We Speak Your Language
ENGLISH®

You can use a "translator"... or you can communicate with a Reality® computer system in ENGLISH.®

ENGLISH® retrieval language isn't a computer programmer's language, with complicated codes and symbols that are foreign to everyone but the "programmer." It's a management language for people who need fast access to information and who don't want to go through a "translator." It's an integral part of the Reality® family of business computer systems and essentially gives whoever needs information direct access to that information stored in the data base. Reality computers can also use the language of your business, since you can insert the jargon of your particular business right into the dictionaries used by ENGLISH retrieval language. A typical ENGLISH retrieval language inquiry into the data base would be a free-form sentence consisting of verbs, adjectives, adverbs and nouns that direct the computer to select the items you want out of data files and deliver them to you in readily understandable and usable form.

Frees up programmer
It makes retrieving data so simple that anyone from the company president on down can use the system effectively. Should you have a programmer on your staff, ENGLISH retrieval language will free this valuable person for the more creative aspects of programming your Reality business computer. What's more, direct access to the computer eliminates the danger of error in relaying information between programmer and user.
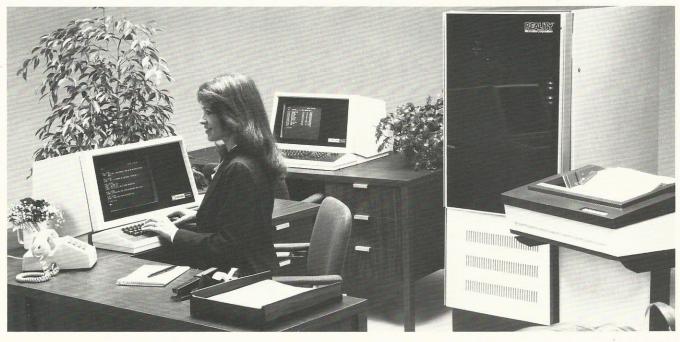
Simple sentence
Here's how simple it is to select data from a file: tell the computer what to do by using a VERB (list, sort, count, select, etc.). Name the files, data attributes (fields), etc., by using a NOUN (inventory, purchase orders, accounts payable, etc.). Use connectives to combine grammatical phrases, alter the report format and modify the action of the verb. Data selection criteria can specify a specific item or record, an entire file or a set of conditions that will identify the kind of items you want the computer to include.

It's in the Dictionary
ENGLISH retrieval language is a dictionary-driven language. The vocabulary used to compose ENGLISH retrieval language sentences is contained in dictionaries. Verbs, file names and connectives are located in each user's master dictionary. Every file consists of a data section and a dictionary which contains a structural definition of the data. ENGLISH retrieval language references the dictionary for data attribute (field) descriptions. These descriptions specify data fields, functional calculations, inter-file retrieval operations and display format and positioning.

```
:SORT THE INVENTORY FILE WITH  VALUE > "100"
AND < "250"

PAGE 1                              09:39:41  17 JUL 1981

INV . . . . .  DESC . . . . . .  QTY . . .  COST . .  VALUE . . .
11-1946       RESISTOR          3000       .04       $120.00
13-7401       IC                 250       .89       $222.50
33-0100       SOCKET             430       .25       $107.50
END OF LIST
```

This SORT statement uses conditional selection criteria (WITH VALUE > "100" AND < "250").   Items in the INVENTORY file that have a value greater than $100.00 and less than $250.00 will be displayed in sorted order. When no sort keys are specified, the item-id (identification code for individual data items), in this case, the part number, will be used as the sort key.

```
:SORT THE INVENTORY FILE BY DESCRIPTION
BREAK-ON DESCRIPTION TOTAL THE QUANTITY COST TOTAL
THE VALUE

PAGE 1                              09:41:15  17 JUL 1981

INV . . . . .  DESC . . . . . .  QTY . . .  COST . .  VALUE . . .
10-5003       CAPACITOR          89        .33       $ 29.37
10-8911       CAPACITOR          73        .27       $ 19.71
              ***               162                  $ 49.08

11-1030       RESISTOR          864        .03       $ 25.92
11-1946       RESISTOR         3000        .04       $120.00
              ***              3864                  $145.92

***                            4026                  $195.00
END OF LIST
```

This SORT statement uses the BREAK-ON/TOTAL feature to summarize (TOTAL) specified attributes. The totals will be displayed every time the value of the BREAK-ON attribute (in this case, DESCRIPTION) changes. A sort key (BY DESCRIPTION) has also been specified.

```
:SORT THE DICTIONARY OF THE INVENTORY FILE
BY CODE BY AMC

PAGE 1                              09:39:41  17 JUL 1981

INV . . .  CODE.  AMC .  S/NAME. .  CONV. .  CORR. .  T. . .  MAX
DESC       A      1                                   T       11
COST       A      2                 MD2               R        5
QTY        A      3                                   R        4
C          A      4                                   L        1
DUE        A      5                                   R        4
DATE       A      6                 D                 L       11
VALUE      A     99                 MD2$    F:2;3;*   R        7
1          S      1     DESC                          T       11
3          S      2     COST        MD2               R        5
2          S      3     QTY                           R        4
5          S      4     CODE                          L        1
4          S     99     VALUE       MD2$    F:2;3;    R        7
```

This SORT statement displays a sorted listing of the dictionary for the INVENTORY file. Two sort keys are specified (BY CODE and BY AMC). The dictionary contains attributes (CODE = A) and attribute synonyms (CODE = S). The AMC specifies the field position of that attribute in the data record. The synonyms "1,2,3,4,5" form the implicit attribute synonym list, that is, the fields that print automatically if none are requested. CONV is the conversion specification (D for date, MD2 for masked decimal with 2 fractional digits). CORR is the correlatives specification (F:2;3;* is the function of multiplying field 2 times field 3). The type (T) specifies display of data either left (L), right (R) or text (T) justified in MAX columns.

```
:STAT INVENTORY VALUE
STATISTICS OF VALUE:
TOTAL = 10349.03, AVERAGE = 29.8243, COUNT = 347

:COUNT INV WITH QTY > "500"

151 ITEMS COUNTED.

:SORT INV DBL-SPC LPTR

:CREATE-FILE (PUR-ORDERS 3,1  23,1)
```

STAT is used to sum a specified attribute (VALUE). The display shows the accumulated total, the count of the number of items that met the selection criteria and the average value. COUNT is used to determine the number of items in a file which meet the selection criteria. On the SORT INV statement, DBL-SPC will cause double spacing, and LPTR will direct the sorted listing to the line printer. CREATE-FILE causes a dictionary and data file to be allocated under the name PUR-ORDERS. The parameters (3,1 23,1) are used to optimize the organization of the file structure.

```
:LIST ORDER#  WITH DATE BEFORE "7/1/80"

PAGE 1                          09:52:37  17 JUL 1981

ORDER#. .  VENDOR. .  ITEM . .  PART#. . .  QTY. . .  DUE DATE

012433     000103     1         11-1031     1000      01/15/81
                                             1000      03/15/81
                                             1000      05/15/81
                      2         13-1139      50        14/15/81
                      3         31-8764      100       01/15/81
                                             100       02/01/81
                                             100       02/15/81
022230     142124     1         10-6500      75        14/30/81
012997     020772     1         26-1200      30        03/01/81
                      2         26-0201      30        03/01/81
```

```
:CHANGE INV 11-3066 DESC TO "IC"
11-3066 UPDATED.

:NEW-PO
...ENTER NEW PURCHASE ORDER...
PO#                    :020316
VENDOR#                :112770
ITEM#. . . .  PART#. . .  QTY. . . . .  DUE DATE
1             11-1032     50            01/15/81
              —           100           02/15/81
2             20-5555     30            02/01/81
END
'020316' ADDED
```

Attributes ITEM and PART# may contain multiple values which are displayed in a columnar format. Also, the attributes QTY and DUE DATE show a further level of indenturing due to multiple sub-values for a given value. ENGLISH retrieval language will automatically display values and sub-values in an indentured format while maintaining their corresponding relationships.

The PROC high-level procedural programming language can be used to write ENGLISH-like verbs or to create an interactive data entry routine. The CHANGE PROC accepts all required information in a single statement. The NEW-PO PROC prompts the user for required information and sets up column headings which can then be filled in by the user.

In plain English, here are some other reasons Reality is for you.

Plain ENGLISH retrieval language is, of course, a compelling reason to use a Microdata Reality computer system. But it's not the only one. Reality was designed with the businessman in mind. So it solves business problems. And it grows as a business grows. It gives you what you need now, tomorrow and the day after.

No wonder businessmen users rated Reality above any other business computer system.

Reality from Microdata "The Businessman's Choice."™

Microdata
"The Businessman's Choice."™

Here's how easy it is to use ENGLISH

Files are organized in a hierarchical structure. This means that a file at one level points to several files at a lower level. There are four file levels: (1) system dictionary, (2) user master dictionary, (3) dictionary file and (4) data file.

1. The *system dictionary file* contains all legal user log-on names, passwords and security codes.

2. The user *master dictionary file* contains all user vocabulary, all accessible file names, application PROCs (procedural programs) and attributes describing the structure of the information in the dictionary files.

3. A *dictionary file* contains attribute and attribute synonym definitions describing the structure of the data in the user-file. These items define the data field names, describe how it is to be accessed and displayed and any other functions or interrelations with other files or data records.

4. The *data file* contains the actual data, stored in a variable field length format. In addition to the normal record/field data structure, a field can contain multiple values — and a value can consist of multiple sub-values. This all adds up to an extremely flexible file structure.

One thing leads to another
What this means, very simply put, is that with the Reality computer system and ENGLISH retrieval language, the user can obtain the desired data in the desired format quickly and easily. The user is not exposed to any of the intricacies of a computer operation because he uses familiar words and phrases. A series of "dictionaries" makes certain that he uses the proper vocabulary for each inquiry.

Compare this to working with another computer system using complicated codes and symbols in place of ENGLISH retrieval language and you'll choose a Reality computer system.

Examples of CRT screens using ENGLISH

```
:LIST THE INVENTORY FILE

PAGE 1                          09:31:05  17 JUL 1981

INV . . . . .  DESC . . . . . .  QTY . . .  COST . .  VALUE . . .  CODE
11-1030        RESISTOR         864          .03      $ 25.92     D
10-8911        CAPACITOR         73          .27      $ 19.71     B
32-5421        BRACKET           19         1.57      $ 29.83
11-1946        RESISTOR        3000          .04      $120.00     B
20-0017        PC BOARD          19        22.50      $427.40
10-5003        CAPACITOR         89          .33      $ 29.37     D
13-7401        IC               250          .89      $222.50     A
33-0100        SOCKET           430          .25      $107.50     A
11-9503        RESISTOR         130          .40      $ 52.00
               MODULE
10-4444        CAPACITOR        133          .65      $ 86.45     A
```

When the system prompts the display with a colon (:), the user may enter an ENGLISH retrieval language statement. LIST THE INVENTORY FILE will cause the system to list all items in the file called "INVENTORY." Column headings are determined by a list in the file dictionary.

```
:LIST THE INVENTORY FILE WITH PART-NUMBER
'11-1946' DESCRIPTION QUANTITY COST
VALUE CODE LOCATION

PAGE 1                          09:35:27  17 JUL 1981

INV    11-1946
DESC   RESISTOR
QTY    3000
COST   .04
VALUE  $120.00
CODE   B
LOCATION    BIN 13-C

1 ITEM LISTED.
```

The LIST statement at the top of the screen explicitly names the attributes to be displayed (DESCRIPTION...LOCATION) from the INVENTORY file. A particular item selection (11-1946) has also been specified. If the attributes cannot fit across the page, ENGLISH retrieval language will automatically revert to non-columnar display, as seen here.

Microdata

Post Office Box 19501, Irvine, CA 92713.
Telephone: 714/540-6730, TWX: 910-595-1764.