW EQUATIONS for this row in the order determined by their ROW CODE NUMBERS.

3) T/Maker then goes to the next row with a MATH SIGN (in the first seven columns) and calculates the column values if the MATH SIGN is an EQUAL SIGN. T/Maker then goes back to Step 2.

This loop continues until T/Maker can't find any more rows with MATH SIGNS or until it finds a BLANK EXAMPLE LINE (one which contains no Model Numbers).

STORING & FETCHING VALUES

T/Maker allows you to STORE a value in one part of a spreadsheet and FETCH it back in another part. Using a JC row equation to STORE & FETCH makes this useful activity easy.

You can STORE up to 26 unique values, one for every letter in the alphabet. Think of T/Maker as having 26 mailboxes or mail "slots". In front of each slot is a letter, denoting the "name" of that particular storage slot. Storage slots start out having a "0" value. We can ask T/Maker to STORE a table value in a particular slot, and FETCH that value from that slot. We'll use the JC row code to accomplish this.

Suppose we need to move a value from the AAA Column to the CCC Column, and a value from the BBB Column to the DDD Column. The table below shows an example of this. When you ask T/Maker to COMPUTE this table, this is the result.

```
-------------------------------------------------------------------------
Example    AAA    BBB     CCC     DDD
ZV

jcl        stA    stB                          <---  STORE a value 10 in slot A
                                                     STORE a value  5 in slot B
+           10      5

jcl                       ftB     ftA          <---  FETCH a value  5 from slot B
                                                     FETCH a value 10 from slot A
+                          5      10

Example
-------------------------------------------------------------------------
```

Suppose that we want to take the value in column AAA and the value in column BBB, add them together, and place them at the bottom of column CCC. After COMPUTE, our table would look like this.

```
-------------------------------------------------------------------------
Example    AAA    BBB     CCCCCCCCCCCCC
ZV

jcl        stA    stB                          <---  STORE a value 10 in slot A
                                                     STORE a value  5 in slot B
+           10      5

jcl                       ftA + ftB + =        <---  FETCH a value 10 from slot A
                                                     ENTER value 10 into CALCULATOR
+                                    15               FETCH a value  5 from slot B
                                                     ENTER value  5 into CALCULATOR
                                                     TOTAL the values in CALCULATOR
Example
```

By the way, that "ZV" under EXAMPLE starts off the ZERO VALUE Line. The Zero Value Line allows us to specify how we want a 0 represented in our tables. We've asked T/Maker to display "blanks" for every 0 value in our tables above. Below are some more examples of the ZV Line.

Example	aaaaa	bbb	ccccc	ddd
ZV	---		zip	0
+	5	5	5	5
+	-5	-5	-5	-5
=	---		zip	0
Example				

RECALCULATION WITH T/MAKER (Passing Values from Bottom to Top)

You can STORE values calculated at the bottom of a column and FETCH them for use at the top of another column by "RE-COMPUTING".

Typing a "COMPUTE C" after COMPUTE asks T/Maker to make an additional pass at the current spreadsheet table. Typing "COMPUTE C C" asks T/Maker to make two additional passes at the current table. Typing three C's after COMPUTE asks T/Maker to re-compute the table three times, and so on.

Suppose we asked T/Maker to "COMPUTE C" on the following table. In super slo-mo, here's what happened.

FIRST PASS OF COMPUTE:

```
-------------------------------------------------------------------------
Example     AAA     BBB

jcl                 fta          <---  FETCH value in slot A (currently 0)

+             5       0          <---  ENTER value  5 into AAA CALCULATOR
                                       ENTER value  0 into BBB CALCULATOR
*            10      10          <---  MULTIPLY value in AAA CALC. by 10
                                       MULTIPLY value in BBB CALC. by 10
jcl         sta                  <---  STORE RESULT of AAA CALC  (50)
                                                                 in slot A
=            50       0

Example
-------------------------------------------------------------------------
```

SECOND PASS OF COMPUTE:

```
-------------------------------------------------------------------------
Example     AAA     BBB

jcl                 fta          <---  FETCH value in slot A (currently 50)

+             5      50          <---  ENTER value  5 into AAA CALCULATOR
                                       ENTER value 50 into BBB CALCULATOR
*            10      10          <---  MULTIPLY value in AAA CALC. by 10
                                       MULTIPLY value in BBB CALC. by 10
jcl         sta                  <---  STORE RESULT of AAA CALC (50)

=            50     500

Example
-------------------------------------------------------------------------
```

As you can see, the JC row code helps us take a controlled look at what T/Maker is doing with our table data.

Next time, we'll get into T/Maker spreadsheet "short cuts".

T/Maker Consultants

To be registered as a consultant and listed in T/MUG, send us a copy of your system complete with explanation and documentation. If it is consistent in quality and degree of completeness with the systems we have previously accepted, we will register you as a consultant.

Dick Danielson, President
Strategic Advisory Consultants, Inc.
P.O. Box 137
Star Prairie, WI 54026
(715) 248-3434

Robert Payne, President
Business Computer Service, Inc.
1210 Smith Street
Charleston, WV 25301
(304) 343-9471

James L. Bullock
Bullock's Electronics
1016 West Street
Norton, KS 67654
(912) 877-3277

Ron Roizen
2011 Francisco Street
Berkeley, CA 94709
(415) 549-0103

Emil Widmer
Computertechnik fur Manager AG
Baarerstrasse 45, Postfach 708
CH-6301 Zug, Switzerland
Tel. 042 21 0887

Doug Beynon
Beynon Howes & Company
P.O. Box 944
Lawrence, KS 66044

James D. Gilbert
Gilbert's Data-Word Processing
237 Luff Lane
Warminster, PA 18974
(215) 443-9963

Robert D. Williamson, President
Can-American Electronics Corp.
7220 Taft Street
Hollywood, FL 33024
(305) 966-5588

Thaitronic Company Ltd.
P.O. Box 11-90 Phrakhanong
Bangkok, Thailand
Tel. 392-8307
TELEX Unitech Erawan TH 82189

Peter Channon, Frank McInnes
The Computer Factory
214 Harbord Road
Brookvale, NSW Australia
(02) 938-2522

D. Michaut
Departement Organisation
Societe Generale Des Cooperatives
de Consommation
27, 33 quai A. le Gallo
92517 Boulogne-Billancourt Cedex
Paris, FRANCE

Don Baack
6495 S.W. Burlingame
Portland, OR 97201
(503) 244-2741

COMPUTING CORRELATIONS ON DATA LODGED IN DATABASE FILES

-- Ron Roizen

This article provides an easy method for computing Pearson's product moment correlation coefficients on data lodged in database files.

Suppose, for the sake of example, that you have a database file named FACTS. FACTS holds information on four variables named v.1, v.2, v.3, and v.4. The record definition in this database file looks like this:

FILE: FACTS

```
---------------------------------------------------------------
<record>
Variable.1:  {v.1>   }
Variable.2:  {v.2>   }
Variable.3:  {v.3>   }
Variable.4:  {v.4>   }
<end>
---------------------------------------------------------------
```

Further suppose that FACTS holds data on seven cases. Finally, suppose that you wanted to determine the correlation between Variable 1 and Variable 2.

First, construct a file exactly like the one shown on the next page. Name it CORR:

Take a close look at the CORR file. It has three parts.

(1) The top line is a DO Command Line. This line--when you use the DO command--will select data from the FACTS file, compute the correlation statistic, and, finally, relocate the screen so that the correlation result appears at its upper left corner.

(2) Next, there is a record definition. Notice that the fields in this record definition have no field names in them.

(3) Finally, there is a table for calculating the correlation coefficient.

-- Correlations, Continued

FILE: CORR

```
select FACTS 7 compute c find COR /97

<record>
+ +     {       } {       }
<end>

               A          B      C      D       E       F      G       H       I       J                    K
example  xxx,xxx.xx  xxx,xxx.xx xxx.xx xxx.xx xxxx.xx xxxx.x xxxx.xx xxxxx.xx xxxxx.xx xxxx.xx              xx.xx
zv
uc1                             fta    ftb
uc2            +                -              =
uc3                       +              -              =
uc4                                             +       *      =
uc5                                            +*                     =
uc6                                                    +*                     =
       <here>

  =
jc7          sta          stb                                                   +       *      sqr
jc8                                                                  +                      /                    =
avr                                                                                         0.00     CORRELATION:
```

How is the file used? In steps:

(1) GET the FACTS file, making it the working file.

(2) Enter the editor and fill in the variable names whose correlation you want--in this case, v.1 and v.2--into the blank fields.

In other words, CORR's record definition will now look like this:

```
<record> + + { v.1 } { v.2 } <end>
```

(3) Home the cursor, quit the editor, type DO, and press RETURN.

-- *Correlations, Continued*

How does CORR work? Notice that there is a "<here>" line in the table just below the uc6 Calculation Control Line. This "<here>" line tells T/Maker to deposit the data selected from FACTS just below it. Once these data have been filled in, T/Maker's COMPUTE command carries out the correlation calculation.

The completed calculation might leave the file looking like the one below:

FILE: CORR, after the DO Command Line has been executed

```
<record>
+ +      {v.1     } {v.2     }
<end>
                A          B      C      D       E       F      G       H       I       J                      K
example  xxx,xxx.xx xxx,xxx.xx xxx.xx xxx.xx xxxx.xx xxxx.x xxxx.xx xxxxx.xx xxxxx.xx xxxx.xx                 xx.xx
zv
uc1                             fta    ftb
uc2             +               -              =
uc3                      +              -              =
uc4                                             +       *      =
uc5                                             +*                     =
uc6                                                     +*                     =
       <here>
+ +           12.00      14.00  41.29  40.14  -29.29  -26.1  765.61   857.65  683.45
+ +           33.00      44.00  41.29  40.14   -8.29    3.9  -31.96    68.65   14.88
+ +           78.00     123.00  41.29  40.14   36.71   82.9 3042.04  1347.94 6865.31
+ +           33.00      12.00  41.29  40.14   -8.29  -28.1  233.18    68.65  792.02
+ +           44.00      55.00  41.29  40.14    2.71   14.9   40.33     7.37  220.73
+ +           77.00      21.00  41.29  40.14   35.71  -19.1 -683.67  1275.51  366.45
+ +           12.00      12.00  41.29  40.14  -29.29  -28.1  824.18   857.65  792.02

  =          289.00     281.00 289.00 281.00    0.00    0.0 4189.71  4483.43 9734.86
jc7           sta        stb                                          +       *      sqr
jc8                                                          +                      /                      =
avr           41.29      40.14  41.29  40.14    0.00    0.0  598.53   640.49 1390.69  943.78    CORRELATION:  0.63
```

Of course, the CORR file will calculate the correlation coefficient for any pair of variables. All you need do to adapt it to your own database file is (1) supply your database's filename to CORR's DO Command Line file and (2) supply the field names of the variables whose correlation you want to calculate inside the CORR file's blank field braces.

In short, an exact copy of CORR will provide you with a ready-to-use template for carrying out correlation computations on data drawn from database files.

T/Mug Back Issues

If you found this issue useful, you might want to consider picking up the back issues. T/Mug has been published since 1982, each issue aiming for a mix of articles for the novice T/Maker user through the expert.

The T/Mug also serves as a forum for users of particular machines and in particular industries. We try to include the applications most requested in letters and phone calls. Past issues have included: invoices, checking account management, statistics, personalized form letters, transferring data files, inventory, tracking athletic events, time calculations, and many more. The applications, like the product, have evolved over the years, so you may find the more recent issues to be of more value.

To order, please fill out this form and enclose it with your check (drawn in US dollars) and mail it to the address below. Customers outside continental US please add $4 per box checked for shipping.

Back Issues -- Please send me:

[] Back issues: 1984 - present (six + issues) $15.00

[] Back issues: 1983 (five issues) $10.00

[] Back issues: 1982 (five issues) $10.00

T/Mug Subscriptions:

You might also want to subscribe (or renew your subscription) to the T/MUG newsletter. This entitles you to six bimonthly issues like the one in your hands. To order, please fill out this form and enclose it with your check (drawn in US dollars) and mail it to the address below. Customers outside continental US please add $20 for first class mailing charges.

Subscription -- please:

[] Enter my subscription for one year $15.00

[] Renew my subscription $15.00 (my T/Mug mailing label is enclosed)

Name:_____

Company:_____

Street:_____

City:_____ State:_____ Zip:_____

T/Maker Serial Number(s):_____ Machine type:_____

Please make check out to T/Maker Users' Group and Mail to:

T/Maker Users' Group * 2115 Landings Drive * Mt. View, CA 94043

T|MUG
T/MAKER USER'S GROUP
T/MAKER COMPANY
2115 LANDINGS DRIVE
MOUNTAIN VIEW, CA 94043

BULK RATE
U.S. POSTAGE
PAID
PALO ALTO, CA
PERMIT NO. 96