TRS-80®

UNIFY®

TUTORIAL MANUAL

Radio Shack®

10 9 8 7 6 5 4 3 2 1

CONTENTS

INTRODUCTION

This manual guides you through the process of developing an application system using Unify. The purpose is to show you how to use each Unify tool in combination with the others to implement an application system. You can use the tutorial as an "applied" reference manual, because each tool is explained in detail in the context of developing the application example. For a concise description of a single tool, consult the appropriate section of the Reference Manual.

You should be familiar with directories and with basic shell commands, and you should know how to use one of the TRS-XENIX text editors. You should also be familiar with concepts of storing and retrieving information. After you complete the tutorial, you know how to design, implement, and maintain application systems using Unify.

The application you will develop is for the Acme Hardware Warehouse, a company that supplies tools to retail hardware stores. Acme Hardware has a warehouse that stocks many different items, and each item must be tracked individually. When an item arrives, it is assigned a unique serial number, since the manufacturer's serial number might not be unique. The items come from many different manufacturers, each of which has a line of models that it supplies to the warehouse. The inventory may contain many instances (items) of any single model. This situation can be represented by three separate tables -- one for manufacturers, one for models, and one for items. The tables in turn consist of a number of columns, each of which describes an attribute of the entity (manufacturer, model, or item). The columns for each of these tables are as follows:

MANUFACTURER

manufacturer ID number	name	address

MODEL

model ID		description
model number	manufacturer ID number	

ITEM

	model ID			
serial number	model number	manufacturer ID number	acquisition date	sale price

The manufacturer table has columns containing the manufacturer's ID number, name, and address. The model table has columns for the model number, manufacturer ID number (so we know who makes this particular model), and a description. The item table contains a serial number, a model ID (so we know what model this item is), the date the item was put into inventory, and the sale price. Filling in a few rows of the tables produces a small database.

MANUFACTURER

manufacturer ID number	name	address
1ØØ	RH Smith Manufacturing	523 Galveston Ave.
1Ø1	Precision Tool Co.	26ØØ West 16th Street
1Ø2	A & H Industries, Inc.	2434 Evergreen Ave.

MODEL

model number	manufacturer ID number	description
1ØØ1	1ØØ	1/2" socket wrench
55271	1Ø1	combination pliers
1ØØ2	1Ø2	leather mallet

ITEM

serial number	model number	manufacturer ID number	acquisition date	sale price
1ØØ1	1ØØ1	1ØØ	2/15/82	$ 9.75
1ØØ6	55271	1Ø1	1/19/82	$ 6.89
1ØØ7	1ØØ2	1Ø2	1/19/82	$ 8.75

You can easily see the relationships between the tables by examining the data in the rows of the tables. For example, model number 1ØØ1 is made by RH Smith Manufacturing and model number 1ØØ2 is a leather mallet.

We use the following terms throughout this manual:

Field	The smallest unit of data recognized by the system. Used instead of "column."
Record	A number of fields grouped together into a single entity. Used instead of "row."
File	A number of records of the same type. Used instead of "table."

In this design, there is a one to many relationship between manufacturers and models and between models and items. Later it also includes customers and orders, and you can use it to group a collection of items on an order for a customer. Since one order can include several items, with each item representing a different model, many models can be represented on an order. A particular model, however, is not limited to a single order, since the inventory can contain many items of the same model type. As a result, the item becomes an intersection that creates a many to many relationship between models and orders. The expanded design can be pictured as follows:

MANUFACTURER

manufacturer ID number	name	address

MODEL

model ID		description
model number	manufacturer ID number	

ITEM

serial number	model ID		acquisition date	sale price	purchase price	order number
	model number	manufacturer ID number				

CUSTOMER

customer ID number	name	street address	city	state	zip code	phone number

ORDERS

order number	date	customer ID number

This design sequence illustrates the process of developing an inventory application for Acme. In this tutorial, you will create this application by following the steps described below:

1. Enter the database design. This description, which resides in the data dictionary, drives the whole applications system design.

2. Create an empty database file. This file contains a description of the database design, which provides dynamic binding of the design. The result is that existing programs do not have to change when you change the database design. This dramatically reduces maintenance costs associated with new releases of software.

3. Create some file maintenance screens using SFORM and ENTER. In most application systems, a large part of the development effort is creating file maintenance programs. With Unify, you can use ENTER rather than a program to implement most screens.

4. Create a simple single-level menu from which you can start the file maintenance screens. This shows how easy it is to create an initial menu structure.

5. Add users and access privileges to the data dictionary. This illustrates the program-level security features of the menu handler.

6. Use the ENTER screens for data entry. Entering test data lets you see how ENTER screens work, as well as providing a test database for queries and reports.

7. Modify the database design to reflect new requirements. Inevitably as systems are used, new capabilities that require database design modifications are necessary. This step shows how to make such changes using Unify.

8. Reconfigure the database to optimize the physical storage layout. Unify insulates applications programs from such changes because of the dynamic binding of the database description.

9. Use SQL to extract information from the database. Since this is a very simple database, it cannot be used to demonstrate

all of SQL's capabilities. (See the Reference Manual for examples of additional SQL features.)

10. Use the Listing Processor to extract information from the database. This section illustrates most of the interactive features of the Listing Processor.

11. Create "canned" reports with shell scripts that users can execute as programs. The last example in this section shows how to do repetitive form letters using information from the database to customize each letter. You can implement many applications entirely by using canned query/report scripts and ENTER screens.

12. Register the TRS-XENIX shell scripts constructed previously with the menu handler so that users can select them easily without having to interact with the shell.

13. Use the ENTER screens to query the database. ENTER provides a powerful and easy-to-use query-by-forms capability. Unsophisticated users can simply fill in the blanks on a screen form, and ENTER queries the database in the most efficient way possible to retrieve the indicated data.

14. Enter and display help documentation for programs. Online help documentation speeds up training by giving the user immediate access to the manual.

15. Create a tree-structured applications menu system. This illustrates the tremendous power and flexibility allowed by the menu handler to create custom menus for each user.

16. Write a data entry program in C. This shows how to create custom screens to fulfill specialized requirements. Unify has a complete host language interface, with more than 90 functions that provide database access, screen-manipulation capability, and report-writing functions.

When you complete these steps, you are familiar with the major components of Unify. The Reference Manual describes advanced features of the system, particularly those related to

tuning database performance, putting your database across several disk drives, and providing an environment with security controlled down to the field level.

Chapter 1
LOGGING IN TO UNIFY

The following conventions are basic to Unify and are used throughout this manual:

<ENTER> Terminate all lines typed in on the keyboard by pressing <ENTER> unless other instructions are given. <ENTER> moves the cursor forward.

<F1> Moves the cursor backward.

<F2> Stores a record.

Note: On the Model II/12/16, the function keys (<F1>,<F2>) are located either above or beside the numeric key pad. On the DT-1, hold down <SHIFT> and press <7> to initiate the <F1> function and hold down <SHIFT> and press <9> to initiate the <F2> function.

_ A box or an underscore on the screen that represents the cursor.

Installing Unify on Your Hard Disk

Please read all these directions before installing Unify on your hard disk. This lets you become familiar with the entire procedure and helps make the installation smoother. During the installation be sure to watch the screen and answer all the prompts.

To begin installing the package:

1. Turn on all peripherals and then turn on the computer.

2. When your screen displays the login prompt, type:

root <ENTER>

3. At the root prompt (#), type:

   install <ENTER>

4. The screen shows the Installation Menu, which contains these prompts:

   1. to install
   q. to quit

   Type **1 <ENTER>** to install Unify.

5. The following prompt appears:

   Insert diskette in Drive Ø and press <ENTER>

   Insert your TRS-XENIX Unify Diskette 1 in Drive Ø and press **<ENTER>**. The screen then displays the name of the package, the version number, and the catalog number.

6. In a few moments, the following information appears:

```
                UNIFY COMPONENTS TO BE INSTALLED
                --------------------------------

                                 Free Space Required
#  Name                        Runtime   Development    Disk #
-  ----------------------      -------   -----------    ------
1. Programs (required)         xxxx       xxxx
2. Libraries (required)        xxxx       xxxx
3. Tutorial (optional)         xxxx       xxxx


                          DISK CONFIGURATIONS

                                 Free Space Available (blocks)
#  Name                        Now       After Installation
-  ----------------------      -------   ------------------
1. /                           xxxxx The following prompt appears
on the bottom of the screen:

        Do you want to install the runtime system or the
        development system?
        Enter "r" for runtime, or "d" for development:_
```

If you have purchased the TRS-XENIX development system, which lets you write your own application programs, type d; otherwise, type r and press <ENTER>.

7. Asterisks (*) appear next to either the Runtime or Development components to indicate that your disk contains enough memory to install these various components. The following prompt appears on the bottom of the screen:

```
        On which disk would you like to install the UNIFY
        programs?
        Please enter a disk number (or "q" to quit):_
```

   Type in the number of the hard disk on which you wish to install the programs and press <ENTER>.

8. The number you choose appears in the Disk # column. A number also appears under Free Space Available -- After

Installation that show the amount of memory still remaining on your hard disk.

The following prompt appears on the bottom of the screen:

```
On which disk would you like to install the UNIFY
libraries?
Please enter a disk number (or "q" to quit):_
```

Type the appropriate hard disk number and press <ENTER>.

9. The number you choose appears on the screen to indicate the location of the Unify libraries. The number under Free Space Available -- After Installation changes to reflect the remaining hard disk space.

The following prompt appears on the bottom of the screen:

```
Do you want to install the tutorial?_
```

Type y and press <ENTER>.

1Ø. The following prompt appears:

```
On which disk would you like to install the UNIFY
tutorial?
Please enter a disk number (or "q" to quit):_
```

Again, enter the appropriate hard disk number.

11. The following prompt appears:

```
The UNIFY runtime [development] system will be
installed as indicated. Proceed?_
```

Type y and press <ENTER>. A new screen appears with the following messages:

```
The runtime [development] system will be installed
Reading first diskette
```

The screen scrolls through the files being installed, and at the end of the installation of each diskette, the screen asks you to insert the next diskette.

If you are installing the runtime system, the following message appears:

> The sixth diskette will not be needed, as it contains the development programs and libraries, and you are installing the runtime system.

After a short pause, the screen shows:

> The UNIFY diskettes have been read successfully.
>
> Installation complete - Remove the diskettes, then press <ENTER>_

Press **<ENTER>** to return to the Installation Menu. The screen shows:

> 1. to install
> q. to quit

Press **q** to quit and return to TRS-XENIX.

Now start up the Unify tutorial by typing:

> **ututorial <ENTER>**

The screen clears and the following question appears at the bottom:

> Do you wish to create a new data dictionary? _

Unify only asks this question if it doesn't find an existing data dictionary in the directory. Since we want to create a new data dictionary to begin the tutorial, type

> **<y> <ENTER>**

The screen then displays the following message:

```
Creating new data dictionary..._
```

When an empty data dictionary file (unify.db) is copied from the system lib directory, the Main Menu appears.

Note: Your screen displays today's date.

As an alternative to building the database according to the instructions in this manual, you may enter **runutut**, which automatically builds the tutorial database. To become familiar with the capabilities of Unify, however, it is best to continue with the tutorial as described in this manual. Nevertheless, if you want to examine this tutorial session as a reference, runutut automatically creates the database and runs the tutorial. Access privileges for the users SSS and jhd are included in the tutorial database to show you how to limit access to various users. To reference the tutorial database created by runutut as one of these users, create these users by running the TRS-XENIX utility mkuser.

```
+-------------------------------------------------------------+
|[entmenu]                      UNIFY SYSTEM                  |
|                          5 OCT 1982 - 15:25                 |
|                              Main Menu                      |
|                                                             |
| 1.  System Menu                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
| SELECTION: 1_                                               |
+-------------------------------------------------------------+
```

The cursor is at the SELECTION prompt, waiting for you to make a selection, either by number or by name. Any numbered line on the screen is a valid selection. The name in the upper left corner of the screen between square brackets (entmenu in this case) is the name of the currently executing menu or program.

Since each menu and program is named, you can use the SELECTION prompt as a command line by typing the name of the desired selection. Later on in the tutorial, we use program names to speed up the selection process.

This menu initially has only 1 option, but later when we register more programs and menus, we create a tree-structured

menu system. Type <1> to bring up the Unify System Menu, which appears as follows:

```
+-------------------------------------------------------------+
|[sysmenu]                      UNIFY SYSTEM                  |
|                          5 OCT 1982 - 15:25                 |
|                              System Menu                    |
|                                                             |
| 1.  Schema Maintenance          9.  MENUH Screen Menu       |
|                                                             |
| 2.  Schema Listing              1Ø.  MENUH Report Menu      |
|                                                             |
| 3.  Create Database             11.  Reconfigure Database   |
|                                                             |
| 4.  SFORM Menu                  12.  Write Database Backup  |
|                                                             |
| 5.  ENTER Screen Registration   13.  Read Database Backup   |
|                                                             |
| 6.  SQL - Query/DML Language    14.  Database Maintenance Menu
|                                                             |
| 7.  Listing Processor                                       |
|                                                             |
| 8.  Database Test Driver                                    |
|                                                             |
|SELECTION:  _                                                |
+-------------------------------------------------------------+
```

This menu is the root menu for all Unify system functions. The options are arranged in the order that applications development usually proceeds, from defining the schema, through creating data entry screens, and then queries and reports. You return often to this menu throughout the tutorial to select the various Unify tools.

Chapter 2
ENTERING A DATABASE DESIGN

The database design in the introduction is modeled as follows:

```
                        SCHEMA REPORTS
                        Schema Listing
RECORD/FIELD          REF        TYPE      LEN  LONG NAME
manf                  1Ø                        manufacturer
    *mano                        NUMERIC     4  number
     mname                       STRING     3Ø  name
     madd                        STRING     3Ø  address

model                 5Ø                        model
    *mokey                       COMB           model_key
       monum                     NUMERIC     7  model_num
       momano          mano      NUMERIC     4  manufacturer_num
     mdes                        STRING     3Ø  description

item                  1ØØ                       inventory_item
    *sno                         NUMERIC     9  serial_number
     imodel          mokey       COMB           model_id
     iad                         DATE           acquisition_date
     isal                        AMOUNT      5  sale_price
```

The column headed RECORD/FIELD lists the record names left justified, with the field names indented underneath. The number following the record name is an estimate of the expected number of records of that type. Thus, we expect 1Ø manf records, 5Ø model records, and 1ØØ item records. Unify uses these numbers to calculate the physical storage requirements of the database. These are estimates and can be increased as necessary.

An asterisk preceding a field name indicates that it is the primary key of the record. You may access a record randomly by

its primary key, which must be unique. You may also define nonunique secondary keys to speed up particular queries.

The REF column indicates the explicit relationships between the various files. The name of the primary key of another record goes in this column. For example, the reference that momano makes to mano indicates to the system that only valid manufacturer numbers can be stored in the model record. Attempts to store invalid numbers are rejected. Also, attempts to delete manufacturers that have associated models are not allowed. These enhancements to the pure relational data model allow Unify to be aware of logical consistency requirements that would otherwise have to be enforced by applications program coding.

The TYPE column indicates the data type of the field. The valid types are NUMERIC, STRING, FLOAT, DATE, TIME, AMOUNT, and COMBINED. LEN is the display length of the field on screens and reports. LONG NAME is a more descriptive name used by SQL, ENTER, and other system utilities.

Record Type Maintenance

To enter the database design, select 1 Schema Maintenance. The screen displays the first page of the schema entry form. You enter record characteristics on this page and field characteristics on Page 2 of the form.

(A Unify screen consists of a collection of data entry prompts, beside which you can enter data. You can move the cursor from one prompt to the next by typing <ENTER> and from one prompt to the previous one by typing <F1>.)

Initially, the cursor is over the 1 after the DATABASE ID prompt. Type **<ENTER>** to move the cursor down and display the prompt at the bottom of the screen as shown:

```
+----------------------------------------------------------------+
|[schent]                   UNIFY SYSTEM                         |
|                        5 OCT 1982 - 15:25                      |
|                        Schema Maintenance                      |
|                                                                |
|DATABASE ID: 1                                                  |
|                                                                |
|LN CMD RECORD   EXPECTED LONG NAME         DESCRIPTION          |
|                                                                |
|    '  '''''''' ''''''   ''''''''''''''''  '''''''''''''''''''' |
|                                                                |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|""  "  """""""" """"""   """"""""""""""""  """""""""""""""""""" |
|                                                                |
|[N]ext page, [P]rev page, [A]dd line, or number _               |
|                                                                |
+----------------------------------------------------------------+

        '''''''' => record data entry area
        """""""" -> record paging area
```

The design of this screen is representative of many screens in the Unify system. You add new entries to the data entry area at the top, and you modify existing entries in the paging area below. The screen displays existing entries in a page format, with 11 entries on a page.

The columns headed

LN CMD RECORD EXPECTED LONG NAME DESCRIPTION

are for the data entry area directly below.

[N]ext page, [P]rev page, [A]dd line, or number:

Paging area prompt, which lets you choose the operational mode for the paging area.

N - Displays the next page of database records.

P - Displays the previous page of database records.

A - Puts you in add mode so that you can add new database records by moving the cursor to the record data entry area.

Number (1-99) - Displays the page that contains the indicated database record and positions the cursor at that record to let you modify or delete it as indicated in the CMD column description.

LN

A line number the system assigns. You use it to reference the line if you wish to modify it or move it.

CMD

Area in which you enter a command to perform an operation on the current line. The cursor moves up and down in the page area of the screen in response to <F1> and <ENTER> within the column

headed CMD. The position of the cursor marks the current line. The following are valid commands:

F - Modify the fields for the current record by going to Page 2 of the screen form.

M - Modify EXPECTED, LONG NAME, or DESCRIPTION for the current record.

D - Delete the current record.

NUMBER (1-99) - Renumber the current record as indicated and reorder the records on the screen.

Q - Redisplay the paging prompt at the bottom of the screen.

RECORD

Unique 8-character record name. It can contain letters (upper- and lower-case), numbers, and the underscore character (_), and it must begin with a letter. You use this name to reference the record throughout the system design.

EXPECTED

The expected number of records of this type.

LONG NAME

A more descriptive record name of 16 characters that is used by ENTER and the database test driver to display error messages. It can contain letters (upper- and lower-case), numbers, and the underscore character (_), and it must begin with a letter.

DESCRIPTION

A comment field used for entering a description of the record.

To add new records to the schema, type <a> to select add mode in response to the prompt at the bottom of the screen. The prompt clears, and the cursor moves to the data entry area directly under the RECORD column. In the data entry area, to move the cursor forward (left to right) under the column headings, type <ENTER>; to move it backward (right to left), type <F1>.

If an entry is required at a prompt, typing **<ENTER>** does not move the cursor. For example, if the cursor is in the data entry area under the RECORD column, typing **<ENTER>** at this point has no affect. You must enter a record name before you can continue. Pressing <F1> while the cursor is in the RECORD column takes you out of add mode and redisplays the paging area prompt.

Once a name is entered in the RECORD column, you cannot return to that field by typing <F1> or <ENTER>. You must delete the entire record and reenter it to change the name. Enter the information as on the following screen form for the manufacturer record.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION

           manf      1Ø         manufacturer       _
```

When the cursor is in the position shown, you may go to Page 2 of the screen form. To display the next page of the form where fields for the current record are added, type <F2>.

Field Type Maintenance

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN   CMD   FIELD      KEY   REF        TYPE      LEN   LONG NAME      COMB. FIELD

      '    '''''''''   '    '''''''''  '''''''   '''   '''''''''''''  '''''''''

""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""
""    "    """""""""   "    """""""""  """""""   """   """""""""""""  """""""""

[N]ext page, [P]rev page, [A]dd line, or number _
```

```
'''''''' => field data entry area
"""""""" => field paging area
```

This page of the form works similar to Page 1, with a data entry area and a paging area.

[N]ext page, [P]rev page, [A]dd line, or number:

Paging area prompt, which lets you choose the operational mode for the paging area.

N - Displays the next page of fields for the current record.

P - Displays the previous page of fields for the current record.

A - Puts you in the add mode so that you can add new fields by moving the cursor to the field data entry area.

Number (1-99) - Displays the page that contains the indicated field and positions the cursor at that field to let you modify or delete it as indicated in the CMD column description.

RECORD

A display-only prompt that shows you to what record you are adding or modifying fields.

LN

A line number the system assigns. You may use it to reference a line that you wish to modify or move.

CMD

Area in which to enter a command that performs an operation on the current line. The cursor moves up and down in the paging area in response to <F1> and <ENTER> within the CMD column. The position of the cursor marks the current line. The following are valid commands:

M - Modify KEY, REF, TYPE, LEN, LONG NAME, or COMB. FIELD for the current field.

D - Delete the current field.

Number (1-99) - Renumber the current field as indicated and then reorder the fields on the screen.

Q - Redisplay the prompt at the bottom of the screen.

FIELD

Unique 8-character field name. It can contain letters (upper- and lower-case), numbers, and the underscore character (_), and it must begin with a letter. You use this name to reference the field throughout the system design.

KEY

An asterisk in this column indicates the current field is the primary key of the record.

REF

The name of the primary key of another record. This is used to establish the explicit relationships discussed earlier.

TYPE

The data type of the field. You need enter only the first character of the type; Unify completes the word automatically. The following are valid data types:

N - NUMERIC
F - FLOAT
S - STRING
D - DATE
T - TIME

A - AMOUNT
C - COMB

LEN

The default display length of the field on screens and reports. For NUMERIC and STRING fields, LEN is exactly the display length. For FLOAT fields, LEN has the form nnd, where nn indicates the total number of display positions, including the decimal point, and d the number of digits to the right of the decimal point. For example, a FLOAT field with a LEN of 123 has a default format of

nnnnnnnn.nnn

A FLOAT field with a LEN of 1ØØ has a default format of

nnnnnnnnnn

Note: There is no decimal point, so the format has 1Ø digits. For AMOUNT fields, LEN indicates the number of digits to the left of the decimal point, with 2 digits to the right of the decimal point assumed. For example, an AMOUNT field with a LEN of 6 has a default format of

nnnnnn.nn

on screens and reports. The total display length of AMOUNT fields is LEN + 3. Maximum lengths for the various field types are as follows:

NUMERIC - 9
FLOAT - 2ØØ
STRING - 256
DATE - n/a, defaults to 8
TIME - n/a, defaults to 5
AMOUNT - 11
COMB - n/a, computed by system

LONG NAME

A more descriptive name of 16 characters for the field. It can contain letters (upper- and lower-case), numbers, and the underscore character (_), and it must begin with a letter. SQL uses this name to identify fields when performing queries. The long name does not have to be unique throughout the database, as does the field name; it does have to be unique within this record type.

COMB. FIELD

The name of the combined field (that is, a field of type COMB) in the current record of which the current field is a part.

To add new fields for the current record, type <a> to select add mode in response to the prompt at the bottom of the screen. The prompt clears, and the cursor moves to the data entry area directly under the FIELD column. As in the previous screen, <ENTER> moves the cursor forward; <F1> moves it backward. <F1> at the first heading, FIELD, takes you out of add mode and redisplays the prompt at the bottom of the page. Enter the information as indicated on the following screen form:

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: manf

LN  CMD  FIELD     KEY  REF      TYPE     LEN  LONG NAME     COMB. FIELD

         mano      *             NUMERIC  4    number
```

After you fill in the LONG NAME, the cursor moves to the KEY column, not to the COMB. FIELD column. The key field of a record cannot be part of another COMB type field; hence, the cursor skips the column where this entry is made. The cursor skips a column if an entry in that column is illegal.

With the cursor positioned as above under the KEY heading, press <F2> to move the current field down into the paging area and clear the data entry area to add another field.

If you attempt to store a field without completing all required information, an error message is displayed on the bottom of the screen. For example, if you enter a field name and then press <F2>, the following message appears:

Type and length are incomplete. Delete?_

You can then type n to return the cursor to the REF column and continue adding the field information.

Now enter the remaining fields for the manufacturer record as indicated by the following screen:

```
[schent]                          UNIFY SYSTEM
 [A]DD                          5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN  CMD  FIELD     KEY  REF      TYPE     LEN  LONG NAME     COMB. FIELD

         _

1        mano       *            NUMERIC  4    number
2        mname                   STRING   3Ø   name
3        madd                    STRING   3Ø   address
```

Note: Each record can have only one primary key. Once a field has been defined as the key, the cursor skips the KEY column when you press <ENTER> or <F1>.

When the cursor is under the FIELD column, <F1> redisplays the paging prompt at the bottom of the screen. If you misspell a field name, and the field is already in the paging area, you must delete the field and reenter it. You can make any other change by typing the line number of the field in response to the paging prompt and then typing <M> in the CMD column. Press <ENTER> to get to the desired column and make the change. Type either <ENTER> or <F1> to get to the first modifiable column (whichever is faster), and then type <F2> to store the change.

When the cursor is in the FIELD column, pressing <F1> twice returns you to the following screen:

```
[schent]                    UNIFY SYSTEM
[A]DD                    5 OCT 1982 - 15:25
                         Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION

           _

1          manf      1Ø         manufacturer
```

The cursor is in the RECORD column. Note that the manufacturer record is in the paging area and has a line number. Since you are still in add mode, type in the data for the model record as shown next.

```
[schent]                        UNIFY SYSTEM
[A]DD                        5 OCT 1982 - 15:25
                             Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION

           model     5Ø         model              _

1          manf      1Ø         manufacturer
```

With the cursor in the DESCRIPTION column, type <F2> to add the fields for the model record. Now type <a> to get into add mode on the fields screen, and add the key field as indicated.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: model

LN   CMD   FIELD     KEY   REF      TYPE     LEN   LONG NAME      COMB. FIELD

           mokey     *             COMB           model_key
```

Now type <F2> to prepare for entering the next field, monum. Mokey moves down into the paging area and is assigned a line number before the cursor returns to the data entry area. The next screen diagram shows how to enter monum in the COMB. FIELD to indicate that it is a component of mokey.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: model

LN  CMD  FIELD     KEY  REF      TYPE      LEN  LONG NAME      COMB. FIELD

         monum               _    NUMERIC   7    model_num      mokey

1        mokey      *           COMB           model_key
```

After defining the monum field, type <F2> and the following screen appears. The program indents monum to show that it is a component of the field mokey.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: model

LN  CMD FIELD     KEY  REF      TYPE     LEN  LONG NAME      COMB. FIELD

        _

1       mokey      *             COMB          model_key
2         monum                  NUMERIC  7    model_num
```

Now add the next component of mokey as shown.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: model

LN  CMD  FIELD    KEY REF    TYPE     LEN  LONG NAME          COMB. FIELD

         momano       mano   NUMERIC  4    manufacturer_num   mokey

1        mokey     *         COMB           model_key
2          monum             NUMERIC  7    model_num
```

Now add the remaining model field. The screen appears as follows:

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: model

LN  CMD  FIELD     KEY REF     TYPE     LEN  LONG NAME          COMB. FIELD

         -
1        mokey      *          COMB          model_key
2          monum                NUMERIC  7    model_num
3          momano       mano   NUMERIC  4    manufacturer_num
4        mdes                   STRING   3Ø   description
```

Type <F1> twice to return to the record screen. Then add the last record type. The record screen appears as follows with the final record type (item) added.

```
[schent]                        UNIFY SYSTEM
[A]DD                        5 OCT 1982 - 15:25
                             Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION

           item      1ØØ        inventory_item     _

1          manf      1Ø         manufacturer
2          model     5Ø         model
```

Type <F2> to move to the next screen. Then type <a> for add mode.

Enter all the fields for the item record as shown on the next screen. The fields for the item record are shown on the next screen:

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: item

LN  CMD  FIELD     KEY  REF      TYPE      LEN  LONG NAME       COMB. FIELD


1        sno        *             NUMERIC   9    serial_number
2        imodel           mokey   COMB           model_id
3        iad                      DATE           acquisition_date
4        isal                     AMOUNT    5    sale_price









[N]ext page, [P]rev page, [A]dd line, or number _
```

Type <F1> twice to return to the record screen. Type <F1> again to go to the prompt at the bottom and then again to get to the DATABASE ID prompt.

```
[Schent]                    UNIFY SYSTEM
                          5 OCT 1982 - 15:25
                          Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION


1          manf      1Ø         manufacturer
2          model     5Ø         model
3          item      1ØØ        inventory_item
```

Type **<F1>** to return to the system menu. Press **<2> <ENTER>** to display the Schema Listing screen and print out the report.

```
[sysmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     System Menu

 1.  Schema Maintenance                9. MENUH Screen Menu

 2.  Schema Listing                    1Ø. MENUH Report Menu

 3.  Create Database                   11. Reconfigure Database

 4.  SFORM Menu                        12. Write Database Backup

 5.  ENTER Screen Registration         13. Read Database Backup

 6.  SQL - Query/DML Language          14. Database Maintenance Menu

 7.  Listing Processor

 8.  Database Test Driver

SELECTION: 2_
```

Schema Listing

The Schema Listing has no selection options. The program displays the screen below.

```
+-------------------------------------------------------------+
|[schlst]                    UNIFY SYSTEM                     |
|                          5 OCT 1982 - 15:25                 |
|                            Schema Listing                   |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|Formatting_                                                  |
+-------------------------------------------------------------+
```

The screen displays the System Menu while the report is printing.

```
[sysmenu]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                    System Menu

 1.  Schema Maintenance              9.  MENUH Screen Menu
 2.  Schema Listing                  1Ø.  MENUH Report Menu
 3.  Create Database                 11.  Reconfigure Database
 4.  SFORM Menu                      12.  Write Database Backup
 5.  ENTER Screen Registration       13.  Read Database Backup
 6.  SQL - Query/DML Language        14.  Database Maintenance Menu
 7.  Listing Processor
 8.  Database Test Driver

SELECTION:  _
```

Chapter 3
CREATING AN EMPTY DATABASE

At the System Menu, select 3, Create Database, to create an empty database file. Type <y> to answer the PROCEED confirmation prompt. To abort the function, type <n> or <F1>.

```
[crdb]                        UNIFY SYSTEM
                          5 OCT 1982 - 15:25
                           Create Database









PROCEED? y_
```

The program displays the prompts below. The database is created as an ordinary TRS-XENIX file in the current directory. Unify also lets you put the database on "raw" devices to achieve better performance. (See the Reference Manual, "Volume Maintenance," for further information.) After the process is

complete, type <ENTER> to answer the final prompt. The screen displays the system menu.

```
+-------------------------------------------------------------+
|[crdb]                      UNIFY SYSTEM                     |
|                        5 OCT 1982 - 15:25                   |
|                         Create Database                     |
|                                                             |
|** Phase I   **                                              |
|                                                             |
|** Phase II  **                                              |
|                                                             |
|** Phase III **                                              |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|Process complete. Back up the database ->->_                 |
+-------------------------------------------------------------+
```

Chapter 4
CREATING DATA ENTRY SCREENS

Now that an initial database file exists, you can use SFORM to create data entry screens with which you can add, inquire, modify, and delete records. You can use the following three screen forms, one each to maintain manufacturer records, model records, and item records.

```
[sman1ØØ]                    UNIFY SYSTEM
[A]DD                      5 OCT 1982 - 15:25
                        Manufacturer Maintenance

MANUFACTURER ID:1ØØ
NAME           :RH Smith Manufacturing
ADDRESS        :523 Galveston Ave.
```

```
[smodlØØ]                    UNIFY SYSTEM
[A]DD                      5 OCT 1982 - 15:25
                            Model Maintenance

MANUFACTURER ID:lØØ RH Smith Manufacturing
MODEL NUMBER   :lØØl
DESCRIPTION    :1/2" socket wrench
```

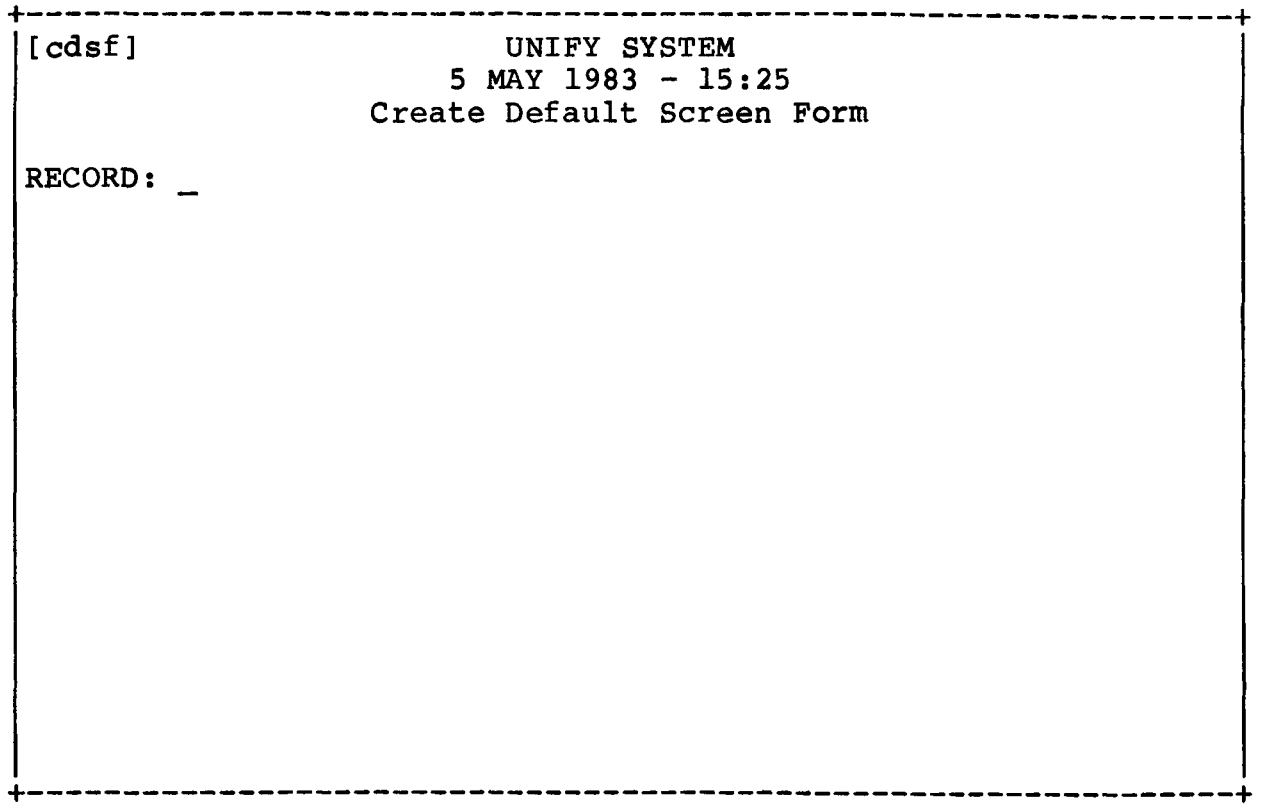
```
[sitml00]                    UNIFY SYSTEM
[A]DD                      5 OCT 1982 - 15:25
                            Item Maintenance

ITEM NUMBER :1001
MANUFACTURER:100      RH Smith Manufacturing
MODEL       :1001     1/2" socket wrench


ADD DATE    : 2/15/82
PRICE       :    9.75
```

Screen Maintenance

To create these forms, select 4, SFORM Menu, from the system menu. The screen displays the following menu:

```
[sfmenu]                        UNIFY SYSTEM
                              5 OCT 1982 - 15:25
                                 SFORM Menu

1. Screen Entry

2. Test Screen

3. Process Screen

4. Screen Reports

5. Restore Screen

6. List Screens

7. Create Default Screen Form



SELECTION: 7_
```

You can create screen forms using either option 1, Screen Entry, or option 7, Create Default Screen Form. The simplest way is to use the Default Screen Formatter. Type <7> **<ENTER>** and the screen shows the following:



Creating a Default Screen Form

You can enter the name of an existing record in response to the RECORD prompt and generate a default screen form for that record. After you create, process, and register the default screen form with ENTER, the screen name clears, and you can enter another record name. <F1> returns you to the SFORM Menu; <ENTER> has no effect.

The last message to appear before you can enter the name of another record is:

The screen can now be used for data entry ->->

To then use the screen, enter its name in response to the SELECTION prompt at the bottom of any menu.

You can modify or delete a default screen form using Screen Entry (see "Screen Entry" section). You can also change the heading by using ENTER Screen Registration (see "ENTER Screen Registration" section). Deleting a screen form cancels its registration with ENTER and removes it from the menu if it is registered.

In general, fields appear on the finished screen form from top to bottom, in order of their appearance in the schema reports. The fields line up, one under the other, starting after the longest long field name. The screen's heading is the record type's long or regular name, followed by the word "Maintenance." Any underscore characters become spaces, and appropriate letters are capitalized.

The prompts align in one, two, or three columns. A field's prompt is its long name (see the schema), or if that is blank, its regular or implicit name. In either case, all underscore characters are replaced with spaces. A database field is said to have an implicit name if it is a reference field. The implicit name consists of the field's regular name, followed by the name of the field it references. The referenced field may also reference another field, thereby adding to the implicit name.

The elements of a COMB field are given separate prompts. Also, if a field references a COMB field, separate prompts are produced for each of its referenced elementary fields.

For example, consider creating a default screen form for the manf record. Here is the manufacturer record layout:

RECORD/FIELD	REF	TYPE	LEN	LONG NAME
manf	1Ø			manufacturer
*mano		NUMERIC	4	number
mname		STRING	3Ø	name
cadd		STRING	3Ø	address

Select "Create Default Screen Form" from the SFORM menu, and enter the record name manf in response to the RECORD prompt. Processing messages appear as ENTER creates, processes, and registers the screen. When the process is complete, the screen appears as follows:

```
[cdsf]                    UNIFY SYSTEM
                      5 MAY 1983 - 15:25
                  Create Default Screen Form

RECORD: manf

The screen will be named manf


Created
Processed
Registered with ENTER









The screen can now be used for data entry. ->->_
```

Press **<F1>** twice to return to the menu, and type **manf** beside the SELECTION prompt. The following screen form, ready for data entry, appears:

```
+-------------------------------------------------------------+
|[manf]                      UNIFY SYSTEM                     |
|                        5 MAY 1983 - 15:25                   |
|                     Manufacturer Maintenance                |
|                                                             |
|number :                                                     |
|name   :                                                     |
|address:                                                     |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|                                                             |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE _                       |
+-------------------------------------------------------------+
```

Type **<F1>** to return to the SFORM menu.

Screen Entry

Now that you have a default screen form, enter a form using Screen Entry. Select 1, Screen Entry, to start up the interactive SFORM maintenance program. The following screen form appears:

```
+--------------------------------------------------------------------------+
|[sfmaint]                       UNIFY SYSTEM                              |
|                             5 OCT 1982 - 15:25                           |
|                                Screen Entry                              |
|                                                                          |
|SCREEN:########                                                           |
|                                                                          |
|LN SFIELD   DFIELD  TP LEN FX FY PROMPT                             PX PY|
|'' ''''''''  ''''''''''''''''''''''''''''''''''''''''''''''''''''         |
|                    '' ''' '' '' ''''''''''''''''''''''''''''''''  '' '' |
|                                                                          |
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|"" """"""""  """"""" "" """ "" "" """"""""""""""""""""""""""""""""  "" ""|
|                                                                          |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE _                                   |
|                                                                          |
+--------------------------------------------------------------------------+
```

```
     ######## => screen form data entry area
     '''''''' => screen field data entry area
     """""""" => screen field paging area
```

This form lets you inquire, add, modify, and delete screen forms and screen fields. There is a data entry area for screen forms, a data entry area for screen fields, and a paging area for screen fields. Following is an explanation of each prompt and column heading on the screen.

[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE

This prompt lets you choose an operational mode for the screen form data entry area of the screen.

I - Inquire, lets you see the screen fields for an existing screen form.

A - Add, lets you add new screen forms and associated screen fields.

M - Modify, lets you add, modify, and delete screen fields for an existing screen form.

D - Delete, lets you delete an existing screen form and all its screen fields.

SCREEN

Unique 8-character screen form name. You use this name to reference the screen throughout the system design.

[N]ext page, [P]rev page, [A]dd line, or number:

Paging area prompt that appears after you answer the screen prompt. The paging prompt lets you choose the operational mode for the paging area. The meaning of each selection is as follows:

N - Displays the next page of screen fields.

P - Displays the previous page of screen fields.

A - Lets you add new screen fields by moving the cursor to the screen field data entry area.

Number (1-99) - Displays the page that contains the indicated screen field and positions the cursor at the screen field to let you modify or delete it.

LN

A line number the system assigns. You use it to reference the line if you wish to modify it.

Column between LN and SFIELD

You use this column to enter a command to perform an operation on the current screen field. It corresponds to the column headed CMD on the schema entry screen. The line is too tight to permit a label. The cursor moves up and down in the page area of the screen in response to <F1> and <ENTER> in this column. The position of the cursor marks the current line. The following are valid commands:

M - Modify DFIELD, TP, LEN, FX, FY, PROMPT, PX, or PY for the current screen field.

D - Delete the current screen field.

Q - Redisplay the paging prompt at the bottom of the page.

SFIELD

A unique 8-character screen field name. You use this name to reference the screen field in programs you write; so it has to be unique from database record and field names. However, different screen forms may use the same screen field names.

DFIELD

Name of the database field to display for this screen field. If you leave this field blank, you must respond to the next two prompts (TP and LEN). If a database field has been specified, SFORM looks up the name in the data dictionary to validate it. You cannot use COMB type fields

directly with SFORM; you must enter the components of the field separately.

TP

Type of the database field. SFORM fills in this field from the data dictionary if you specified a database field name.

LEN

Display length of the database field. SFORM fills in this field from the data dictionary if you specified a database field name.

FX

X coordinate (column) of the database field on the screen form. It is a number in the range Ø to 79.

FY

Y coordinate (row) of the database field on the screen form. It can be a number in the range Ø to 23, but for ENTER screens, we recommend values of 3 to 2Ø. ENTER uses the last three lines of the screen, and the menu handler uses the first three lines.

PROMPT

Text displayed on the screen to indicate what data is being entered and/or displayed. You can enter literal text or "escaped" characters that indicate the prompt is to be displayed in a special mode. The escape character is the tilde (~)(<CTRL><6>). Model II/12/16 and Data Terminal-1 support the video mode.

~ r	-	Starts reverse video
~ s	-	Ends reverse video

In addition Data Terminal-1 supports the underline mode.

~ u	-	Starts underline
~ v	-	Ends underline

If you start a special display mode for a prompt, be sure to end it.

PX

X coordinate (column) of the prompt on the screen form. It is a number in the range Ø to 79.

PY

Y coordinate (row) of the prompt on the screen form. It can be a number in the range Ø to 23, but for ENTER screens, we recommend values in the range 3 to 2Ø. ENTER uses the last three lines of the screen, and the menu handler uses the first three lines.

To add the three screen forms, type <A> in response to the prompt at the bottom of the screen. The prompt disappears, and the cursor moves to the SCREEN prompt to let you add a new form. Add the manufacturer maintenance form, smanlØØ, as follows:

```
+-------------------------------------------------------------------+
|[sfmaint]                    UNIFY SYSTEM                          |
|[A]DD                     5 OCT 1982 - 15:25                       |
|                             Screen Entry                          |
|                                                                   |
|SCREEN: smanlØØ                                                    |
|                                                                   |
|LN SFIELD   DFIELD  TP LEN FX FY PROMPT                      PX PY |
|   _                                                               |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
+-------------------------------------------------------------------+
```

After you enter the form name, the cursor moves to the data entry area directly under SFIELD. In the data entry area, <ENTER> moves the cursor forward; <F1> moves it backward. Enter the first screen field as indicated next.

```
[sfmaint]                    UNIFY SYSTEM
[A]DD                      5 OCT 1982 - 15:25
                             Screen Entry

SCREEN: sman1ØØ

LN SFIELD    DFIELD   TP LEN FX FY PROMPT                    PX PY
   smano     mano
                       N 4   17 4  MANUFACTURER ID:          1  4
```

When you type <F2> with the cursor under DFIELD, the current screen field moves into the paging area to make room in the data entry area for a new screen field. Now add the screen fields for the sman1ØØ screen as follows:

```
[sfmaint]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                                  Screen Entry

SCREEN: sman1ØØ

LN SFIELD   DFIELD   TP LEN FX FY PROMPT                          PX PY


 1 smano    mano     N 4   17  4 MANUFACTURER ID:                1  4
 2 smname   mname    S 3Ø  17  5 NAME            :               1  5
 3 smadd    madd     S 3Ø  17  6 ADDRESS         :               1  6









[N]ext page, [P]rev page, [A]dd line, or number: _
```

When the cursor is under SFIELD, type <F1> to redisplay the paging prompt at the bottom as shown. Type <F1> again to clear the screen in preparation for adding the two following screen forms, one for model maintenance and the other for item maintenance:

SCREEN: smodlØØ

SFIELD	DFIELD	TP	LEN	FX	FY	PROMPT	PX	PY
smomano	momano	N	4	17	4	MANUFACTURER ID:	1	4
smname	mname	S	3Ø	22	4		22	4
smonum	monum	N	7	17	5	MODEL NUMBER :	1	5
smdes	mdes	S	3Ø	17	6	DESCRIPTION :	1	6

SCREEN: sitmlØØ

SFIELD	DFIELD	TP	LEN	FX	FY	PROMPT	PX	PY
sitmno	sno	N	9	16	4	ITEM NUMBER :	1	4
smanf	imodel_momano	N	4	16	5	MANUFACTURER :	1	5
smname	mname	S	3Ø	25	5		23	5
smodl	imodel_monum	N	7	16	6	MODEL :	1	6
sdesc	mdes	S	3Ø	25	6		23	6
sadate	iad	D	8	16	8	ADD DATE :	1	8
sprice	isal	A	5	16	9	PRICE :	1	9

When constructing SFORM screens, be aware of the following ENTER conventions.

(1) An ENTER screen manipulates only the specified "target" record. (The target record is defined in ENTER Screen Registration, discussed later in this section.) Fields from other record types ("secondary" fields) are display only. The screen displays the contents of secondary fields only if an earlier field on the screen is an explicit reference field from the target record to that record type. For example, the model screen displays the manufacturer name because it is preceded by the manufacturer id number. The manufacturer id number is an explicit reference from the model to the manufacturer.

(2) Place all parts of a combination field together on the screen. ENTER gathers all parts of a combination field in sequence,

no matter where they are on the screen, but if the parts of a combination field are scattered, the cursor moves down the screen in a scattered manner instead of in an orderly manner.

(3) In ADD mode the record key is accepted before any other field. If the key is a combination, all portions must be accepted and validated. Therefore, it is usually best to position the record key at the top of the screen.

(4) All screen fields must have both the database field coordinates, FX and FY, and the prompt coordinates, PX and PY, filled in, even if there is no database field or prompt. ENTER uses the FX-FY combination when erasing data from the screen. ENTER uses the PX-PY combination to determine how the cursor moves from one field to the next. If FX-FY is zero, the top line of the screen is erased when the screen clears. If PX-PY is zero, the cursor moves erratically from one field to the next.

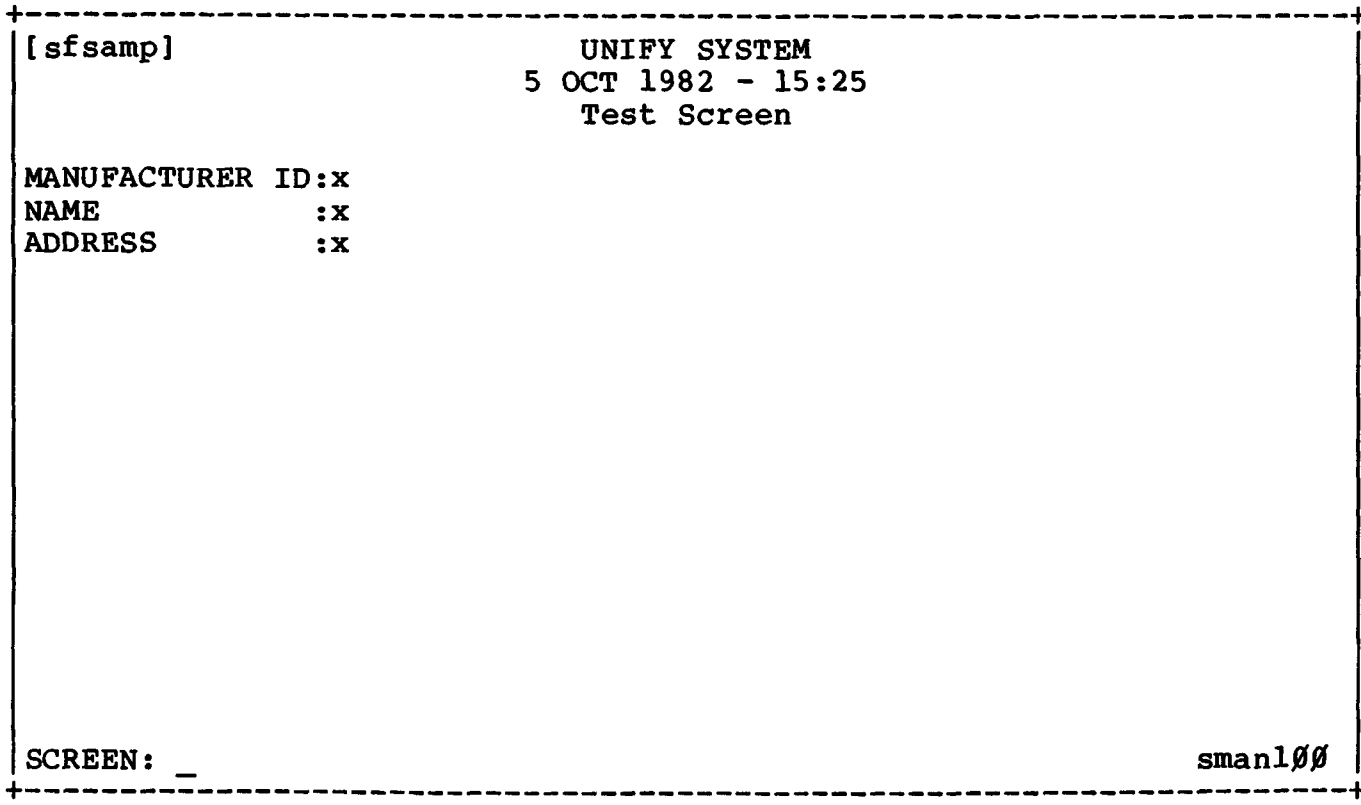
(5) ENTER uses the last three lines of the screen (Lines 21, 22, 23) to display information, prompts, and error messages. If you try to use these lines to display your own data, ENTER overwrites your data.

Test Screen

After you enter the three screens, type <F1> three times to return to the SFORM menu. Select 2, Test Screen, to try out the screen forms just entered. The following screen appears:

```
+-------------------------------------------------------------------------+
|[sfsamp]                          UNIFY SYSTEM                           |
|                                5 OCT 1982 - 15:25                       |
|                                   Test Screen                           |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|SCREEN: sman1ØØ_                                                         |
+-------------------------------------------------------------------------+
```

Answer the SCREEN prompt by typing the name of the first screen form, **sman1ØØ.** The following screen appears:



If the screen on your terminal does not duplicate the one above, go back to the Screen Entry program (sfmaint) and check your design. If sman1ØØ appears correct, enter the next form name, **smod1ØØ**, in response to the prompt. The following screen appears:

```
+-------------------------------------------------------------------+
|[sfsamp]                          UNIFY SYSTEM                     |
|                                5 OCT 1982 - 15:25                 |
|                                   Test Screen                     |
|                                                                   |
|MANUFACTURER ID:x     x                                            |
|MODEL NUMBER    :x                                                 |
|DESCRIPTION     :x                                                 |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SCREEN: _                                                smodlØØ   |
+-------------------------------------------------------------------+
```

Type **sitmlØØ** in response to the prompt. The following screen appears:

```
+-------------------------------------------------------------------+
|[sfsamp]                          UNIFY SYSTEM                     |
|                                5 OCT 1982 - 15:25                 |
|                                   Test Screen                     |
|                                                                   |
|ITEM NUMBER     :x                                                 |
|MANUFACTURER    :x          x                                      |
|MODEL           :x          x                                      |
|                                                                   |
|                                                                   |
|ADD DATE        :x                                                 |
|PRICE           :x                                                 |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SCREEN: _                                                sitml00   |
+-------------------------------------------------------------------+
```

Now type <F1> to return to the SFORM menu. The screen forms are now in the data dictionary where you can easily modify them. However, in order for ENTER or custom programs to access them, they must first be processed.

Process Screen

Select Option 3, Process Screen, to start the program. The following screen appears:

```
[sfproc]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                  Process Screen

SCREEN: sman1ØØ_
```

Enter the form names (sman1ØØ, smod1ØØ, sitm1ØØ) one at a time in response to the SCREEN prompt. It takes a few seconds to process a form. Then the screen clears, and the program is ready for another form name. After you enter the form names, type **<F1>** to return to the SFORM menu.

Screen Reports

Select **4**, Screen Reports, to produce a hard-copy listing of the screen forms. You can use the listing as documentation for the system or as a form of offline backup. The program uses the following screen:

```
+-------------------------------------------------------------------+
|[sfrep]                          UNIFY SYSTEM                      |
|                               5 OCT 1982 - 15:25                  |
|                                  Screen Reports                   |
|                                                                   |
|SCREEN _                                                           |
|                                                                   |
|LISTING [Y or N]:                                                  |
|PRINT SCREEN:                                                      |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
+-------------------------------------------------------------------+
```

The meaning of each prompt is as follows:

SCREEN

Name of an existing screen form to print. You can also print all forms in the data dictionary by responding with ALL to this prompt.

LISTING [Y or N]

Type <Y> to print a tabular report that lists the attributes of each screen field. Type <N> if you do not want to print this part of the report.

PRINT SCREEN

Type <Y> to print an image of the screen as it appears on a CRT. Type <N> if you do not want to print the image.

Respond to the prompts as shown in the following screen to print the screens just entered.

```
[sfrep]                          UNIFY SYSTEM
                               5 OCT 1982 - 15:25
                                  Screen Reports

SCREEN  all

LISTING [Y or N]:y
PRINT SCREEN:    y

Running









Formatting_
```

While the report is being printed, the screen displays the SFORM menu.

Now that the screen forms are created, you have two options. You can register the screens with ENTER to provide automatic data entry with no programming, or you can write programs using the applications function libraries provided with Unify to drive the screens. In the following examples, you use ENTER to drive the screens.

ENTER Screen Registration

To register a screen form with ENTER, type <F1> to return to the system menu. Select 5, ENTER Screen Registration, and the following screen appears:

```
+-------------------------------------------------------------------+
|[entmnt]                          UNIFY SYSTEM                     |
|                                5 OCT 1982 - 15:25                 |
|                           ENTER Screen Registration               |
|                                                                   |
|SCREEN NAME:                                                       |
|                                                                   |
|TARGET RECORD:                                                     |
|                                                                   |
|HEADING:                                                           |
|                                                                   |
|REPORT SCRIPT 1:                                                   |
|              2:                                                   |
|              3:                                                   |
|              4:                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE _                             |
+-------------------------------------------------------------------+
```

This is a simple single-level screen that lets you register one screen at a time with ENTER. The meaning of the prompts is as follows:

[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE:

This prompt lets you choose an operational mode for the program.

I - Inquire, lets you see a screen form already registered with ENTER.

A - Add, lets you register screen forms with ENTER.

M - Modify, lets you modify TARGET RECORD, HEADING, and REPORT SCRIPT 1 through 4 for a screen form already registered with ENTER.

D - Delete, lets you remove the ENTER registration for a screen form.

SCREEN NAME

Name of the screen form to register.

TARGET RECORD

Name of the record to be maintained by the screen form (that is, the one added, modified, and so on). You must specify the target record because fields from many different records may be displayed on the screen form.

HEADING

When ENTER is executing, the screen displays a string of up to 34 characters on Line 3. The screen also displays the heading on any menus on which the screen form is placed. You can enter literal text or "escaped" text that indicates the prompt is to be displayed in a special mode. The escape character is the tilde (~) (<CTRL><6>). Model II/12/16 and Data Terminal-1 support reverse video.

~r	-	Starts reverse video
~s	-	Ends reverse video

In addition, Data Terminal-1 supports the underline mode.

~u	-	Starts underline
~v	-	Ends underline

If you start a special display mode for a heading, be sure to end it. The screen displays the heading in the

specific mode only on menus and the third line of the screen form.

```
REPORT SCRIPT 1
              2
              3
              4
```

Each of these prompts lets you enter an 8-character name of a report script to be associated with the screen. Thus, you can associate as many as four different reports with a given screen. The report script is a TRS-XENIX file in the bin directory that contains commands to the Listing Processor on how to format the report. (This manual discusses reporting in more detail in following sections.)

Type **<a>** in response to the prompt at the bottom to begin registering screen forms with ENTER. Register smanlØØ as follows:

```
[entmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                            ENTER Screen Registration

SCREEN NAME:    sman1ØØ

TARGET RECORD: manf

HEADING:        Manufacturer Maintenance

REPORT SCRIPT 1:_
              2:
              3:
              4:
```

Type <F2> to clear the screen, and then register smod1ØØ and sitm1ØØ as follows:

```
[entmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                            ENTER Screen Registration

SCREEN NAME:    smodlØØ

TARGET RECORD: model

HEADING:        Model Maintenance

REPORT SCRIPT 1:
              2:
              3:
              4:
```

```
[entmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                            ENTER Screen Registration

SCREEN NAME:    sitml00

TARGET RECORD: item

HEADING:        Item Maintenance

REPORT SCRIPT 1:
              2:
              3:
              4:
```

After registering the screen forms with ENTER, return to the system menu by typing <F1> twice. At this point, the menu handler can call ENTER directly to execute these screens; simply type the name of the desired screen in response to the SELECTION menu prompt. You may, however, want to put the screens on a menu for people who don't know or can't remember the screen names. The next section explains how to do this.

Chapter 5
CREATING MENUS

Select 9, MENUH Screen Menu, to get to the menu maintenance option. The following menu appears:

```
[scrmen]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                MENUH Screen Menu

1. Executable Maintenance

2. Menu Maintenance

3. Group Maintenance

4. Employee Maintenance

5. Enter Help Documentation

6. Program Loading

7. System Parameter Maintenance



SELECTION: 2_
```

Select 2, Menu Maintenance, to start the interactive menu maintenance program. The following screen appears:

```
+--------------------------------------------------------------------+
|[menumnt]                        UNIFY SYSTEM                       |
|                              5 OCT 1982 - 15:25                    |
|                               Menu Maintenance                     |
|                                                                    |
|NAME: ######## HEADING: #################################           |
|CMD  LINE  MENU/PROG  M/P   PROMPT                                  |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
| "      ""   """"""""    "    """""""""""""""""""""""""""""""""     |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE _                             |
|                                                                    |
+--------------------------------------------------------------------+
```

```
     ######## => menu data entry area
     """""""" => menu line data entry area
```

This form lets you inquire, add, modify, and delete menus and menu lines. There is a data entry area for the menu name and description and a multiline data entry area for menu lines. Following is an explanation of each prompt and column heading.

[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE

This prompt lets you choose an operational mode for the program.

I - Inquire, lets you see an existing menu.

A - Add, lets you create new menus.

M - Modify, lets you add, modify, and delete lines on an existing menu.

D - Delete, lets you delete an entire menu.

NAME

An 8-character menu name. The name must be distinct from other menu, screen form, and program names.

HEADING

A 34-character string that appears on Line 3 of the screen when the menu is displayed and on other menus on which this menu is a selection. You can enter literal text or "escaped" text that indicates the prompt is to be displayed in a special mode. The escape character is the tilde (~), which is generated when you type <CTRL> <6>. Model II/12/16 and Data Terminal-1 support the reverse video.

~r	-	Starts reverse video
~s	-	Ends reverse video

In addition, Data Terminal-1 supports the underline mode.

~u	-	Starts underline
~v	-	Ends underline

If you start a special display mode for a heading, be sure to end it. The screen displays the special mode

only on menus from which this menu screen may be selected and not on Line 3 of the heading for this menu.

CMD

Use this column to enter a command to perform an operation on the current menu line. The cursor moves up and down in the multiline data entry area of the screen in response to <F1> and <ENTER>. The position of the cursor marks the current menu line. The following are valid commands:

A - Add a new line to the menu. This response is valid only on the first empty line in the multiline area. In other words, you have to move the cursor to the first blank line before you can add a new menu line.

M - Modify LINE or MENU/PROG for the current menu line.

D - Delete the current menu line.

Q - Leave the multiline data entry area and position the cursor at the HEADING prompt.

LINE

Position on the menu that the current menu line occupies. You may enter any number in the range 1 to 16, and the menu lines on the screen shift accordingly.

MENU/PROG

Name of the menu, program, or ENTER screen that is to be executed when you select this menu line.

M/P

Display-only field that indicates the type of the above entry. The possibilities are:

M - Current menu line starts a menu.

P - Current menu line starts a program.

E - Current menu line starts an ENTER screen.

PROMPT

Display-only field that indicates the text to appear on the menu for the current menu line.

To add the ENTER screens to the initial menu, entmenu, select M for modify mode. The cursor moves to the NAME prompt. Enter the menu name, and the screen displays entmenu as follows:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Menu Maintenance

NAME: entmenu  HEADING: Main Menu
CMD   LINE   MENU/PROG   M/P    PROMPT
        1    sysmenu      M     System Menu
```

Since you do not want to change the heading, type <ENTER> to get to the multiline data entry area. To add the three ENTER screens defined earlier -- smanlØØ, smodlØØ, and sitmlØØ -- as well as SQL for some examples that come later, type <ENTER> until the cursor is at the first blank line on the screen. Type <A> <ENTER> to add a new line, and then enter the line as shown. Note that, as with the other screens, <ENTER> moves the cursor forward, and <F1> moves it backward.

Note: If you inadvertently type <A>, Unify requires that you complete the process. Go ahead and enter a line number and press <F2>. You can then delete the erroneous entry by pressing <ENTER> or <F1> to move the cursor next to the incorrect menu line and by typing <D> <ENTER>.

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Menu Maintenance

NAME: entmenu  HEADING: Main Menu
CMD   LINE   MENU/PROG   M/P    PROMPT
        1    sysmenu      M     System Menu
        1    smanlØØ      E     Manufacturer Maintenance
```

Notice the line is numbered 1, since you want it at the top of the menu. Type <F2> with the cursor in the LINE column as shown to complete the entry of the menu line. The lines sort according to the numbers in the LINE column, with the most recently modified line coming first if two lines have the same number. To enter the next line, type <A> <ENTER> again, since you are already at a blank line. Now add the new line for smodlØØ as shown.

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Menu Maintenance

NAME: entmenu  HEADING: Main Menu
CMD   LINE   MENU/PROG   M/P    PROMPT
        1    smanlØØ      E     Manufacturer Maintenance
        2    sysmenu      M     System Menu
        2    smodlØØ      E     Model Maintenance
```

Type <F2> to reorder the lines on the screen. Now you can add sitmlØØ in the same way you added the previous two entries. The menu appears as follows:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Menu Maintenance

NAME: entmenu  HEADING: Main Menu
CMD   LINE   MENU/PROG   M/P    PROMPT
        1    sman1ØØ      E     Manufacturer Maintenance
        2    smod1ØØ      E     Model Maintenance
        3    sitm1ØØ      E     Item Maintenance
        4    sysmenu      M     System Menu
 a_
```

Now add SQL as Line 4, giving you a menu that looks as follows:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Menu Maintenance

NAME: entmenu  HEADING: Main Menu
CMD   LINE   MENU/PROG   M/P    PROMPT
        1    smanlØØ      E     Manufacturer Maintenance
        2    smodlØØ      E     Model Maintenance
        3    sitmlØØ      E     Item Maintenance
        4    sql          P     SQL - Query/DML Language
        5    sysmenu      M     System Menu
 q_
```

Type <Q> to return rapidly to the HEADING prompt. Then type <F1> three times to get to the MENUH menu.

Chapter 6
PROGRAM LEVEL SECURITY

Now that you have an application menu and several database maintenance screens, you may want to restrict access to those screens. This requires login id's other than that of root for most people. Group Maintenance and Employee Maintenance give you the means to restrict program usage for users other than root, who has access to all menus and programs.

With Group Maintenance, you define an employee group, the Data Entry Clerks (DEC), that has add, inquire, and modify privileges for manufacturers, models, and items, but that does not have access to the system functions menu. Then, using Employee Maintenance, add a clerk and a supervisor. In addition to the access privileges defined for the data entry clerks, the supervisor has delete capability for the three previous programs and has access to SQL.

Group Maintenance

Select 3 (Group Maintenance) as follows to add a new employee group to the data dictionary.

```
[scrmen]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                MENUH Screen Menu

1. Executable Maintenance

2. Menu Maintenance

3. Group Maintenance

4. Employee Maintenance

5. Enter Help Documentation

6. Program Loading

7. System Parameter Maintenance



SELECTION: 3_
```

The following screen appears:

```
+----------------------------------------------------------------------------------+
|[grpmnt]                        UNIFY SYSTEM                                      |
|                           5 OCT 1982 - 15:25                                     |
|                            Group Maintenance                                     |
|                                                                                  |
|GROUP ID: ####   NAME: ##############################                            |
|                                                                                  |
|SYSTEM ENTRY PT: ######## - #                                                     |
|                                                                                  |
|ACCESS LEVELS:                                                                    |
| LN  MENU/PROG  M/P  INQ  ADD  MOD  DEL   LN  MENU/PROG  M/P  INQ  ADD  MOD  DEL  |
| ''  ''''''''    '    '    '    '    '                                           |
|                                                                                  |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
| ""  """"""""    "    "    "    "    "    ""  """"""""    "    "    "    "    "   |
|                                                                                  |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE:  _                                          |
|                                                                                  |
+----------------------------------------------------------------------------------+
```

```
    ######## => employee group data entry area
    '''''''' => access privilege data entry area
    """""""" => access privilege paging area
```

This form lets you inquire, add, modify, and delete employee groups and access privileges. There is a data entry area for employee groups, a data entry area for access privileges, and a paging area for access privileges. Following is an explanation of each prompt and column heading.

[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE:

This prompt lets you choose an operational mode for the employee group data entry area of the screen.

I - Inquire, lets you see the access privileges for an existing employee group.

A - Add, lets you add new employee groups and associated access privileges.

M - Modify, lets you add, modify, and delete access privileges for an existing employee group.

D - Delete, lets you delete an existing employee group and all its access privileges.

GROUP ID

A 4-character code to identify the employee group.

NAME

A 3Ø-character employee group name, used for identification and documentation only.

SYSTEM ENTRY PT

Name of the initial menu that employees in this group see when they log in to Unify.

[N]ext page, [P]rev page, [A]dd line, or number:

Paging area prompt that appears after you answer the SYSTEM ENTRY PT prompt. The paging prompt lets you choose the operational mode for the paging area.

N - Displays the next page of access privileges.

P - Displays the previous page of access privileges.

A - Lets you add new access privileges by moving the cursor to the access privilege data entry area.

Number (1-999) - Displays the page that contains the indicated access privilege and positions the cursor at that access privilege to let you modify or delete it, as described next.

Column to the left of LN

You use this area to enter a command to perform an operation on the current access privilege. It corresponds to the CMD column on the schema entry screen. The line is too tight to permit a label. The cursor moves up and down in the page area of the screen in response to <F1> and <ENTER> in this column. The position of the cursor marks the current line. The following are valid commands:

M - Modifies INQ, ADD, MOD, or DEL access privileges if it is for a program or for an ENTER screen.

D - Deletes access privileges for this menu, program, or ENTER screen.

Q - Redisplays the paging prompt at the bottom of the page.

LN

A line number the system assigns. You use it to reference the access privilege if you wish to modify it.

MENU/PROG

Name of a menu, a program, or an ENTER screen that employees in this group can access.

M/P

A display-only field that indicates the type of entry made in the MENU/PROG column. The possibilities are:

M - The current access privilege is for a menu.

P - The current access privilege is for a program.

E - The current access privilege is for an ENTER screen.

INQ

Type <Y> in this column to indicate that employees in this group may use inquire mode for a program or an ENTER screen. Type <N> to restrict this usage.

ADD

Type <Y> in this column to indicate that employees in this group may use add mode for a program or an ENTER screen. Type <N> to restrict this usage.

MOD

Type <Y> in this column to indicate that employees in this group may use modify mode for a program or an ENTER screen. Type <N> to restrict this usage.

DEL

Type <Y> in this column to indicate that employees in this group may use delete mode for a program or an ENTER screen. Type <N> to restrict this usage.

To add an employee group, type <A> <ENTER> in response to the prompt at the bottom of the screen. The prompt clears, and the cursor moves to the GROUP ID prompt to let you add a new employee group. Add the Data Entry Clerk group as shown:

```
[grpmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                                Group Maintenance

GROUP ID: DEC     NAME: Data Entry Clerks

SYSTEM ENTRY PT: entmenu   - M

ACCESS LEVELS:
 LN  MENU/PROG M/P INQ ADD MOD DEL  LN  MENU/PROG M/P INQ ADD MOD DEL

[N]ext page, [P]rev page, [A]dd line, or number: a_
```

After you enter the system entry point, the paging area prompt appears at the bottom of the screen as shown. Type <A> <ENTER> beside this prompt to add access privileges. The prompt disappears, and the cursor moves to the access privilege data entry area directly under the MENU/PROG heading. In the data entry area, <ENTER> moves the cursor forward; <F1> moves it backward. Type <F1> when the cursor is in the MENU/PROG column to get out of the current mode and redisplay the

paging prompt at the bottom of the screen. Enter the first access privilege as indicated next.

```
[grpmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                                Group Maintenance

GROUP ID: DEC     NAME: Data Entry Clerks

SYSTEM ENTRY PT: entmenu   - M

ACCESS LEVELS:
 LN   MENU/PROG M/P INQ ADD MOD DEL   LN   MENU/PROG M/P INQ ADD MOD DEL
      sman1ØØ    E   y   y   y   n
```

Type <F2> with the cursor under the INQ heading, and the current access privilege moves into the paging area to make room in the data entry area for a new access privilege. Finish adding access privileges for the data entry clerks as follows:

```
[grpmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                                Group Maintenance

GROUP ID: DEC     NAME: Data Entry Clerks

SYSTEM ENTRY PT: entmenu   - M

ACCESS LEVELS:
 LN   MENU/PROG M/P INQ ADD MOD DEL   LN   MENU/PROG M/P INQ ADD MOD DEL

  1   smanlØØ    E   y   y   y   n
  2   smodlØØ    E   y   y   y   n
  3   sitmlØØ    E   y   y   y   n

[N]ext page, [P]rev page, [A]dd line, or number:
```

Remember, typing <F1> when the cursor is under the MENU/PROG column in the data entry area redisplays the paging prompt at the bottom as shown. Type <F1> four times to get to the MENUH menu.

Employee Maintenance

Now that you have defined access privileges for a group, you must specify the employees that are within this group. To do so, select 4, Employee Maintenance. The following screen appears:

```
+-------------------------------------------------------------------------+
|[empmnt]                          UNIFY SYSTEM                           |
|                                5 OCT 1982 - 15:25                       |
|                              Employee Maintenance                       |
|                                                                         |
|LOGIN ID: #### NAME: ##############################                     |
|GROUP ID: #### NAME: ##############################                     |
|                       SYSTEM ENTRY PT: ########  - #                    |
|                                                                         |
|ACCESS LEVELS DIFFERENT FROM GROUP:                                      |
|  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL                            |
|  ''  ''''''''   '     '       '   '   '   '                             |
|                                                                         |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|  "" """""""""   "     "       "   "   "   "                             |
|                                                                         |
|[I]QUIRE, [A]DD, [M]ODIFY, [D]ELETE                                      |
|                                                                         |
+-------------------------------------------------------------------------+
```

```
######## => employee data entry area
'''''''' => access privilege data entry area
"""""""" => access privilege paging area
```

This form lets you inquire, add, modify, and delete employees and access privileges. The access privileges on this screen let you specify differences from those of the group to which this employee belongs. In other words, the privileges of the group determine the basic privileges for the employee, but you may vary the privileges of individual employees.

There is a data entry area for employees, a data entry area for access privileges, and a paging area for access privileges. Following is an explanation of each prompt and column heading.

[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE:

This prompt lets you choose an operational mode for the employee data entry area of the screen.

I - Inquire, lets you see the access privileges for an existing employee.

A - Add, lets you add new employees and associated access privileges.

M - Modify, lets you add, modify, and delete access privileges for an existing employee.

D - Delete, lets you delete an existing employee and all access privileges.

LOGIN ID

A 4-character code that identifies the employee. The employee uses this code to log in to Unify.

NAME

A 3Ø-character string for the employee's name. This is for documentation and identification only.

GROUP ID

A 4-character code to identify the group to which the employee belongs.

NAME

A display-only field that shows the name of the employee group.

SYSTEM ENTRY PT

Name of the initial menu that the employee sees when logging in. It defaults to the entry point of the group, but you can change it for the individual.

[N]ext page, [P]rev page, [A]dd line, or number:

Paging area prompt that appears after you answer the SYSTEM ENTRY PT prompt. The paging prompt lets you choose the operational mode for the paging area. The meaning of each selection is as follows:

N - Displays the next page of access privileges.

P - Displays the previous page of access privileges.

A - Lets you add new access privileges by moving the cursor to the access privilege data entry area.

Number (1-999) - Displays the page that contains the indicated access privilege and positions the cursor at that access privilege to let you modify or delete it.

Column to the left of LN

You use this area to enter a command to perform an operation on the current access privilege. You use it in the same way as you use the corresponding column on the Group Maintenance screen. The cursor moves up and down in the page area of the screen in response to <F1> and <ENTER> in this column. The position of the cursor

marks the current line. The following are valid commands:

M - Modifies ACCESS, INQ, ADD, MOD, or DEL for the current access privilege.

D - Deletes the current access privilege.

Q - Redisplays the paging prompt at the bottom of the page.

LN

A line number the system assigns. You use it to reference the access privilege if you wish to modify it.

MENU/PROG

Name of a menu, program, or ENTER screen to which the employee has a different access privilege than that of others in the group.

M/P

A display-only field that indicates the type of entry made in the MENU/PROG column. The possibilities are:

M - The current access privilege is for a menu.

P - The current access privilege is for a program.

E - The current access privilege is for an ENTER screen.

ACCESS

Type <Y> in this column to indicate that an employee has access to the indicated menu, program, or ENTER screen. Type <N> to restrict this access.

INQ

If access permitted, you may type <Y> in this column to indicate that an employee may use inquire mode for the program or ENTER screen. Type <N> to restrict this usage.

ADD

If access is permitted, you may type <Y> in this column to indicate that an employee may use add mode for the program or ENTER screen. Type <N> to restrict this usage.

MOD

If access is permitted, you may type <Y> in this column to indicate that an employee may use modify mode for the program or ENTER screen. Type <N> to restrict this usage.

DEL

If access is permitted, you may type <Y> in this column to indicate that an employee may use delete mode for the program or ENTER screen. Type <N> to restrict this usage.

To add an employee, type <A> <ENTER> in response to the prompt at the bottom of the screen. The prompt disappears, and the cursor moves to the LOGIN ID prompt to let you add a new employee. Add the supervisor of the Data Entry Clerk group as shown:

```
[empmnt]                          UNIFY SYSTEM
[A]DD                           5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                    SYSTEM ENTRY PT: entmenu  - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROD M/P  ACCESS?  INQ ADD MOD DEL




[N]ext page, [P]rev page, [A]dd line, or number: a
```

Type <ENTER> to get past the system entry point, leaving it at the default value. The paging area prompt then appears at the bottom of the screen as shown. Type <A> <ENTER> beside this prompt to add access privileges different from those of the group. The prompt clears, and the cursor moves to the access privilege data entry area directly under the MENU/PROG heading. In the data entry area, <ENTER> moves the cursor forward; <F1> moves it backward. Typing <F1> when the cursor is in the MENU/PROG column takes you out of the current mode and redisplays the paging prompt at the bottom of the screen. Enter the first access privilege as indicated next.

```
[empmnt]                          UNIFY SYSTEM
[A]DD                           5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                     SYSTEM ENTRY PT: entmenu    - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL
      sman1ØØ    E     y       y   y   y   y
```

Note that the supervisor has delete capability, which goes beyond the capabilities of the group. Type <F2> with the cursor under the ACCESS heading, and the current access privilege moves into the paging area to make room in the data entry area for a new access privilege. Finish adding access privileges for the supervisor as follows:

```
[empmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                     SYSTEM ENTRY PT: entmenu    - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL

   1  sman1ØØ    E     y       y   y   y   y
   2  smod1ØØ    E     y       y   y   y   y
   3  sitm1ØØ    E     y       y   y   y   y
   4  sql        P     y

[N]ext page, [P]rev page, [A]dd line, or number:
```

Remember, typing <F1> when the cursor is under the MENU/PROG column in the data entry area redisplays the paging prompt at the bottom as shown. Type <F1> three times to clear the screen but remain in ADD mode.

Add the clerk in the same manner. Note that since none of the clerk's access privileges are different from those of the group, no additional entries are required. When you are finished, the following screen appears:

```
[empmnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: jhd  NAME: John H. Doe
GROUP ID: DEC  NAME: Data Entry Clerks
                     SYSTEM ENTRY PT: entmenu    - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL




[N]ext page, [P]rev page, [A]dd line, or number:
```

To see how the menu handler controls system security, type <F1> seven times to get to the login screen. Define two system users, one with the login ID of sss and one with the ID of jhd. Execute mkuser for each user to be defined and enter the appropriate responses. Then log in as Sam Spade, the supervisor. In response to the system prompt ($), enter ututorial to execute the tutorial.

```
+-------------------------------------------------------------------+
|[entmenu]                          UNIFY SYSTEM                    |
|                                 5 OCT 1982 - 15:25                |
|                                     Main Menu                     |
|                                                                   |
|1. Manufacturer Maintenance                                        |
|                                                                   |
|2. Model Maintenance                                               |
|                                                                   |
|3. Item Maintenance                                                |
|                                                                   |
|4. SQL - Query/DML Language                                        |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION:                                                         |
|                                                                   |
+-------------------------------------------------------------------+
```

Type <F1> to return to the system prompt. Log in as John Doe, data entry clerk, by typing **login jhd**. At the system prompt, reexecute the Unify tutorial by typing **ututorial** <ENTER>.

```
+-------------------------------------------------------------------+
|[entmenu]                        UNIFY SYSTEM                      |
|                              5 OCT 1982 - 15:25                   |
|                                  Main Menu                        |
|                                                                   |
|1. Manufacturer Maintenance                                        |
|                                                                   |
|2. Model Maintenance                                               |
|                                                                   |
|3. Item Maintenance                                                |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: _                                                       |
+-------------------------------------------------------------------+
```

Note that the screen does not display SQL and system menu lines as selection options. This is how logging in as a different user effectively restricts access to programs as defined in Group Maintenance and Employee Maintenance. Even if you know the name of a restricted program, the menu handler prevents you from executing it.

Chapter 7
ADDING DATA WITH ENTER

Now add some test data so that you have a database for queries and reports. Select 1, Manufacturer Maintenance, and the following screen appears:

```
[sman1ØØ]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                              Manufacturer Maintenance

MANUFACTURER ID:
NAME            :
ADDRESS         :









[I]NQUIRE, [A]DD, [M]ODIFY  _
```

This screen operates in the same manner as all ENTER screens. These screens deal with a single target record, which is the one being added, deleted, modified, or inquired upon. Fields from other records

("secondary" records) may appear on the screen to create "views" of the database. These secondary fields are for display only. For ENTER to work properly, the target record and any secondary record must be explicitly related through a REF entry in Schema Maintenance.

An ENTER screen has four modes of operation -- inquire, add, modify, and delete. One or more of the modes may be disallowed, depending on your access privileges as defined with Group Maintenance and Employee Maintenance. In add mode, the primary key of the target record is obtained first. If the key is composed of multiple fields, all fields must appear on the screen. In the remaining three modes, you may find records by filling in any desired prompt(s) on the screen. ENTER looks for records that match. (This is explained in more detail in the "Query by Forms" section.)

When ENTER adds or finds the record, the cursor moves to the first prompt on the screen. From there, <ENTER> moves the cursor down the screen; <F1> moves it up the screen. When you reach the last prompt, <ENTER> loops the cursor back to the top. Typing <F2> at the first prompt erases the data in preparation for the next operation. Typing <F1> redisplays the operation mode prompt at the bottom.

To change the data displayed at a prompt, type <ENTER> until you are positioned at the prompt. Then enter the new data and type <ENTER>. If ENTER accepts the data, it is stored in the database, and the cursor moves to the next prompt. If ENTER does not accept the data, an error message appears at the bottom of the screen, and the database is not updated. Type <ENTER> to acknowledge the error, and then enter the data again.

If you start to type some data and change your mind, type <F1> to abort the entry without changing the database. To change the data displayed from a secondary record, you must define another ENTER screen with that secondary record as the target. Of course, a custom program can deal with multiple target records on the same screen (similar to many of the Unify programs, such as the SFORM maintenance program, sfmaint).

Now add some manufacturer records to the database. Type <A> <ENTER> to select add mode, and add the first manufacturer as follows:

```
[sman1ØØ]                          UNIFY SYSTEM
[A]DD                            5 OCT 1982 - 15:25
                               Manufacturer Maintenance

MANUFACTURER ID:1ØØ
NAME           :RH Smith Manufacturing
ADDRESS        :523 Galveston Ave.









[I]NQUIRE, [A]DD, [M]ODIFY _
```

Type <F2> to clear the screen of data. Now you are ready to add the next manufacturer. Here is a complete list of the manufacturers in the test database:

ID	NAME	ADDRESS
1ØØ	RH Smith Manufacturing	523 Galveston Ave.
1Ø1	Precision Tool Co.	26ØØ West 16th Street
1Ø2	A & H Industries, Inc.	2434 Evergreen Ave.
1Ø3	Grover Parts and Supplies	9462 Jackson Road
1Ø4	The Tool Depot	7562 Orange Dr.
1Ø5	BHP Ltd.	17385 Weatherby Rd.

After you add the manufacturers, type <F1> to return to the Main Menu. Then select Model Maintenance by entering 2 at the SELECTION prompt.

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance









SELECTION: 2
```

The next two pages show the model screen with the first model entered and a list of all models to be entered in the test database.

```
[smodlØØ]                          UNIFY SYSTEM
[A]DD                            5 OCT 1982 - 15:25
                                  Model Maintenance

MANUFACTURER ID:1ØØ  RH Smith Manufacturing
MODEL NUMBER   :1ØØ1
DESCRIPTION    :1/2" socket wrench
```

ID	NAME	MODEL	DESCRIPTION
1ØØ	RH Smith Manufacturing	1ØØ1	1/2" socket wrench
		1ØØ2	3/4" socket wrench
		1ØØ3	1/2" open end wrench
1Ø1	Precision Tool Co.	55Ø71	1/2" socket wrench
		55171	1/2" box end wrench
		55271	combination pliers
		55371	needle nose pliers
1Ø2	A & H Industries, Inc.	1ØØ1	vise grips
		1ØØ2	leather mallet
1Ø3	Grover Parts and Supplies	61117	3" Phillips screwdriver
		71117	3" slotted screwdriver
		81117	6" Phillips screwdriver
		91117	6" slotted screwdriver
1Ø4	The Tool Depot	1Ø11	combination pliers
		1Ø12	1/2" socket wrench

Type <F1> to return to the Main Menu; then select Item Maintenance. Following is the screen with the first item added and a list of all items to be entered in the test database.

```
[sitml00]                          UNIFY SYSTEM
[A]DD                            5 OCT 1982 - 15:25
                                  Item Maintenance

ITEM NUMBER     :1001
MANUFACTURER    :100       RH Smith Manufacturing
MODEL           :1001      1/2" socket wrench


ADD DATE        : 02.15.82
PRICE           :    9.75
```

You can enter dates using dot (.) as the delimiter, as well as the slash (/). This lets you enter dates from the numeric keypad very rapidly.

ITEM NO	MANF	MODEL	ADD DATE	PRICE
1ØØ1	1ØØ	1ØØ1	Ø2/15/82	9.75
1ØØ2	1ØØ	1ØØ1	Ø2/15/82	9.75
1ØØ3	1ØØ	1ØØ1	Ø2/15/82	9.75
1ØØ4	1Ø1	55Ø71	Ø2/23/82	1Ø.25
1ØØ5	1Ø1	55Ø71	Ø2/25/82	1Ø.75
1ØØ6	1Ø1	55271	Ø2/15/82	6.89
1ØØ7	1Ø2	1ØØ2	Ø1/19/82	8.75
1ØØ8	1Ø3	61117	Ø1/15/82	2.35
1ØØ9	1Ø3	61117	Ø2/15/82	2.69
1Ø1Ø	1Ø3	91117	Ø1/15/82	2.89
1Ø11	1Ø3	91117	Ø2/15/82	3.19
1Ø12	1Ø4	1Ø11	Ø2/Ø8/82	5.5Ø
1Ø13	1Ø4	1Ø12	Ø2/Ø8/82	9.5Ø
1234	1ØØ	1ØØ2	Ø2/23/82	15.ØØ

After you enter the data, type <F1> to return to the Main Menu.

Chapter 8
MODIFYING A DATABASE DESIGN

The inventory system as it now stands is suitable only for keeping track of items on hand. To create orders for customers, print order forms, and track the difference between inventory cost and sale price, you must add some new record types and fields to the database design. The new record types required are as follows:

RECORD/FIELD	REF	TYPE	LEN	LONG NAME
customer	1Ø			customer
*cnum		NUMERIC	5	customer_number
cname		STRING	3Ø	name
caddr		STRING	3Ø	address
ccity		STRING	2Ø	city
cstate		STRING	2	state
czip		NUMERIC	5	zip_code
cphone		STRING	14	phone_number
orders	1ØØ			orders
*onum		NUMERIC	9	order_number
odate		DATE		date_ordered
ocust	cnum	NUMERIC	5	customer_number

The modified fields and new fields to be added to existing record types follow. The additional address fields in the manufacturer record are added to give a more realistic address capability. The name field is lengthened to allow for some new manufacturers with longer names.

```
RECORD: manf

MODIFIED
FIELDS: mname                   STRING    35  name

NEW
FIELDS: mcity                   STRING    2Ø  city
        mstate                  STRING     2  state
        mzip                    NUMERIC    5  zip_code

RECORD: item

NEW
FIELDS: iorder     onum         NUMERIC    9  order_number
        ipamt                   AMOUNT     5  purchase_price
```

The complete modified database design is shown next.

SCHEMA REPORTS
Schema Listing

RECORD/FIELD		REF	TYPE	LEN	LONG NAME
manf	1Ø				manufacturer
*mano			NUMERIC	4	number
mname			STRING	35	name
madd			STRING	3Ø	street_address
mcity			STRING	2Ø	city
mstate			STRING	2	state
mzip			NUMERIC	5	zip_code
model	5Ø				model
*mokey			COMB		model_key
monum			NUMERIC	7	model_num
momano		mano	NUMERIC	4	manufacturer_num
mdes			STRING	3Ø	description
item	1ØØ				inventory_item
*sno			NUMERIC	9	serial_number
imodel		mokey	COMB		model_id
iad			DATE		acquisition_date
isal			AMOUNT	5	sale_price
iorder		onum	NUMERIC	9	order_number
ipamt			AMOUNT	5	purchase_price
customer	1Ø				customer
*cnum			NUMERIC	5	customer_number
cname			STRING	3Ø	name
caddr			STRING	3Ø	address
ccity			STRING	2Ø	city
cstate			STRING	2	state
czip			NUMERIC	5	zip_code
cphone			STRING	14	phone_number
orders	1ØØ				orders
*onum			NUMERIC	9	order_number
odate			DATE		date_ordered
ocust		cnum	NUMERIC	5	customer_number

To make these changes, log out and then log in as root, since John Doe does not have access to Schema Maintenance. Type <F1> to back out to the system prompt. Then log in as root and execute **ututorial.**

Start up the schema entry program by entering **schent** in response to the SELECTION prompt on the Main Menu. You may enter the name of a program or menu at the SELECTION prompt on any menu to bypass levels of menus and directly execute any allowed program.

In this example, Schema Entry is an option within the System Menu. By typing the program name, you can bypass the Main Menu and the System Menu. When you type <F1> to exit Schema Entry, Unify returns you to the Main Menu from which you selected schent.

```
[entmenu]                    UNIFY SYSTEM
                          5 OCT 1982 - 15:25
                              Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: schent_
```

The following screen appears. Type <ENTER> and select **add** mode to add the new records.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION


1          manf      1Ø         manufacturer
2          model     5Ø         model
3          item      1ØØ        inventory_item









[N]ext page, [P]rev page, [A]dd line, or number: a_
```

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD     EXPECTED   LONG NAME          DESCRIPTION

           customer   1Ø         customer           _

1          manf       1Ø         manufacturer
2          model      5Ø         model
3          item       1ØØ        inventory_item
```

Add the customer record as shown. Then type <F2> to go to the fields screen to add the fields for the record.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: customer

LN   CMD   FIELD     KEY   REF       TYPE     LEN   LONG NAME     COMB. FIELD




[N]ext page, [P]rev page, [A]dd line, or number: a_
```

Type <A> <ENTER> to select add mode as above to add the fields for the customer record. Add the first field as shown next.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: customer

LN   CMD   FIELD     KEY   REF     TYPE      LEN   LONG NAME        COMB. FIELD

           cnum       *            NUMERIC   5     customer_number
```

With the cursor in the KEY column, type <F2> to move the field into the paging area. Then add the rest of the fields as shown next.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: customer

LN  CMD  FIELD     KEY  REF    TYPE      LEN  LONG NAME        COMB. FIELD


1        cnum       *          NUMBERIC  5    customer number
2        cname                 STRING    3Ø   name
3        caddr                 STRING    3Ø   address
4        ccity                 STRING    2Ø   city
5        cstate                STRING    2    state
6        czip                  NUMERIC   5    zip code
7        cphone                STRING    14   phone number
```

Now type <F1> twice to return to the record screen, where you add the orders record.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION

           orders    1ØØ        orders             _

1          manf      1Ø         manufacturer
2          model     5Ø         model
3          item      1ØØ        inventory_item
4          customer  1Ø         customer
```

Type <F2> to go to the fields screen to add the fields for the record.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: orders

LN  CMD  FIELD    KEY  REF      TYPE    LEN  LONG NAME    COMB. FIELD




[N]ext page, [P]rev page, [A]dd line, or number a_
```

Type <A> <ENTER> to select add mode as above to add the fields for the orders record. Add the first field as shown next.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: orders

LN   CMD   FIELD     KEY   REF     TYPE      LEN   LONG NAME      COMB. FIELD

           onum       *            NUMERIC   9     order_number
```

With the cursor in the KEY column, type <F2> to move the field into the paging area. Then add the rest of the fields as shown next.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: orders

LN  CMD  FIELD     KEY  REF     TYPE     LEN  LONG NAME       COMB. FIELD


1        onum       *           NUMERIC  9    order number
2        odate                  DATE          date ordered
3        ocust           cnum   NUMERIC  5    customer number
```

Now type <F1> twice to return to the record screen. Type <F1> to display the paging prompt as shown. Then type <1> <ENTER> to position the cursor at the manufacturer record.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION


1          manf      1Ø         manufacturer
2          model     5Ø         model
3          item      1ØØ        inventory_item
4          customer  1Ø         customer
5          orders    1ØØ        orders




[N]ext page, [P]rev page, [A]dd line, or number 1_
```

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION


1    f_    manf      1Ø         manufacturer
2          model     5Ø         model
3          item      1ØØ        inventory_item
4          customer  1Ø         customer
5          orders    1ØØ        orders
```

Type <F> <ENTER> to modify the fields of the manufacturer record. We will lengthen the name field and add the new fields.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN  CMD  FIELD     KEY  REF     TYPE     LEN  LONG NAME      COMB. FIELD


1        mano       *           NUMERIC  4    number
2        mname                  STRING   3Ø   name
3        madd                   STRING   3Ø   address









[N]ext page, [P]rev page, [A]dd line, or number 2_
```

Type <2> <ENTER> to modify the name field. The cursor moves to the specified line in the CMD column.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN  CMD  FIELD    KEY  REF     TYPE     LEN  LONG NAME      COMB. FIELD


1        mano      *           NUMERIC  4    number
2    m_  mname                 STRING   3Ø   name
3        madd                  STRING   3Ø   address
```

Type <M> <ENTER> as shown to modify the length of mname. Press <ENTER> to position the cursor below the LEN column. Change 3Ø to 35.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN  CMD  FIELD     KEY  REF     TYPE      LEN  LONG NAME     COMB. FIELD


1        mano      *            NUMERIC   4    number
2        mname           _      STRING    35   name
3        madd                   STRING    3Ø   address
```

Type <F2> to store the change. To display the paging prompt at the bottom of the screen, press <ENTER> twice.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN  CMD  FIELD    KEY  REF      TYPE     LEN  LONG NAME     COMB. FIELD


1        mano      *             NUMERIC  4    number
2        mname                   STRING   35   name
3        madd                    STRING   3Ø   address









[N]ext page, [P]rev page, [A]dd line, or number a_
```

Type <A> <ENTER> so that you can add the new address fields necessary. Add the first field as shown.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN  CMD  FIELD     KEY  REF       TYPE     LEN  LONG NAME     COMB. FIELD

         mcity          _         STRING   2Ø   city

1        mano       *             NUMERIC  4    number
2        mname                    STRING   35   name
3        madd                     STRING   3Ø   address
```

Type <F2> to move the field down into the paging area. Add the other fields as shown next.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: manf

LN  CMD  FIELD     KEY  REF      TYPE     LEN   LONG NAME     COMB. FIELD

         _

1        mano       *            NUMERIC  4     number
2        mname                   STRING   35    name
3        madd                    STRING   3Ø    address
4        mcity                   STRING   2Ø    city
5        mstate                  STRING   2     state
6        mzip                    NUMERIC  5     zip_code
```

Type <F1> twice to return to the record screen.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD     EXPECTED   LONG NAME          DESCRIPTION



1          manf       1Ø         manufacturer
2          model      5Ø         model
3     f_   item       1ØØ        inventory_item
4          customer   1Ø         customer
5          orders     1ØØ        orders
```

Type <ENTER> to position the cursor on the item record line. Then type <F> as shown to modify the item record fields.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

RECORD: item

LN  CMD  FIELD     KEY  REF     TYPE     LEN  LONG NAME        COMB. FIELD


1        sno        *           NUMERIC  9    serial_number
2        imodel           mokey  COMB          model_id
3        iad                     DATE          acquisition_date
4        isal                    AMOUNT   5    sale_price









[N]ext page, [P]rev page, [A]dd line, or number a_
```

To add the two new fields, type <A> <ENTER> and add the first new field as shown next.

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: item

LN  CMD  FIELD     KEY  REF      TYPE     LEN  LONG NAME      COMB. FIELD

         iorder         onum     NUMERIC  9    order_number

1        sno       *             NUMERIC  9    serial_number
2        imodel         mokey    COMB           model_id
3        iad                     DATE           acquisition_date
4        isal                    AMOUNT   5     sale_price
```

Type <F2> to move the field down into the paging area. Add the final new field, and the screen appears as follows:

```
[schent]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                               Schema Maintenance

RECORD: item

LN   CMD   FIELD    KEY   REF      TYPE     LEN   LONG NAME          COMB. FIELD

           _

1          sno      *              NUMERIC  9     serial_number
2          imodel          mokey   COMB           model_id
3          iad                     DATE           acquisition_date
4          isal                    AMOUNT   5     sale_price
5          iorder          onum    NUMERIC  9     order_number
6          ipamt                   AMOUNT   5     purchase_price
```

This completes the schema changes.

```
[schent]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                Schema Maintenance

DATABASE ID: 1

LN   CMD   RECORD    EXPECTED   LONG NAME          DESCRIPTION



1          manf      1Ø         manufacturer
2          model     5Ø         model
3          item      1ØØ        inventory_item
4          customer  1Ø         customer
5          orders    1ØØ        orders
```

Type <F1> repeatedly until the Main Menu appears. Select Schema Listing as follows, and check the results against the listing given earlier in this section.

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: schlst_
```

The Schema Listing program name is entered in response to the SELECTION prompt on the Main Menu in this example; however, this program could have been selected from a list of menu options in the System Menu.

Chapter 9
RECONFIGURING A DATABASE

After you change the schema, you must reconfigure the database to reflect the new database design. First, however, make a backup of the database file in case a hardware or software failure occurs during the reconfiguration process. Select Write Database Backup (budb) from the Main Menu as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: budb_
```

You can select Database Backup by its program name, as shown in this example, or you can select it as a menu option from within the System Menu. The following screen appears:

```
+-------------------------------------------------------------------+
|[budb]                          UNIFY SYSTEM                       |
|                             5 OCT 1982 - 15:25                    |
|                          Write Database Backup                    |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|PROCEED? Y_                                                        |
+-------------------------------------------------------------------+
```

Insert a formatted diskette and make sure it is write-enabled before responding to the above prompt. When the backup is finished, the screen displays the menu. To stop the backup process, type <N> <ENTER> or <F1> <ENTER> in response to the above prompt.

Now select Reconfigure Database (scom) as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: scom_
```

You can also select Reconfigure Database as an option within the System Menu. The following screen appears:

```
+-------------------------------------------------------------------+
|[scom]                          UNIFY SYSTEM                       |
|                            5 OCT 1982 - 15:25                     |
|                          Reconfigure Database                     |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|PROCEED? y_                                                        |
+-------------------------------------------------------------------+
```

Type <Y> in response to the prompt to start the process. To stop the process and return to the menu, type <N> or <F1>. The screen displays the following messages next:

```
+-----------------------------------------------------------------------+
|[scom]                          UNIFY SYSTEM                           |
|                            5 OCT 1982 - 15:25                         |
|                          Reconfigure Database                         |
|                                                                       |
|** Phase I   **                                                        |
|                                                                       |
|** Phase II  **                                                        |
|                                                                       |
|                                                                       |
|Nominal hash table loading factor:  5Ø%                                |
|New load factor without rebuilding: 52%                                |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|REBUILD HASH INDEX? n_                                                 |
+-----------------------------------------------------------------------+
```

At this point, the schema is compiled and reconfiguration is about to begin. Since you increased the total expected number of records in the database, you may here increase the size of the hash table. This is a very small increase, however, so type <N> <ENTER> to skip the hash table rebuild step. The following prompts appear:

```
+-------------------------------------------------------------------+
|[scom]                          UNIFY SYSTEM                       |
|                              5 OCT 1982 - 15:25                   |
|                            Reconfigure Database                   |
|                                                                   |
|** Phase I   **                                                    |
|                                                                   |
|** Phase II  **                                                    |
|                                                                   |
|** Phase III **                                                    |
|                                                                   |
|Nominal hash table loading factor:  5Ø%                            |
|New load factor without rebuilding: 52%                            |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|use diskette as the temporary file? n_                             |
+-------------------------------------------------------------------+
```

This question gives you an opportunity to reconfigure to diskette. If the database file is too large to copy in the space available on your hard disk, type <Y> in response to this prompt. In this case, the small test database can be reconfigured entirely on hard disk. Type <N> <ENTER>.

When the process is complete, back up the database again by selecting Write Database Backup (budb) as before. The database file is reconfigured, and the data dictionary is updated to reflect the new schema design.

You do not need to recompile or reload existing programs because the schema description is bound dynamically. However, you must modify the screen forms, which already exist for the manf and item records, to

add the new fields. Use the Screen Entry program as described in the "Screen Entry" section of Chapter 4.

You can select the Screen Entry program by entering the program name sfmaint in response to the SELECTION prompt on any menu or as a menu option within the SFORM Menu that can be accessed through the System Menu.

Enter the screen form name (sman1ØØ) for the manf record, and then select a to add lines. Be sure to locate the data entry fields and screen prompts as follows:

DFIELD	FX	FY	PX	PY	PROMPTS:
mcity	17	7	1	7	CITY :
mstate	17	8	1	8	STATE :
mzip	17	9	1	9	ZIP CODE :

Type <F1> to clear the screen in preparation for adding lines to the item screen form, sitm1ØØ. Be sure to locate the data entry fields and screen prompts as follows:

DFIELD	FX	FY	PS	PY	PROMPTS:
iorder	16	1Ø	1	1Ø	ORDER NUMBER :
ipamt	16	11	1	11	PURCHASE PRICE:

You do not need to register either screen with ENTER Screen Registration, but you must regenerate the screen forms to allow ENTER or custom programs to access the new fields by running the Process Screen (sfproc) program as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: sfproc_
```

You can also select the Process Screen program as an option within the SFORM Menu. The following screen appears:

```
+----------------------------------------------------------------------+
|[sfproc]                       UNIFY SYSTEM                            |
|                           5 OCT 1982 - 15:25                          |
|                             Process Screen                            |
|                                                                       |
|SCREEN: _                                                              |
|                                                                       |
+----------------------------------------------------------------------+
```

Type the names of the forms you modified -- sman1ØØ and sitm1ØØ -- one at a time. When you are finished, type <F1> to return to the Main Menu. Then select System Menu, 5.

To create screen forms for the new customer and orders records, select Create Default Screen Form. You can execute Create Default Screen Form directly from the System Menu by entering the program name, cdsf. For this example, however, enter **sfmenu** to go directly to the SFORM Menu and select Option 7. In response to the RECORD prompt, enter the names of the new records one at a time. Now type <F1> twice to get to the System Menu.

You can now use the new screen forms to add the following data by entering the record name in response to the SELECTION prompt. (To review data entry, see Chapter 7, "Adding Data with ENTER.")

For the customer screen enter:

```
customer number: 1
name           : Smith & Sons Hardware
address        : 1234 State Street
city           : Wheatville
state          : CA
zip code       : 95817
phone number   : 916-924-ØØ75

customer number: 2
name           : Creative Manufacturing
address        : 9124 Industrial Blvd.
city           : Redding
state          : CA
zip code       : 921Ø2
phone number   : 916-91Ø-9999

customer number: 3
name           : Reliable Construction Co.
address        : 2113 Folsom Blvd.
city           : Sacramento
state          : CA
zip code       : 92343
phone number   : 916-945-3434
```

After you enter all the customer data, type <F1> twice to get to the System Menu. Type **orders** in response to the selection prompt, and add the following data:

```
order number    : 1
date ordered    : Ø4/Ø2/82
customer number: 1

order number    : 2
date ordered    : Ø4/Ø2/82
customer number: 3

order number    : 3
date ordered    : Ø6/28/82
customer number: 2
```

After you enter all the data, type <F1> twice to get to the System Menu.

To add data in the new fields of the item record, type **sitmlØØ** in response to the SELECTION prompt on the System Menu. To add the order number and purchase price for each item, select **modify** mode, and the following screen appears:

```
[sitml00]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Item Maintenance

ITEM NUMBER    :
MANUFACTURER   :
MODEL          :

ADD DATE       :
PRICE          :
ORDER NUMBER   :
PURCHASE PRICE:
```

Add the order numbers and purchase prices shown below.

ITEM NO	ORDER NUMBER	PURCHASE PRICE
1ØØ1	1	7.25
1ØØ2	3	7.35
1ØØ3	1	7.25
1ØØ4	1	8.9Ø
1ØØ5	3	9.25
1ØØ6	1	6.ØØ
1ØØ7	1	5.25
1ØØ8	2	2.2Ø
1ØØ9	2	2.ØØ
1Ø1Ø	1	3.ØØ
1Ø11	1	2.2Ø
1Ø12	1	5.75
1Ø13	2	9.23
1234	2	13.23

Type <F1> three times to return to the Main Menu.

Chapter 1Ø
USING SQL

The major Unify query language is Structured Query Language, or SQL (pronounced see-quel). SQL is an English keyword-oriented query language of great power and flexibility. It is easy enough for nonprogrammers to learn, yet powerful enough for data-processing professionals. The examples in this section do not illustrate all the capabilities of the language, as the inventory database does not lend itself to describing all the features. (See the section on SQL in the Reference Manual for more tutorial examples.)

A SQL query consists of "clauses," each of which is preceded by a keyword. These keywords have special meaning to SQL, and so you cannot use them for database record or field names. (See the Reference Manual for a list of these reserved keywords.) Some clauses are required, and some are optional. The required clauses are:

select	some data (a list of field names)
from	some place (a list of record types)

The optional clauses are:

where	a condition (a true/false statement)
group by	some data (a list of field names)
having	a group condition (a true/false statement)
order by	some data (a list of field names)
into	a TRS-XENIX file

These examples are based on the expanded database design developed in the previous sections. We added some additional data -- customers, orders, and items on the orders -- so that queries yield more interesting results.

Select SQL as follows:

```
[entmenu]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                    Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: 4_
```

The screen clears, and the SQL prompt (sql>) appears.

Help Features

Unify SQL provides three levels of help: (1) general syntax; (2) specific keywords or keyword phrases; and (3) the valid record and field names for the database you are using.

To get the Level 1, type:

```
sql> help <ENTER>
```

The information below appears:

```
-- Structured Query Language (SQL)

SQL is a query language based on an English keyword syntax.
A query is composed of a series of keyword clauses.
The keywords that introduce each clause are as follows:
                select
                from
                where
                group by
                having
                order by
                into

Help about any keyword can be obtained by typing "help <keyword>".
The complete list of SQL keywords is as follows:
        and     asc     avg       between by      count
        desc    edit    end       fields  from    group
        having  help    in        into    is      lines
        max     min     not       or      order   records
        restart select  separator start   sum     unique
        unlock  where

sql>_
```

To get further information about a specific keyword (the second level of help), type **help** followed by the keyword. Thus, to find out more about the select clause, type **help select** as shown next.

```
sql> help select

-- select

The 'select' clause introduces every query.  It has the following form:

     select {field or expression} [{field or expression}, ...]
     from {record type} [{record type}, ...] /

Note that a 'from' clause is also required for a valid query.

The slash character ("/") must end every query.  If you don't enter a
slash at the end of your query, SQL doesn't know you are finished.

To see how to get a list of the valid record types and fields, type
'records' or 'fields' (a preceding 'help' is not required).

Suppose there were a record type named "emp" in the database.  You
could select all the fields in this record with the following query:

                    select * from emp /

sql>_
```

Now let's see the list of valid record types and fields that you can use in queries. Use the **records** keyword to see all the valid record types and the **fields** keyword to see the field descriptions for particular record types.

```
sql> records
manf      model     item      customer orders

sql> fields manf

NAME                                          TYPE     LENGTH

number                                        INTEGER  4
name                                          STRING   35
address                                       STRING   3Ø
city                                          STRING   2Ø
state                                         STRING   2
zip_code                                      LONG     5

sql> fields model

NAME                                          TYPE     LENGTH

model_num                                     LONG     7
manufacturer_num                              INTEGER  4
description                                   STRING   3Ø

sql> fields item

NAME                                          TYPE     LENGTH

serial_number                                 LONG     9
imodel_monum                                  LONG     7
imodel_momano                                 INTEGER  4
acquisition_date                              DATE     2
sale_price                                    AMOUNT   5
order_number                                  LONG     9
purchase_price                                AMOUNT   5
```

Note: When you enter a query, SQL requires that you use a field's long name. SQL also retrieves the long name when answering a query.

Selecting Records

The simplest SQL query selects all the fields from a particular record type. The asterisk (*) indicates that all fields are to be listed.

EXAMPLE: List the contents of the model record type.

```
sql> select *
sql> from model/
recognized query!

model_num|manufacturer_num|description
------------------------------------------------------------
     1ØØ1|             1ØØ|1/2" socket wrench
     1ØØ2|             1ØØ|3/4" socket wrench
     1ØØ3|             1ØØ|1/2" open end wrench
    55Ø71|             1Ø1|1/2" socket wrench
    55171|             1Ø1|1/2" box end wrench
    55271|             1Ø1|combination pliers
    55371|             1Ø1|needle nose pliers
     1ØØ1|             1Ø2|vise grips
     1ØØ2|             1Ø2|leather mallet
    61117|             1Ø3|3" Phillips screwdriver
    71117|             1Ø3|3" slotted screwdriver
    81117|             1Ø3|6" Phillips screwdriver
    91117|             1Ø3|6" slotted screwdriver
     1Ø11|             1Ø4|combination pliers
     1Ø12|             1Ø4|1/2" socket wrench
```

Besides listing all the fields, you can list only some fields. In addition, you can use the where clause to select a subset of records.

EXAMPLE: List the manufacturer number, name, and address for manufacturers whose ID number is greater than 1ØØ.

```
sql> select number, name, address
sql> from manf
sql> where number > 1ØØ /
recognized query!
```

number	name	address
1Ø1	Precision Tool Co.	26ØØ West 16th Street
1Ø2	A & H Industries, Inc.	2434 Evergreen Ave.
1Ø3	Grover Parts and Supplies	9462 Jackson Road
1Ø4	The Tool Depot	7562 Orange Dr.
1Ø5	BHP Ltd.	17385 Weatherby Rd.

If you make a mistake when typing a query, an error message indicates the nature of the error. For example, if in the previous query you typed the word "number" as "numfer" in the select clause, the following error message appears:

```
sql> select numfer, name, address
sql> from manf
sql> where number > 1ØØ /
Invalid field: numfer
```

You do not have to retype the query. SQL saves the last query you typed, so you can edit it. To get into the editor, type:

```
sql> edit
```

vi is the standard Unify editor. (See the Unify Reference Manual, "Appendix B.")

```
select numfer, name, address
from manf
where number > 1ØØ /
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/plØØnnnn" 3 lines, 59 characters
```

Now you can use the editor to make the required correction. In this case, change the "f" to a "b" in the word "numfer" by following these steps:

1. Place the cursor under "f" by pressing the <Space Bar> or <l>.

2. Type **r** for replace and then type **b**.

3. Press <ESC> to exit the replace mode.

Now write the file and quit the editor by typing:

:wq

You are now at the SQL prompt. To restart the current query, type **restart**. SQL executes the saved query just as if you had typed it at the terminal.

```
sql> restart
recognized query!
```

number	name	address
101	Precision Tool Co.	2600 West 16th Street
102	A & H Industries, Inc.	2434 Evergreen Ave.
103	Grover Parts and Supplies	9462 Jackson Road
104	The Tool Depot	7562 Orange Dr.
105	BHP Ltd.	17385 Weatherby Rd.

If you want to save this query, type **edit** to get back into the editor. Now write the temporary file to a file in your directory named "saved_query" by typing:

```
:w saved_query
```

```
select number, name, address
from manf
where number > 1ØØ /
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:q
```

Now exit the editor and return to SQL. When you want to use this query again, type **start saved_query**.

```
sql> start saved_query
recognized query!
```

number	name	address
1Ø1	Precision Tool Co.	26ØØ West 16th Street
1Ø2	A & H Industries, Inc.	2434 Evergreen Ave.
1Ø3	Grover Parts and Supplies	9462 Jackson Road
1Ø4	The Tool Depot	7562 Orange Dr.
1Ø5	BHP Ltd.	17385 Weatherby Rd.

The where clause has several special features that simplify queries. For example, you may specify that a number, date, or amount be between two values.

EXAMPLE: List the sale price of items that sell for between three and ten dollars.

```
sql> select sale_price
sql> from item
sql> where sale_price between 3.ØØ and 1Ø.ØØ/
recognized query!

     sale_price
---------------
           9.75
           9.75
           9.75
           6.89
           8.75
           3.19
           5.5Ø
           9.5Ø
```

Notice that the list contains some duplications. If you want to list only the unique prices, SQL lets you eliminate the duplicates.

EXAMPLE: List only unique sale prices for those items that sell for between three and ten dollars.

```
sql> select unique sale_price
sql> from item
sql> where sale_price between 3.ØØ and 1Ø.ØØ/
recognized query!

     sale_price
----------------
           3.19
           5.5Ø
           6.89
           8.75
           9.5Ø
           9.75
```

The where clause can also contain compound Boolean expressions composed of simple expressions connected with the **and** and **or** keywords. You can control precedence by enclosing the parts of the expression you want evaluated first within square brackets ([]).

EXAMPLE: List the serial number, manufacturer number, and sale price for items made either by manufacturer 1Ø1 or manufacturer 1Ø3 that sell for seven dollars or less.

```
sql> select serial_number, imodel_momano, sale_price
sql> from item
sql> where [imodel_momano = 1Ø1 or imodel_momano = 1Ø3] and
sql>       sale_price <= 7.ØØ /
recognized query!

serial_number|imodel_momano|     sale_price
-------------------------------------------
         1ØØ6|          1Ø1|           6.89
         1ØØ8|          1Ø3|           2.35
         1ØØ9|          1Ø3|           2.69
         1Ø1Ø|          1Ø3|           2.89
         1Ø11|          1Ø3|           3.19
```

Arithmetic Expressions

SQL lets you use standard arithmetic operators (+, -, *, and /) to calculate numeric values. You may use both constants and fields in arithmetic expressions, which are allowed on fields of all types except STRING and COMB. You may use arithmetic expressions wherever a simple field is allowed in the select, where, and having clauses.

EXAMPLE: List the model number, sale price, purchase price, and markup (sale price minus purchase price) for the items made by manufacturer number 1Ø3.

```
sql> select imodel_monum, sale_price, purchase_price,
sql>        sale_price-purchase_price
sql> from item
sql> where imodel_momano = 1Ø3/
recognized query!
```

imodel_monum	sale_price	purchase_price	sale_price-purchase_price
61117	2.35	2.2Ø	Ø.15
61117	2.69	2.ØØ	Ø.69
91117	2.89	3.ØØ	-Ø.11
91117	3.19	2.2Ø	Ø.99

Because DATE fields are stored in integer format, you can use them like numbers in calculations. To find the items added to inventory at least one week before a given date, you can use the following query.

EXAMPLE: List the serial number, acquisition date, and date one week after the acquisition date for those items acquired more than one week before 2/22/82.

```
sql> select serial_number, acquisition_date, acquisition_date + 7
sql> from item
sql> where acquisition_date + 7 < 2/22/82 /
recognized query!
```

serial_number	acquisition_date	acquisition_date+7
1ØØ7	Ø1/19/82	Ø1/26/82
1ØØ8	Ø1/15/82	Ø1/22/82
1Ø1Ø	Ø1/15/82	Ø1/22/82
1Ø12	Ø2/Ø8/82	Ø2/15/82
1Ø13	Ø2/Ø8/82	Ø2/15/82

If the arithmetic expression is complicated, you can use parentheses to control the order of computation. SQL calculates expressions within parentheses before those outside parentheses.

EXAMPLE: List the serial number, sale price, purchase price, and the formula (purchase price + 2) * Ø.75 for those items where the sale price is greater than the formula price.

```
sql> select serial_number, sale_price, purchase_price,
sql>        (purchase_price + 2) * Ø.75
sql> from item
sql> where sale_price > (purchase_price + 2) * Ø.75 /
recognized query!
```

serial_number	sale_price	purchase_price	(purchase_price+2)*Ø.75
1ØØ1	9.75	7.25	6.94
1ØØ2	9.75	7.35	7.Ø1
1ØØ3	9.75	7.25	6.94
1ØØ4	1Ø.25	8.9Ø	8.18
1ØØ5	1Ø.75	9.25	8.44
1ØØ6	6.89	6.ØØ	6.ØØ
1ØØ7	8.75	5.25	5.44
1Ø11	3.19	2.2Ø	3.15
1Ø13	9.5Ø	9.23	8.42
1234	15.ØØ	13.23	11.42

Ordering Output

The **order by** clause lets you explicitly specify the sequence of the rows that result from a query. The default sort sequence is ascending, with STRING fields sorted in alphabetic order from A to Z.

Note: Numbers always sort before letters.

EXAMPLE: List all models, sorted by description.

```
sql> select *
sql> from model
sql> order by description /
recognized query!
```

model_num	manufacturer_num	description
55171	101	1/2" box end wrench
1003	100	1/2" open end wrench
1001	100	1/2" socket wrench
1012	104	1/2" socket wrench
55071	101	1/2" socket wrench
61117	103	3" Phillips screwdriver
71117	103	3" slotted screwdriver
1002	100	3/4" socket wrench
81117	103	6" Phillips screwdriver
91117	103	6" slotted screwdriver
1011	104	combination pliers
55271	101	combination pliers
1002	102	leather mallet
55371	101	needle nose pliers
1001	102	vise grips

You can sort the results of a query by more than one field, and you can specify that some fields sort in ascending order, others in descending order.

EXAMPLE: List the acquisition date and sale price for items acquired between 1/1/82 and 3/1/82. Sort them by descending date and ascending price within date.

```
sql> select asquisition_date, sale_price
sql> from item
sql> where acquisition_date between 1/1/82 and 3/1/82
sql> order by acquisition_date desc, sale_price asc /
recognized query!
```

acquisition_date	sale_price
Ø2/25/82	1Ø.75
Ø2/23/82	1Ø.25
Ø2/23/82	15.ØØ
Ø2/15/82	2.69
Ø2/15/82	3.19
Ø2/15/82	6.89
Ø2/15/82	9.75
Ø2/15/82	9.75
Ø2/15/82	9.75
Ø2/Ø8/82	5.5Ø
Ø2/Ø8/82	9.5Ø
Ø1/19/82	8.75
Ø1/15/82	2.35
Ø1/15/82	2.89

Aggregate Functions

SQL provides five built-in functions that allow you to calculate aggregate values in a query result. The functions are count, sum, minimum, maximum, and average. You can use aggregate functions in the select and having clauses of a query.

EXAMPLE: Calculate the average sale price for all items in inventory.

```
sql> select avg(sale_price)
sql> from item /
recognized query!

avg(sale_price)
---------------
           7.64
```

You can calculate several aggregate functions in the same query. The next query lists both the highest and lowest price for all the items.

EXAMPLE: List the maximum and the minimum sale price for any item in inventory.

```
sql> select max(sale_price), min(sale_price)
sql> from item /
recognized query!

max(sale_price)|min(sale_price)
--------------------------------
          15.ØØ|            2.35
```

The count function tells you how many records satisfied the query. The notation for the function is always the same, and it was chosen to make it consistent with the other aggregate functions. (In other words, a keyword followed by an item enclosed in parentheses. You can't "count" anything but "*".)

EXAMPLE: Count the total number of manufacturers.

```
sql> select count(*)
sql> from manf /
recognized query!

count(*)
----------
         6
```

Grouping Records

The **group by** clause lets you compute aggregate functions on groups of records that have common characteristics. Thus, using a group by clause without an aggregate function has no meaning.

The effect of a group by is to sort the selected rows by the indicated fields and then perform the aggregate functions at each level break. The output sorts also.

EXAMPLE: List the manufacturer number and average sale price for the items made by each manufacturer.

```
sql> select imodel_momano, avg(sale_price)
sql> from item
sql> group by imodel_momano/
recognized query!

imodel_momano|avg(sale_price)
-------------------------------
          1ØØ|            11.Ø6
          1Ø1|             9.3Ø
          1Ø2|             8.75
          1Ø3|             2.78
          1Ø4|             7.5Ø
```

Since each line of the query output represents a computed value from a group of records, you can only list fields that have common values for the whole group. Thus, listing the serial numbers for the items is not possible in this query.

You can use more than one field in a group by clause, which creates more level breaks in the query result. In addition, you can specify a where clause that first qualifies all the records to be included in the calculation.

EXAMPLE: List the manufacturer number, model number, and number of items for all items that sell for more than five dollars, broken down by manufacturer and model number.

```
sql> select imodel_momano, imodel_monum, count(*)
sql> from item
sql> where sale_price > 5.ØØ
sql> group by imodel_momano, imodel_monum/
recognized query!

imodel_momano|imodel_monum| count(*)
-------------------------------------
          1ØØ|        1Ø1|        3
          1ØØ|        1Ø2|        1
          1Ø1|       55Ø71|        2
          1Ø1|       55271|        1
          1Ø2|        1ØØ2|        1
          1Ø4|        1Ø11|        1
          1Ø4|        1Ø12|        1
```

Nested Queries

Nested queries allow you to answer a whole new set of questions that you cannot answer using the capabilities of SQL presented so far. Nesting lets you use the results of one query as input to another. In other words, you can use the results of one question to answer another. For example, if you want to find the serial numbers of items in inventory that sell for the maximum price, using the capabilities presented so far, you can find the maximum price for any item as follows:

```
select max(sale_price)
from item /
```

Or you can find the items that sell for a specific amount, such as all those that sell for ten dollars:

```
select serial_number
   from item
where sale_price = 1Ø.ØØ /
```

But the only way to find out which are the most expensive items is to perform the first query and then use the resulting maximum sale price as the constant value in the second query. Nesting allows you to do just that, automatically.

EXAMPLE: List the serial number, acquisition date, and sale price for the items with the maximum selling price.

```
sql> select serial_number, acquisition_date, sale_price
sql> from item
sql> where sale_price = select max(sale_price)
sql>                      from item /
recognized query!

serial_number|acquisition_date|    sale_price
-----------------------------------------------
        1234|        Ø2/23/82|          15.ØØ
```

SQL evaluates the query in two steps. First, SQL executes the inner query (select max (sale_price) from item) to get the maximum sale price. Then, SQL performs the outer query, using the results of the inner query.

You can nest queries to any number of levels, not just one. For example, to find the items that have the second highest sale price, first find the items that have the maximum price and then exclude these items from another query that finds the maximum price among the remaining items.

A final query finds the items that sell for this price.

EXAMPLE: List the serial number, acquisition date, and sale price for the items that sell for the second highest price.

```
sql> select serial_number, acquisition_date, sale_price
sql> from item
sql> where sale_price =
sql>       select max(sale_price)
sql>       from item
sql>       where serial_number ^=
sql>             select serial_number
sql>             from item
sql>             where sale_price =
sql>                   select max(sale_price)
sql>                   from item /
recognized query!

serial_number|acquisition_date|    sale_price
-----------------------------------------------
         1ØØ5|        Ø2/25/82|          1Ø.75
```

Note: ^= in the SQL statement indicates "not equal."

To understand this query, start from the innermost query, which finds the maximum sale price for all items.

```
select max(sale_price) from item /
```

The result of this query is 15.ØØ. The next query finds the serial numbers of items that sell for that price. Thus this query becomes

```
select serial_number from item
where sale_price = 15.ØØ /
```

The result of this query is the serial number 1234. You then use this serial number in the next query, which finds the maximum sale price of all items except for this highest price item. If you substitute this value, the query reads as follows:

```
select max(sale_price)
```

```
from item
where serial_number ^= 1234 /
```

The result of this query is 1Ø.75. The final query finds the serial number, acquisition date, and sale price of the items that sell for this price, which is the second highest price. Thus this query becomes:

```
select serial_number, acquisition_date, sale_price
from item
where sale_price = 1Ø.75 /
```

Having Clause

The **having** clause lets you select some of the groups formed by a previous group by clause and reject others. The selection is based on the results of an aggregate function contained within the having clause. This gives you a capability equivalent to using an aggregate function in a where clause, which is not allowed. For example, you can use the having clause to select the models whose average sale price exceeds six dollars.

EXAMPLE: List the model number and average sale price for models having an average sale price of more than six dollars.

```
sql> select imodel_monum, avg (sale_price)
sql> from item
sql> group by imodel_monum
sql> having avg(sale_price) > 6.ØØ /
recognized query!

imodel_monum|avg(sale_price)
------------------------------
        1ØØ1|             9.75
        1ØØ2|            11.88
        1Ø12|             9.5Ø
       55Ø71|            1Ø.5Ø
       55271|             6.89
```

If, for example, you have found the models selling for an average price of more that six dollars, you can now list the individual items. Modify the previous query slightly to return only the model number and then nest it within another query that lists the other information.

EXAMPLE: List the serial number, model number, sale price, and acquisition date for items of models that sell for more than six dollars on the average. Order the results by model number.

```
sql> select serial_number, imodel_monum, sale_price, acquisition_date
sql> from item
sql> where imodel_monum =
sql>       select imodel_monum
sql>       from item
sql>       group by imodel_monum
sql>       having avg(sale_price) > 6.ØØ;
sql> order by imodel_monum /
recognized query!
```

serial_number	imodel_monum	sale_price	acquisition_date
1ØØ1	1ØØ1	9.75	Ø2/15/82
1ØØ2	1ØØ1	9.75	Ø2/15/82
1ØØ3	1ØØ1	9.75	Ø2/15/82
1ØØ7	1ØØ2	8.75	Ø1/19/82
1234	1ØØ2	15.ØØ	Ø2/23/82
1Ø13	1Ø12	9.5Ø	Ø2/Ø8/82
1ØØ4	55Ø71	1Ø.25	Ø2/23/82
1ØØ5	55Ø71	1Ø.75	Ø2/25/82
1ØØ6	55271	6.89	Ø2/15/82

The inner query in this example ends with a semicolon. The entire query is valid with or without the semicolon, but in this case, it tells SQL that the order by clause goes with the outer query. Without the semicolon, the order by clause goes with the inner query, and the outer query result displays in system-determined order.

You can also use the having clause in conjunction with a where clause. SQL processes the query in the following steps: First, it applies the where clause to select qualifying records; then it forms the groups indicated by the group by clause; last, it applies the having clause to select qualifying groups.

EXAMPLE: List the model number and item count for models that have at least two items that cost more than five dollars.

```
sql> select imodel_monum, count(*)
sql> from item
sql> where sale_price > 5.ØØ
sql> group by imodel_monum
sql> having count(*) >= 2/
recognized query!

imodel_monum| count(*)
------------------------
        1ØØ1|        3
        1ØØ2|        2
       55Ø71|        2
```

Multifile Queries

So far, all queries have involved only a single record type. However, a SQL statement may list fields from any number of record types in a single query. Queries that list fields from several record types are called "join" queries, because they combine, or join, the different record types. You list the different record types to be involved in the query in the from clause, in any convenient order. SQL then determines the most efficient method of selection and qualification.

EXAMPLE: List the model number, model description, and manufacturer name for all the models in inventory.

```
sql> select model_num, description, name
sql> from manf, model
sql> where number = manufacturer_num/
recognized query!
```

model_num	description	name
1ØØ1	1/2" socket wrench	RH Smith Manufacturing
1ØØ2	3/4" socket wrench	RH Smith Manufacturing
1ØØ3	1/2" open end wrench	RH Smith Manufacturing
55Ø71	1/2" socket wrench	Precision Tool Co.
55171	1/2" box end wrench	Precision Tool Co.
55271	combination pliers	Precision Tool Co.
55371	needle nose pliers	Precision Tool Co.
1ØØ1	vise grips	A & H Industries, Inc.
1ØØ2	leather mallet	A & H Industries, Inc.
61117	3" Phillips screwdriver	Grover Parts and Supplies
71117	3" slotted screwdriver	Grover Parts and Supplies
81117	6" Phillips screwdriver	Grover Parts and Supplies
91117	6" slotted screwdriver	Grover Parts and Supplies
1Ø11	combination pliers	The Tool Depot
1Ø12	1/2" socket wrench	The Tool Depot

The join is indicated by the where clause, which relates one field from the model record type, the manufacturer number, to the equivalent field in the manufacturer record. In this case, an explicit relationship (defined in the REF column of the model record schema) exists between the manufacturer and model record types that optimize this join. However, SQL does not require that you define any particular access method; it uses whatever is available. The only difference is in the speed of the operation. (For a full discussion, see the Reference Manual.)

You may join more than two record types at the same time using SQL. List the record types in the from clause and indicate how they are to be related in the where clause.

EXAMPLE: For each item, list its model description, acquisition date, and name of the manufacturer, sorted by acquisition date within description.

```
sql> select description, acquisition_date, name
sql> from item, model, manf
sql> where model_num = imodel_monum and
sql>       manufacturer_num = imodel_momano and
sql>       manufacturer_num = number
sql> order by description, acquisition_date /
recognized query!
```

description	acquisition_date	name
1/2" socket wrench	Ø2/Ø8/82	The Tool Depot
1/2" socket wrench	Ø2/15/82	RH Smith Manufacturing
1/2" socket wrench	Ø2/15/82	RH Smith Manufacturing
1/2" socket wrench	Ø2/15/82	RH Smith Manufacturing
1/2" socket wrench	Ø2/23/82	Precision Tool Co.
1/2" socket wrench	Ø2/25/82	Precision Tool Co.
3" Phillips screwdriver	Ø1/15/82	Grover Parts and Supplies
3" Phillips screwdriver	Ø2/15/82	Grover Parts and Supplies
3/4" socket wrench	Ø2/23/82	RH Smith Manufacturing
6" slotted screwdriver	Ø1/15/82	Grover Parts and Supplies
6" slotted screwdriver	Ø2/15/82	Grover Parts and Supplies
combination pliers	Ø2/Ø8/82	The Tool Depot
combination pliers	Ø2/15/82	Precision Tool Co.
leather mallet	Ø1/19/82	A & H Industries, Inc.

The preceding examples illustrate only some of SQL's capabilities. You can use many other techniques to answer more complicated questions about the database. See the Reference Manual for additional examples and ideas about how to use SQL.

Now type <END> to return to the menu.

```
sql> end
```

Chapter 11
USING THE LISTING PROCESSOR

The Listing Processor is a selection and formatting language. With it, you can quickly and easily produce simple file listings with totals, subtotals, and column headings. Its capabilities complement those of SQL by providing a simple language that assigns most page-formatting details by default.

Use the following clauses to build your Listing Processor queries. Notice that some of these clauses are required and some are optional.

```
          * select  record type
                 where  a condition   end

[optional]       into  a TRS-XENIX file

          * report  record type
[either]         -list  some data (a list of field names)  end
                 -go

[or]             -print  some data (a list of field names)  end

[optional]       -sort  some data
[optional]       -total  some data
```

Differences between the Listing Processor and SQL

Many of the clauses you use in the Listing Processor are similar to those used in SQL. There are some specific functional and syntactical differences noted below:

- The SELECT statement in the Listing Processor specifies only the record type, instead of a list of fields from a record as is the case in SQL.

- When using the Listing Processor and specifying fields for reporting, you must use the short field names, instead of the long names that are required in SQL.

- Functions such as SORT and TOTAL are Reporting options in the Listing Processor. In SQL, these functions (ORDER BY and SUM) are options in the SELECT statement.

- In the Listing Processor, the "not equal" operator is indicated as !=. In SQL, not equal is shown as ^=.

- When using the Listing Processor to display a column resulting from an arithmetic expression, no column heading is used unless specified. In SQL, the resulting column uses the arithmetic expression as the column heading.

The Listing Processor and SQL share many overlapping query capabilities, but both methods perform very different functions, so choose the method appropriate for your task. Use the Listing Processor's powerful reporting capabilities to customize your query output, or select SQL's powerful relational capabilities to perform inpromptu or more complex queries.

The Listing Processor examples in this chapter are based on the expanded database design developed in the previous chapters. Select the Listing Processor as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: lst_
```

(The Listing Processor can also be selected as an option from the System Menu.)

The screen clears, and the asterisk (*) prompt appears. The asterisk prompt is for selecting records; the dash (-) prompt is for formatting the records for printing. The following examples show how this works.

List the Database Contents

The following examples list the contents of the temporary files created as a result of the **select** statements. Since "1" is always true, the select statements in this section retrieve all records from the record type specified. Unless you specify otherwise, records are selected into a temporary file that has the same name as the record type. You can then produce reports from this temporary file.

If there is already a temporary file with the same name as the record type, selection begins with the records in that temporary file. Therefore, to select records from the entire record type, you must remove the temporary file each time the reporting is finished. You can rename temporary files to save them for future use. Temporary files are "virtual" in nature and take up little actual disk space.

EXAMPLE: List all the records in the manufacturer record type.

```
* select manf where 1 end
6 records
* report manf
- list mano,mname,madd end
- go
```

mano	mname	madd
1ØØ	RH Smith Manufacturing	523 Galveston St.
1Ø1	Precision Tool Co.	26ØØ West 16th Street
1Ø2	A & H Industries, Inc.	2434 Evergreen Ave.
1Ø3	Grover Parts and Supplies	9462 Jackson Road
1Ø4	The Tool Depot	7562 Orange Dr.
1Ø5	BHP Ltd.	17385 Weatherby Rd.

```
- end
* remove manf
manf removed
```

EXAMPLE: List all the records in the model record type.

```
* select model where 1 end
15 records
* report model
- list monum,momano,mdes end
- go
     monum     momano mdes
-------------------------------------------------------
      1ØØ1        1ØØ 1/2" socket wrench
      1ØØ2        1ØØ 3/4" socket wrench
      1ØØ3        1ØØ 1/2" open end wrench
     55Ø71        1Ø1 1/2" socket wrench
     55171        1Ø1 1/2" box end wrench
     55271        1Ø1 combination pliers
     55371        1Ø1 needle nose pliers
      1ØØ1        1Ø2 vise grips
      1ØØ2        1Ø2 leather mallet
     61117        1Ø3 3" Phillips screwdriver
     71117        1Ø3 3" slotted screwdriver
     81117        1Ø3 6" Phillips screwdriver
     91117        1Ø3 6" slotted screwdriver
      1Ø11        1Ø4 combination pliers
      1Ø12        1Ø4 1/2" socket wrench
-------------------------------------------------------

- end
* remove model
model removed
```

EXAMPLE: List all the records in the inventory item record type.

```
* select item where 1 end
14 records
* report item
- list sno,imodel_momano,imodel_monum,iad,isal,iorder,ipamt end
- go
```

sno	imodel_momano	imodel_monum	iad	isal	iorder	ipamt
1ØØ1	1ØØ	1ØØ1	Ø2/15/82	9.75	1	7.25
1ØØ2	1ØØ	1ØØ1	Ø2/15/82	9.75	3	7.35
1ØØ3	1ØØ	1ØØ1	Ø2/15/82	9.75	1	7.25
1ØØ4	1Ø1	55Ø71	Ø2/23/82	1Ø.25	1	8.9Ø
1ØØ5	1Ø1	55Ø71	Ø2/25/82	1Ø.75	3	9.25
1ØØ6	1Ø1	55271	Ø2/15/82	6.89	1	6.ØØ
1ØØ7	1Ø2	1ØØ2	Ø1/19/82	8.75	1	5.25
1ØØ8	1Ø3	61117	Ø1/15/82	2.35	2	2.2Ø
1ØØ9	1Ø3	61117	Ø2/15/82	2.69	2	2.ØØ
1Ø1Ø	1Ø3	91117	Ø1/15/82	2.89	1	3.ØØ
1Ø11	1Ø3	91117	Ø2/15/82	3.19	1	2.2Ø
1Ø12	1Ø4	1Ø11	Ø2/Ø8/82	5.5Ø	1	5.75
1Ø13	1Ø4	1Ø12	Ø2/Ø8/82	9.5Ø	2	9.23
1234	1ØØ	1ØØ2	Ø2/23/82	15.ØØ	2	13.23

```
- end
* remove item
item removed
```

The **under** syntax lets you list columns under one another. The headings are listed in the same manner as the columns.

EXAMPLE: Select all the records from the customer record type. List the address fields under each other so the report fits inside 8Ø columns.

```
* select customer where 1 end
3 records
* report customer
- list cnum,cname,caddr end
- list ccity under caddr end
- list cstate,czip,cphone under ccity end
- go
   cnum cname                          caddr
                                       ccity
                                       cstate      czip cphone
-------------------------------------------------------------------
      1 Smith & Sons Harware           1234 State Street
                                       Wheatville
                                       CA          95817 916-924-ØØ75
      2 Creative Manufacturing         9124 Industrial Blvd.
                                       Redding
                                       CA          921Ø2 916-91Ø-9999
      3 Reliable Construction Co.      2113 Folsom Blvd.
                                       Sacramento
                                       CA          92343 916-945-3434
-------------------------------------------------------------------

- end
* remove customer
customer removed
```

EXAMPLE: List all the records in the orders record type.

```
* select orders where 1 end
3 records
* report orders
- list onum,odate,ocust end
- go
      onum     odate      ocust
-------------------------------
         1 Ø4/Ø2/82          1
         2 Ø4/Ø2/82          3
         3 Ø6/28/82          2
-------------------------------

- end
* remove orders
orders removed
```

Selecting from a File

By specifying selection criteria, you can determine which records to place in the temporary file. The list command specifies which fields are output.

EXAMPLE: List the name of the manufacturer whose number is 1Ø1.

```
* select manf where mano = 1Ø1 end
1 records
* report manf
- list mname end
- go
mname
-------------------------------------
Precision Tool Co.
_____________________________________


- end
* remove manf
manf removed
```

EXAMPLE: List the model number, manufacturer number, and description for all models whose description begins with the string "combination pliers."

```
* select model where mdes = "combination pliers*" end
2 records
* report model
- list monum,momano,mdes end
- go
     monum     momano mdes
-------------------------------------------------
     55271        1Ø1 combination pliers
      1Ø11        1Ø4 combination pliers
_________________________________________________

- end
* remove model
model removed
```

You can use the **where** clause to compare two fields.

EXAMPLE: Select inventory items that sell for a price higher than their cost. For these records list the serial number, manufacturer number, model number, purchase price, and sale price.

```
* select item where ipamt > isal end
2 records
* report item
- list sno,imodel_momano,imodel_monum,ipamt,isal end
- go
```

sno	imodel_momano	imodel_monum	ipamt	isal
1Ø1Ø	1Ø3	91117	3.ØØ	2.89
1Ø12	1Ø4	1Ø11	5.75	5.5Ø

```
- end
* remove item
item removed
```

You can combine conditions with Boolean operators to yield complex queries.

EXAMPLE: Select models produced by manufacturer 1Ø1 that have the description "combination pliers." For these, list the model number.

```
* select model where mdes = "combination pliers" and
momano = 1Ø1 end
1 records
* report model
- list monum end
- go
```

monum
55271

```
- end
* remove model
model removed
```

EXAMPLE: Select items that have a sale price between $2.5Ø and $9.5Ø. For these, list the serial number, manufacturer number, model number, and sale price.

```
* select item where isal > 2.5Ø and isal < 9.5Ø end
6 records
* report item
- list sno,imodel_momano,imodel_monum,isal end
- go
```

sno	imodel_momano	imodel_monum	isal
1ØØ6	1Ø1	55271	6.89
1ØØ7	1Ø2	1ØØ2	8.75
1ØØ9	1Ø3	61117	2.69
1Ø1Ø	1Ø3	91117	2.89
1Ø11	1Ø3	91117	3.19
1Ø12	1Ø4	1Ø11	5.5Ø

```
- end
* remove item
item removed
```

EXAMPLE: Select items that entered inventory before 2/1/82 or that sell for more than $1Ø.ØØ. For these, list the serial number, add date, and sale price.

```
* select item where iad < 2/1/82 or isal > 1Ø.ØØ end
6 records
* report item
- list sno,iad,isal end
- go
```

sno	iad	isal
1ØØ4	Ø2/23/82	1Ø.25
1ØØ5	Ø2/25/82	1Ø.75
1ØØ7	Ø1/19/82	8.75
1ØØ8	Ø1/15/82	2.35
1Ø1Ø	Ø1/15/82	2.89
1234	Ø2/23/82	15.ØØ

```
- end
* remove item
item removed
```

Partial matching is allowed with string fields. The question mark (?) matches any single character. The asterisk (*) matches any number of characters. The three dots (...) syntax matches single characters in a given set.

? - The "wild character." The question mark matches any single character. For example, if you want to find all the Smiths but do not know if they are spelled "Smith" or "Smyth," you can use "Sm?th."

* - The "wild string." The asterisk matches any string of characters of any length, including zero length strings (also called "null strings").

[...] - The three dots stand for a set of characters that define a "character class." The character class matches any single character that is a member of the class within the brackets. You may specify ranges of characters by separating two characters with a dash (-). For example, you can represent all upper-case letters by the class

[ABCDEFGHIJKLMNOPQRSTUVWXYZ]

or somewhat more conveniently as [A-Z]. You can represent all letters, upper- and lower-case together, as [a-zA-Z]. You can construct other classes similarly.

EXAMPLE: Select models produced by manufacturer 1Ø3 whose description does not contain the string "slotted screwdriver." For these, list the number and description.

```
* select model where mdes != "*slotted screwdriver*"
and momano = 1Ø3 end
2 records
* report model
- list monum,mdes end
- go
     monum mdes
-----------------------------------------
     61117 3" Phillips screwdriver
     81117 6" Phillips screwdriver
-----------------------------------------

- end
* remove model
model removed
```

EXAMPLE: Select 1- through 3-inch screwdrivers from the model record. For these, list the manufacturer number, model number, and description.

```
* select model where mdes = "[1-3]*screwdriver*" end
2 records
* report model
- list momano,monum,mdes end
- go
momano      monum mdes
-------------------------------------------------------
1Ø3      61117 3" Phillips screwdriver
1Ø3      71117 3" slotted screwdriver
-------------------------------------------------------

- end
* remove model
model removed
```

Ordering Output

You can order a report by any field, combination of fields, or expression by using the **sort** command. Fields are listed in order of significance.

EXAMPLE: List the number, name, and address for manufacturers whose number is greater than 1Ø2. Order the output by name.

```
* select manf where mano > 1Ø2 end
3 records
* report manf
- sort mname end
- list mano,mname,madd end
- go
```

mano	mname	madd
1Ø5	BHP Ltd.	17385 Weatherby Rd.
1Ø3	Grover Parts and Supplies	9462 Jackson Road
1Ø4	The Tool Depot	7562 Orange Dr.

```
- end
* remove manf
manf removed
```

EXAMPLE: Select models whose manufacturer number is greater than 1Ø2. For these, list the description, manufacturer number, and model number ordered by manufacturer number within description.

```
* select model where momano > 1Ø2 end
6 records
* report model
- sort mdes,momano end
- list mdes,momano,monum end
- go
```

mdes	momano	monum
1/2" socket wrench	1Ø4	1Ø12
3" Phillips screwdriver	1Ø3	61117
3" slotted screwdriver	1Ø3	71117
6" Phillips screwdriver	1Ø3	81117
6" slotted screwdriver	1Ø3	91117
combination pliers	1Ø4	1Ø11

You can reverse the sort order by using the **sort reverse** syntax.

EXAMPLE: Using the results of the previous selection, list the same fields in reverse sequence.

```
- sort reverse mdes end
- sort momano end
- list mdes,momano,monum end
- go
```

mdes	momano	monum
combination pliers	1Ø4	1Ø11
6" Phillips screwdriver	1Ø3	81117
6" slotted screwdriver	1Ø3	91117
3" Phillips screwdriver	1Ø3	61117
3" slotted screwdriver	1Ø3	71117
1/2" socket wrench	1Ø4	1Ø12

```
- end
* remove model
model removed
```

Listing Items Uniquely

You may specify that an item be listed uniquely. This means that a particular result appears only once in the output.

EXAMPLE: List all model descriptions in the model record type.

```
* select model where 1 end
15 records
* report model
- sort uniquely mdes end
- list mdes end
- go
mdes
--------------------------------
1/2" box end wrench
1/2" open end wrench
1/2" socket wrench
3" Phillips screwdriver
3" slotted screwdriver
3/4" socket wrench
6" Phillips screwdriver
6" slotted screwdriver
combination pliers
leather mallet
needle nose pliers
vise grips
--------------------------------

- end
* remove model
model removed
```

EXAMPLE: List all the manufacturer/model combinations that were added to inventory on 2/15/82.

```
* select item where iad = 2/15/82 end
6 records
* report item
- sort uniquely imodel_momano,imodel_monum end
- list imodel_momano,imodel_monum end
- go
imodel_momano imodel_monum
--------------------------
       1ØØ           1ØØ1
       1Ø1          55271
       1Ø3          61117
       1Ø3          91117

- end
* remove item
item removed
```

Arithmetic Expressions

You can use arithmetic expressions throughout queries and reports. The following operators are understood:

+	addition
-	subtraction
*	multiplication
/	division

If a column of a report from the listing processor is the result of an arithmetic expression, then by default it has no heading. Later we show how to specify a heading.

EXAMPLE: Select items produced by manufacturer 1Ø3. For these, list manufacturer number, model number, sale price, purchase price, and markup (sale price less purchase price).

```
* select item where imodel_momano = 1Ø3 end
4 records
* report item
- list imodel_momano,imodel_monum,isal,ipamt, isal - ipamt end
- go
```

imodel_momano	imodel_monum	isal	ipamt	
1Ø3	61117	2.35	2.2Ø	Ø.15
1Ø3	61117	2.69	2.ØØ	Ø.69
1Ø3	91117	2.89	3.ØØ	-Ø.11
1Ø3	91117	3.19	2.2Ø	Ø.99

```
- end
* remove item
item removed
```

Since dates are stored internally in Julian format, you can add or subtract integer expressions and get meaningful results. The result is a date which is that many days after or before the original date. The subtraction of two dates yields the number of days between them.

EXAMPLE: Select items that were added to inventory more than seven days before 2/22/82. For these, list serial number, add date, and the date seven days after the add date.

```
* select item where iad + 7 < 2/22/82 end
5 records
* report item
- list sno, iad, iad + 7 end
- go
       sno       iad
-----------------------------
      1ØØ7 Ø1/19/82 Ø1/26/82
      1ØØ8 Ø1/15/82 Ø1/22/82
      1Ø1Ø Ø1/15/82 Ø1/22/82
      1Ø12 Ø2/Ø8/82 Ø2/15/82
      1Ø13 Ø2/Ø8/82 Ø2/15/82
-----------------------------

- end
* remove item
item removed
```

Parentheses establish precedence within arithmetic expressions.

EXAMPLE: Select items for which the sale price is greater than a formula based on the purchase price. For these list the serial number, sale price, purchase price, and the results of the formula.

```
* select item where isal > (ipamt + 2) * Ø.75 end
1Ø records
* report item
- list sno,isal,ipamt,(ipamt + 2) * Ø.75 end
- go
```

sno	isal	ipamt	
1ØØ1	9.75	7.25	6.94
1ØØ2	9.75	7.35	7.Ø1
1ØØ3	9.75	7.25	6.94
1ØØ4	1Ø.25	8.9Ø	8.18
1ØØ5	1Ø.75	9.25	8.44
1ØØ6	6.89	6.ØØ	6.ØØ
1ØØ7	8.75	5.25	5.44
1Ø11	3.19	2.2Ø	3.15
1Ø13	9.5Ø	9.23	8.42
1234	15.ØØ	13.23	11.42

```
- end
* remove item
item removed
```

Specifying Column Headings

You can specify report column headings by placing a string, which serves as the column heading, in front of the field or expression to be printed. If the column heading contains spaces, the character string must be enclosed in quotes (" "). To split a column heading onto multiple lines, insert a slash (/) at the point(s) of division.

EXAMPLE: For all items, list the serial number, add date, and markup. Give the columns appropriate headings.

```
* select item where 1 end
14 records
* report item
- list SERIAL/NUMBER sno, "ADD DATE" iad, MARKUP isal - ipamt end
- go
```

SERIAL NUMBER	ADD DATE	MARKUP
1ØØ1	Ø2/15/82	2.5Ø
1ØØ2	Ø2/15/82	2.4Ø
1ØØ3	Ø2/15/82	2.5Ø
1ØØ4	Ø2/23/82	1.35
1ØØ5	Ø2/25/82	1.5Ø
1ØØ6	Ø2/15/82	Ø.89
1ØØ7	Ø1/19/82	3.5Ø
1ØØ8	Ø1/15/82	Ø.15
1ØØ9	Ø2/15/82	Ø.69
1Ø1Ø	Ø1/15/82	-Ø.11
1Ø11	Ø2/15/82	Ø.99
1Ø12	Ø2/Ø8/82	-Ø.25
1Ø13	Ø2/Ø8/82	Ø.27
1234	Ø2/23/82	1.77

Once you establish a column heading, as in the sort command that follows, you need not respecify its contents. In fact, the following command results in a syntax error if iad is included after "ADD DATE" in the list command.

EXAMPLE: Sort the previous report by add date.

```
- sort "ADD DATE" iad end
- list "ADD DATE",SERIAL/NUMBER sno,MARKUP isal - ipamt end
- go
```

ADD DATE	SERIAL NUMBER	MARKUP
Ø1/15/82	1ØØ8	Ø.15
Ø1/15/82	1Ø1Ø	-Ø.11
Ø1/19/82	1ØØ7	3.5Ø
Ø2/Ø8/82	1Ø12	-Ø.25
Ø2/Ø8/82	1Ø13	Ø.27
Ø2/15/82	1ØØ1	2.5Ø
Ø2/15/82	1ØØ2	2.4Ø
Ø2/15/82	1ØØ3	2.5Ø
Ø2/15/82	1ØØ6	Ø.89
Ø2/15/82	1ØØ9	Ø.69
Ø2/15/82	1Ø11	Ø.99
Ø2/23/82	1ØØ4	1.35
Ø2/23/82	1234	1.77
Ø2/25/82	1ØØ5	1.5Ø

EXAMPLE: For all items in inventory, list the add date, serial number, and percentage markup. (The previous query has already selected all items in inventory.) Sort the output by percentage markup and give the columns appropriate headings.

```
- sort "PCT. MARKUP" ((isal - ipamt) * 1ØØ.Ø) / isal end
- list "ADD DATE" iad,SERIAL/NUMBER sno, "PCT. MARKUP" end
- go
```

ADD DATE	SERIAL NUMBER	PCT. MARKUP
Ø2/Ø8/82	1Ø12	-4.55
Ø1/15/82	1Ø1Ø	-3.81
Ø2/Ø8/82	1Ø13	2.84
Ø1/15/82	1ØØ8	6.38
Ø2/23/82	1234	11.8Ø
Ø2/15/82	1ØØ6	12.92
Ø2/23/82	1ØØ4	13.17
Ø2/25/82	1ØØ5	13.95
Ø2/15/82	1ØØ2	24.62
Ø2/15/82	1ØØ1	25.64
Ø2/15/82	1ØØ3	25.64
Ø2/15/82	1ØØ9	25.65
Ø2/15/82	1Ø11	31.Ø3
Ø1/19/82	1ØØ7	4Ø.ØØ

Grouping and Totaling

You can sort the lines on a report by any combination of fields and produce totals, subtotals, and sums of the sorted items.

EXAMPLE: Using the previously selected item file, list the total purchases by manufacturer for all items in inventory.

```
- sort MANF imodel_momano end
- total PURCHASES ipamt by MANF end
- go
      MANF  PURCHASES
----------------------
       1ØØ      35.Ø8
       1Ø1      24.15
       1Ø2       5.25
       1Ø3       9.4Ø
       1Ø4      14.98
```

EXAMPLE: Compute the markup for every item in inventory. Total the results by model within manufacturer and produce a grand total.

```
- sort MANF imodel_momano, MODEL imodel_monum end
- total MARKUP isal - ipamt by MANF,MODEL end
- total MARKUP end
- go
      MANF      MODEL      MARKUP
---------------------------------
                 1ØØ1        7.4Ø
                 1ØØ2        1.77
       1ØØ                   9.17
                55Ø71        2.85
                55271        Ø.89
       1Ø1                   3.74
                 1ØØ2        3.5Ø
       1Ø2                   3.5Ø
                61117        Ø.84
                91117        Ø.88
       1Ø3                   1.72
                 1Ø11       -Ø.25
                 1Ø12        Ø.27
       1Ø4                   Ø.Ø2

                            18.15
```

You can total records into categories with the following type of syntax. This is possible because the result of a Boolean expression is either one if true or zero if false.

EXAMPLE: Categorize the items in inventory according to markup. Produce a count of how many items fall into each of four categories.

```
- total "<Ø" isal - ipamt <= Ø,
"Ø - 2" isal - ipamt > Ø and isal - ipamt <= 2,
"2 - 4" isal - ipamt > 2 and isal - ipamt <= 4,
">4" isal - ipamt > 4 end
- go
        <Ø      Ø - 2      2 - 4         >4
-----------------------------------------

         2          8          4          Ø
- end
*remove item
item removed
```

Using Data from Other Files

Reports and queries can bring together, or "join," information from several files. SQL joins two record types by relating data common to both records, but the Listing Processor uses an "x.y" syntax to join records. Access the data from a field (y) in another record based on the reference to the key (x) of that record. This syntax produces the natural join of the files. You can join any number of files in a single query.

EXAMPLE: Select models made by manufacturers that are on "Evergreen" street. For these models, list the manufacturers name, the model number, and the description.

```
* select model where momano.madd = "*Evergreen*" end
2 records
* report model
- list momano.mname, monum, mdes end
- go
momano.mname                                monum mdes
-------------------------------------------------------------------------
A & H Industries, Inc.                      1ØØ1 vise grips
A & H Industries, Inc.                      1ØØ2 leather mallet
-------------------------------------------------------------------------

- end
* remove model
model removed
```

You may combine the dot (.) syntax used in "joins" with boolean or other operators as part of the selection.

EXAMPLE: Select the items that were ordered on 4/2/82 that were made by manufacturer number 1Ø3. For these, list the serial number, order date, and model description.

```
* select item where iorder.odate = 4/2/82 and imodel.momano = 1Ø3 end
4 records
* report item
- list sno,iorder.odate,imodel.mdes end
- go
      sno iorder.odate  imodel.mdes
-------------------------------------------------------
     1ØØ8 Ø4/Ø2/82      3" Phillips screwdriver
     1ØØ9 Ø4/Ø2/82      3" Phillips screwdriver
     1Ø1Ø Ø4/Ø2/82      6" slotted screwdriver
     1Ø11 Ø4/Ø2/82      6" slotted screwdriver
-------------------------------------------------------

- end
```

Sending Output to the Printer

The PRINT command sends the output of a report to the spooled output device. Appropriate page headings and titles are added. Titles can contain multiple lines.

EXAMPLE: Produce the percentage markup report of Section 11.6. Send the results to the hard-copy output device.

```
* remove item
item removed
* select item where 1 end
14 records
* report item
- sort "PCT. MARKUP" ((isal - ipamt) * 1ØØ.Ø) / isal end
- list "ADD DATE" iad, SERIAL/NUMBER sno, "PCT. MARKUP" end
- print "ACME HARDWARE WAREHOUSE/INVENTORY MARKUPS"
- end
```

The following output spools to the printer.

DATE: Ø7/28/82 PAGE: Ø1
TIME: 17:45

ACME HARDWARE WAREHOUSE
INVENTORY MARKUPS

ADD DATE	SERIAL NUMBER	PCT. MARKUP
Ø2/Ø8/82	1Ø12	-4.55
Ø1/15/82	1Ø1Ø	-3.81
Ø2/Ø8/82	1Ø13	2.84
Ø1/15/82	1ØØ8	6.38
Ø2/23/82	1234	11.8Ø
Ø2/15/82	1ØØ6	12.92
Ø2/23/82	1ØØ4	13.17
Ø2/25/82	1ØØ5	13.95
Ø2/15/82	1ØØ2	24.62
Ø2/15/82	1ØØ1	25.64
Ø2/15/82	1ØØ3	25.64
Ø2/15/82	1ØØ9	25.65
Ø2/15/82	1Ø11	31.Ø3
Ø1/19/82	1ØØ7	4Ø.ØØ

Manipulating Temporary Files

You can specify the name for a temporary file by using the **select into** syntax. When using this format, always select from all records in the specified file. After you make your selection, you can use the CALL command to rename the temporary file.

EXAMPLE: Select items whose model description begins either with 1/2" or contains the string "Phillips" into a temporary file called "myitems."

```
* select item into myitems where imodel.mdes = "1/2*" or
imodel.mdes = "*Phillips*" end
8 records
* report myitems
- list sno,imodel.mdes end
- go
        sno         imodel.mdes
-----------------------------------------
       1ØØ1 1/2" socket wrench
       1ØØ2 1/2" socket wrench
       1ØØ3 1/2" socket wrench
       1ØØ4 1/2" socket wrench
       1ØØ5 1/2" socket wrench
       1ØØ8 3" Phillips screwdriver
       1ØØ9 3" Phillips screwdriver
       1Ø13 1/2" socket wrench
-----------------------------------------

- end
```

You can review the current temporary files by using the LIST command. The argument to the command is the file name you wish to list. You may use wild card characters in the name. RECORD is the

underlying record type to which the temporary file refers. LENGTH is the number of records in the temporary file.

EXAMPLE: List all current temporary files.

```
* list *
FILE              RECORD       LENGTH

item              item             14
myitems           item              8
```

EXAMPLE: Rename the item temporary file "some_items."

```
* call myitems some_items
myitems renamed to some_items
```

EXAMPLE: List temporary files whose name contains the string "ome."

```
* list *ome*
FILE              RECORD       LENGTH

some_items        item              8
```

Customizing Output

To customize reports, you can specify your own output formats and line organization. You can achieve the first level of customization using string constants and the "under" statement. In the following example, colons separate columns, and a blank line separates manufacturers. When listing a constant expression such as a colon (:), you must always specify a heading.

EXAMPLE: Produce an attractive report from the manufacturer record.

```
* select manf where 1 end
6 records
* report manf
- list NUMBER mano, : :, NAME mname, " : " end
- list : under : end
- list ADDRESS madd under NAME end
- list : under " : " end
- list " " " " under ADDRESS end
- go
    NUMBER : NAME                                    :
           : ADDRESS                                 :

-------------------------------------------------------
       1ØØ : RH Smith Manufacturing                  :
           : 523 Galveston St.                       :

       1Ø1 : Precision Tool Co.                      :
           : 26ØØ West 16th Street                   :

       1Ø2 : A & H Industries, Inc.                  :
           : 2434 Evergreen Ave.                     :

       1Ø3 : Grover Parts and Supplies               :
           : 9462 Jackson Road                       :

       1Ø4 : The Tool Depot                          :
           : 7562 Orange Dr.                         :

       1Ø5 : BHP Ltd.                                :
           : 17385 Weatherby Rd.                     :

-------------------------------------------------------
- end
```

Numeric, float, and amount fields are normally printed using a default format. You can change this default format with the **using** syntax and a "template" composed of special characters, one for each

position in the field width. The special character at each position in the template indicates what is printed at that position. The special characters for numeric and amount fields are as follows:

- If a digit is in this position, print the digit. Otherwise, print a blank. This pads a numeric field with blanks on the left.

& - If a digit is in this position, print the digit. Otherwise, print a zero. This pads a numeric field with zeros on the left.

* - If a digit is in this position, print the digit. Otherwise, print an asterisk. You can use this to print amounts on checks, for example, which are frequently padded with asterisks on the left.

$ - If a digit is in this position, print the digit. Otherwise, print a dollar sign. If a dollar sign is already printed, print a space. You can use this to print either a fixed or a floating dollar sign.

+ - If a digit is in this position, print the digit. Otherwise, print a plus sign. If a plus sign is already printed, print a space.

- - If a digit is in this position, print the digit. Otherwise, if this is a negative number, print a minus sign. If a minus sign is already printed, print a space.

(- If a digit is in this position, print the digit. Otherwise, if this is a negative number, print a left parenthesis. If a left parenthesis is already printed, print a space.

) - If this is a negative number, print a right parenthesis in this position.

, - If a digit is to the left of this position, print a comma. Otherwise, print a space.

. - Print a decimal point in this position.

For float type fields, use a print specification exactly like the "printf" function in the C programming language. The print specification has the following format:

%[-][minimum field width][.][precision]f | e | g

Specifications enclosed in square brackets ([]) are optional. The list of items separated by the vertical bar (|) indicates that you must choose one of the items in the list. The percent sign (%) is required.

The optional minus sign (-) in front of the conversion specification indicates that the result is to be left justified in the field width. The minimum field width is a number indicating the minimum number of print positions in the result. If the result has fewer characters, it is padded to fill up the field width. Precision gives the number of digits to appear after the decimal point. If precision is Ø, no digits after the decimal point are printed. The conversion characters (f, e, and g) have the following meanings:

f - The field is converted to decimal notation in the style "[-]ddd.ddd", where the number of digits after the decimal point is equal to the precision specification. If the precision is missing, 6 digits are output ([-]ddd.ddd); if the precision is explicitly Ø, then no decimal point is printed.

e - The field is converted in the style "[-]d.ddde±dd", where there is one digit before the decimal point and the number of digits after is equal to the precision specification. If the precision is missing, 6 digits are output ([-]d.ddde±dd); if the precision is explicitly Ø, then no decimal point is printed.

g - The field is converted in style f or style e, whichever gives full precision in minimum space.

The following examples illustrate how various templates affect output:

Format	Value	Result
"#####"	123	" 123"
"#####.##"	Ø	" .ØØ"
"#####.&&"	Ø	" .ØØ"
"+++,+++,+++"	23456	" +23,456"
"---,---.&&"	23456.78	" 23,456.78"
"---,---.&&"	-2345.67	" -2,345.67"
"($$,$$&.&&)"	2345.ØØ	" $2,345.ØØ "
"($$,$$&.&&)"	-2345.67	"($2,345.67)"
"$##,##&.&&"	1234	"$ 1,234.ØØ"
"$**,***.&&"	123	"$**123.ØØ"
"%1Ø.2f"	12.3	" 12.3Ø"
"%1Ø.2f"	123.456	" 123.46"
"%12.4e"	123.456	" 1.235 e+ØØ2"
"%1Ø.4g"	123.456	" 123.4"
"%8.4g"	123456789	"1.23e+Ø8"

EXAMPLE: List the total dollar volume of items purchased by each customer, with a breakdown of our cost, what we were paid, and the gross profit for each order.

```
* select item where iorder != Ø end
14 items
* report item
- sort CUSTOMER iorder.ocust.cname,
       ORDER iorder
end
- total "COST OF/GOODS" ipamt using "  $##&.&&",
        ORDER/AMOUNT isal using "  $##&.&&",
        GROSS/PROFIT isal - ipamt using "  $##&.&&"
      by CUSTOMER, ORDER
end
- total "COST OF/GOODS",
        ORDER/AMOUNT,
        GROSS/PROFIT
end
- go
```

CUSTOMER	ORDER	COST OF GOODS	ORDER AMOUNT	GROSS PROFIT
	3	$ 16.6Ø	$ 2Ø.5Ø	$ 3.9Ø
Creative Manufacturing		$ 16.6Ø	$ 2Ø.5Ø	$ 3.9Ø
	2	$ 26.66	$ 29.54	$ 2.88
Reliable Construction Co.		$ 26.66	$ 29.54	$ 2.88
	1	$ 45.6Ø	$ 56.97	$ 11.37
Smith & Sons Hardware		$ 45.6Ø	$ 56.97	$ 11.37
		$ 88.86	$1Ø7.Ø1	$ 18.15

```
- end
```

You can place fields more precisely on the page by using column and line numbers to specify where they are to go. You can also turn off column headings with the "nohead" command so that the output can be directed to other TRS-XENIX filters or special forms on the printer. Note the use of the template "#####" to print the zip code in this example. This format overrides the default one, which has a wider field width.

EXAMPLE: Print mailing labels for customers, using six lines per label.

```
* select customer where 1 end
3 records
* report customer
- sort czip end
- list column 1 line 1 " " " " end
- list column 1 line 2 cname end
- list column 1 line 3 caddr end
- list column 1 line 4 ccity end
- list column 12 line 4 cstate end
- list column 15 line 4 czip using ##### end
- list column 1 line 5 "  " " " end
- nohead
- go

Creative Manufacturing
9124 Industrial Blvd.
Redding    CA 921Ø2


Reliable Construction Co.
2113 Folsom Blvd.
Sacramento CA 92343


Smith & Sons Hardware
1234 State Street
Wheatville CA 95817

- end
```

You can manipulate string fields with the substring and concatenation operators that are provided. You can print part of a string by specifying the starting and ending character positions as shown below. You can concatenate string fields and string constants by using the plus (+) symbol. The example below illustrates appending some blanks to a database field.

EXAMPLE: Print customer names and the description of items they have ordered in a 2-column format.

```
* report item
- sort uniquely iorder.ocust.cname, imodel.mdes end
- list CUSTOMER/NAME iorder.ocust.cname substr 1-11 + "   ",
       ITEM/DESCRIPTION imodel.mdes substr 1-11 end
- list iorder.ocust.cname substr 12-3Ø under CUSTOMER/NAME end
- list imodel.mdes substr 12-3Ø under ITEM/DESCRIPTION end
- list column 1 line 2 " " " " end
- go
CUSTOMER          ITEM
NAME              DESCRIPTION

---------------------------------------
Creative Ma       1/2" socket
nufacturing        wrench

Reliable Co       1/2" socket
nstruction Co.     wrench

Reliable Co       3" Phillips
nstruction Co.     screwdriver

Reliable Co       3/4" socket
nstruction Co.     wrench

Smith & Son       1/2" socket
s Hardware         wrench

Smith & Son       6" slotted
s Hardware        screwdriver

Smith & Son       combination
s Hardware         pliers

Smith & Son       leather mal
s Hardware        let

---------------------------------------

- end
* end
```

When you type <END> to the Listing Processor, the Main Menu appears.

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: lst_
```

Chapter 12
CREATING CANNED QUERIES

You can use the Listing Processor as part of an application by creating predefined reports that can run as programs from the menu handler. You can also use other TRS-XENIX utilities to extend the formatting capabilities of the Listing Processor. This use of the Listing Processor creates a different "front-end" for queries. Instead of typing in a long and complex query command, you can simply fill in the specific selection criteria.

If you are not familiar with the shell, skim this chapter and the following chapter and continue with the "Query by Forms" section. Query by Forms lets you do most of the tasks explained in this section, and it's easier to learn. When you have some experience with the shell, you can return to these exercises.

The first report is a list of items in inventory, with a subtotal by type or item. The user supplies a date range in which the items were acquired. Here is the selection screen, followed by the finished report.

```
[itm3ØØ]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                   Item Listing


STARTING ACQUISITION DATE: 1/1/82

ENDING ACQUISITION DATE:   3/1/82

[running]
_
```

DATE: Ø9/23/82
TIME: 15:ØØ

PAGE: Ø1

ITEM LISTING - ITM3ØØ

ITEM NO	MANF NO	MODEL NO	ADD DATE	PRICE	DESC	TOTAL
1ØØ1	1ØØ	1ØØ1	Ø2/15/82	9.75		
1ØØ2	1ØØ	1ØØ1	Ø2/15/82	9.75		
1ØØ3	1ØØ	1ØØ1	Ø2/15/82	9.75		
1ØØ4	1Ø1	55Ø71	Ø2/23/82	1Ø.25		
1ØØ5	1Ø1	55Ø71	Ø2/25/82	1Ø.75		
1Ø13	1Ø4	1Ø12	Ø2/Ø8/82	9.5Ø		
					1/2" socket wrench	6
1ØØ8	1Ø3	61117	Ø1/15/82	2.35		
1ØØ9	1Ø3	61117	Ø2/15/82	2.69		
					3" Phillips screwdriver	2
1234	1ØØ	1ØØ2	Ø2/23/82	15.ØØ		
					3/4" socket wrench	1
1Ø1Ø	1Ø3	91117	Ø1/15/82	2.89		
1Ø11	1Ø3	91117	Ø2/15/82	3.19		
					6" slotted screwdriver	2
1ØØ6	1Ø1	55271	Ø2/15/82	6.89		
1Ø12	1Ø4	1Ø11	Ø2/Ø8/82	5.5Ø		
					combination pliers	2
1ØØ7	1Ø2	1ØØ2	Ø1/19/82	8.75		
					leather mallet	1
						14

To create a "canned" report such as this, exit Unify and return to the shell. Unify supplies an executable named "sh" that starts up the shell, so you can now exit to the shell as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu




SELECTION: sh_
```

("sh" is the only executable defined in Unify that does not appear as an option on any menu.)

The following screen appears:

```
[sh]                                UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                   TRS-XENIX Shell


unify> _
```

Note: The prompt character is different to remind you that you are now in a subshell. You are still in the executable directory in which you started.

To create command scripts, use the vi text editor available on your system. Start up the editor by typing vi in response to the Unify prompt (>). Press <i> and begin entering the commands for the ITM3ØØ file shown below. If you make an error before pressing <ENTER> to move to the next line, press <Backspace> to move back to the incorrect characters and retype the line. If you discover that you have made an error in a line which has already been entered, **type <ESC> to exit insert mode,** and then move the cursor to the appropriate line and position in the line by entering:

k to move up
j to move down
l to move left
h to move right

Once you have positioned the cursor at the incorrect line, you may insert or delete characters to correct the entry. Type:

<x> to delete a character
<a> to insert or append a character

To terminate an insert (<a>), press <ESC>.

Once the file has been correctly entered, type:

:w ITM3ØØ to save the file

then

:q or **zz** to exit the editor.

If you need to make changes to the ITM3ØØ command script, you may reenter vi by typing **vi ITM3ØØ**. (For more information on how to use the vi text editor, refer to Appendix B in the Unify Reference Manual.) You may now create the second command script (itm3ØØ.r) by following the above steps.

File: ITM3ØØ

```
echo
echo
echo-n "STARTING ACQUISITION DATE: "
read sad
echo
echo-n "ENDING ACQUISITION DATE:   "
read ead
echo remove item > itm3ØØ.i
echo select item where iad gte $sad and \
iad lte $ead end >> itm3ØØ.i
echo report item >> itm3ØØ.i
cat itm3ØØ.r >> itm3ØØ.i
echo
echo [running]
LST itm3ØØ.i > /dev/null
```

File: itm3ØØ.r

```
sort DESC imodel.mdes, "MANF NO" imodel_momano,
"MODEL NO" imodel_monum end
list "ITEM NO" sno, "MANF NO", "MODEL NO", "ADD DATE"
iad, PRICE isal end
total TOTAL 1 by DESC end
total TOTAL end
print "ITEM LISTING - ITM3ØØ"
end
```

The first file, ITM3ØØ, builds a screen to get the selection options from the user, uses these responses to create the query input file, and then starts up the report.

You use echo to move the cursor down the screen, and you use read to get the selection options. The query specifications are echoed into the query input source file, itm3ØØ.i, after the dates have been entered. (Note that you enter the condition gte instead of >= to differentiate this condition from the shell commands that define standard input and output.) The report format, contained in itm3ØØ.r, is appended to itm3ØØ.i using cat. The listing processor is then

called, with any diagnostics being sent to /dev/null, so they don't appear on the screen.

ITM3ØØ must be made executable, so change its mode to executable as follows:

```
chmod +x ITM3ØØ
```

Now you can execute the report by typing **ITM3ØØ** beside the Unify prompt (>). The item listing report is an example of sending a canned query to the printer. You can also create a command script query that sends its results to the screen. The next command script we create lets you do that.

For example, to produce a report that shows Acme how long items are kept in inventory, the selection option is a date from which to do the aging. The screen displays a matrix of models, with the total number of items of that model in inventory for a given time period. The time periods are less than 7 days from the aging date; 7 to 14 days from the aging date; 15 to 3Ø days; and more than 3Ø days. For purposes of the report, we assume that when an item is sold it is deleted from the file, so that all items in the database are actually in inventory. Using the data in the test database, an example report is as follows:

```
[itm31Ø]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                              Inventory Aging Report

 INVENTORY AGING AS OF: 3/1/82

 MODEL                        < 7 days  7-14 days  15-3Ø days  > 3Ø days
-------------------------------------------------------------------------
 1/2" socket wrench               2         3          1          Ø
 3" Phillips screwdriver          Ø         1          Ø          1
 3/4" socket wrench               1         Ø          Ø          Ø
 6" slotted screwdriver           Ø         1          Ø          1
 combination pliers               Ø         1          1          Ø
 leather mallet                   Ø         Ø          Ø          1


Type <ENTER> when finished_
```

This report shows that 1 combination pliers has been in inventory for 7-14 days, and 1 for 15-3Ø days. The 1 leather mallet in stock has been there for more than 3Ø days. The one 3/4" socket wrench has been in inventory for fewer than 7 days. Enter the following file to perform this report.

File: ITM31Ø

```
echo
echo
echo-n "INVENTORY AGING AS OF: "
read adate
echo
echo remove item > itm31Ø.i
echo select item where iad lte $adate end >> itm31Ø.i
echo report item >> itm31Ø.i
echo sort MODEL imodel.mdes end >> itm31Ø.i
echo total \" \< 7 days\" $adate - iad lt 7, \
\" 7-14 days\" $adate - iad gte 7 and $adate - iad lt 15,\
\" 15-3Ø days\" $adate - iad gte 15 and $adate - iad lte 3Ø,\
\" \> 3Ø days\" $adate - iad gt 3Ø by MODEL end >> itm31Ø.i
echo go >> itm31Ø.i
LST itm31Ø.i
echo
echo-n Enter \<cr\> when finished
read x
```

As with ITM3ØØ, echo moves the cursor and prompts for input. The query input file, itm31Ø.i, is created with the add date parameter both in the selection part and in the report part. The results of this query are sent to the screen with the GO command. The final "read x" keeps the shell from returning to MENUH until you are finished looking at the screen and type <ENTER>. The back slash lets special characters be displayed on the screen. After you enter the file, be sure to change its mode to executable so that the shell can execute it.

You can combine reports with the text-processing tools available on the TRS-XENIX Development System to produce just about any kind of report. Nroff commands can be printed as string constants that can then be processed later. In addition, you can use filters such as sed and awk to further customize the output. The following examples of form letters illustrate some of the possibilities.

Note: The text-processing tools illustrated in the following examples are available only with the TRS-XENIX Development System.

You can construct the letters using three files to perform the processing. The first is the control file, LTR3ØØ, that directs the process. It starts the listing processor, taking its input from the report file ltr3ØØ.r and sending the output to a temporary file. Then nroff is used to format the repetitive letters, one for each company name and address. Finally, the temporary file is removed.

File: LTR3ØØ

```
LST ltr3ØØ.r > ltr3ØØ.tmpl
echo ".ex" >> ltr3ØØ.tmpl
nroff ltr3ØØ < ltr3ØØ.tmpl | lpr
rm ltr3ØØ.tmpl
```

File: ltr3ØØ.r

```
remove customer
select customer where 1 end
report customer
sort czip end
list column 1 line 1 cname end
list column 1 line 2 caddr end
list column 1 line 3 ccity end
list column 12 line 3 cstate end
list column 15 line 3 czip using ##### end
list column 1 line 4 " " ".sp 2" end
list column 1 line 5 "  " "" end
nohead
go
```

File: ltr3ØØ

```
.po 2
.nf
.sp 6
June 15, 1983
.sp 2
.rd
Gentlemen:

.fi
.na
Thank you for your business!
We at Acme are looking forward to providing you with the
products and services you need, when you need them, and at
reasonable prices.
.sp 2
.nf
Sincerely,
.sp 3
John J. Jones
Manager
.bp
.so ltr3ØØ
```

After you enter the three files, make LTR3ØØ executable so it can run as a program from the shell. Here are the letters produced by running LTR3ØØ.

June 15, 1983

Creative Manufacturing
9124 Industrial Blvd.
Redding CA 921Ø2

Gentlemen:

Thank you for your business! We at Acme are looking forward to providing you with the products and services you need, when you need them, and at reasonable prices.

Sincerely,

John J. Jones
Manager

June 15, 1983

Reliable Construction Co.
2113 Folsom Blvd.
Sacramento CA 92343

Gentlemen:

Thank you for your business! We at Acme are looking forward to providing you with the products and services you need, when you need them, and at reasonable prices.

Sincerely,

John J. Jones
Manager

June 15, 1983

Smith & Sons Hardware
1234 State Street
Wheatville CA 95817

Gentlemen:

Thank you for your business! We at Acme are looking forward to providing you with the products and services you need, when you need them, and at reasonable prices.

Sincerely,

John J. Jones
Manager

Chapter 13
REGISTERING PROGRAMS WITH THE MENU HANDLER

In the previous section, you created several command scripts that you can execute as programs. It is more convenient, however, to select them from a menu. To do this, return to Unify, register them with the menu handler as executables, and then place them on a menu.

First, type <CTRL> <D> to terminate the subshell, and you return to the menu from which you executed "sh." Select Executable Maintenance as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. SQL - Query/DML Language

5. System Menu









SELECTION: execmnt_
```

You may also select Executable Maintenance as an option from the MENUH Screen Menu, which you can access from the System Menu. The executable maintenance screen appears as follows:

```
+-------------------------------------------------------------------+
|[execmnt]                        UNIFY SYSTEM                      |
|                               5 OCT 1982 - 15:25                  |
|                            Executable Maintenance                 |
|                                                                   |
|EXECUTABLE'S NAME: ''''''''                                        |
|USES SYSRECEV ? '                                                  |
|PROGRAMS:                                                          |
|CMD  NAME       HEADING                          SCREEN    DIRECTORY|
|  Ø  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  1  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  2  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  3  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  4  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  5  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  6  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  7  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  8  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|  9  """""""""  """""""""""""""""""""""""""""""  """""""""  """""""""|
|                                                                   |
|                                                                   |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE _                             |
+-------------------------------------------------------------------+
```

```
''''''''' => executable data entry area
""""""""" => program data entry area
```

This form lets you inquire, add, modify, and delete executables and programs within those executables. An executable is a TRS-XENIX file that you may execute by typing its name at the shell prompt. An executable may contain multiple entry points, each of which may start up a different program. This is usually only applicable to custom programs, not command scripts, which generally handle the execution of

programs or usage of other command scripts within the script itself. An explanation of each screen prompt is given below.

[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE

This prompt lets you choose an operational mode for the program.

I - Inquire, lets you see an executable that is already registered with the menu handler, MENUH.

A - Add, lets you register a new executable with MENUH.

M - Modify, lets you modify HEADING, SCREEN, and DIRECTORY for programs within an existing executable or add and delete programs.

D - Delete, lets you delete an existing executable and all its associated programs.

EXECUTABLE'S NAME

An 8-character string that is the name of the TRS-XENIX file that contains the executable. This file name passes to the shell when you select a program within the executable.

USES SYSRECEV ?

Type <Y> if the executable is loaded with the "main" Unify routine called sysrecev that interprets the arguments passed by MENUH. This only applies to custom programs. For shell scripts or for executables that contain only one program and don't need it, type <N> to indicate that sysrecev is not used.

CMD

Use this column to enter a command to perform an operation on the current program. The cursor moves up and down in the multiline data entry area of the screen in response to <F1> and <ENTER>. The position of the cursor marks the current program. The following are valid commands:

a - Adds a new program to the executable. This response is valid only on the first empty line in the multiline area. In other words, you must move the cursor to the first blank line before you can add a new program.

m - Modifies HEADING, SCREEN, or DIRECTORY for the current program. If you wish to modify NAME, you must delete the line and reinsert it.

d - Deletes the current program.

q - Leaves the multiline data entry area and positions the cursor at the USES SYSRECEV prompt.

NAME

An 8-character name of the program within the executable. This name is the one typed in response to the menu SELECTION prompt in order to execute the program. If the executable is a custom program and the name is an entry point into the executable, this name must be the same as the host language function (C and RM COBOL) name that is the entry point for the program.

HEADING

A 36-character string that the screen displays on the Line 3 of the screen when the program is running and on menus where this program is a selection option. You can enter literal text or "escaped" text that indicates the

prompt is to be displayed in a special mode. The escape character is the tilde (~). Model II/12/16 and Data Terminal-1 support the reverse video.

~r	-	Starts reverse video
~s	-	Ends reverse video

In addition Data Terminal-1 supports the underline mode.

~u	-	Starts underline
~v	-	Ends underline

If you start a special display mode for a heading, be sure to end it. The screen displays the special mode only on menus from which this menu screen may be selected and not on Line 3 of the heading for this menu.

SCREEN

The name of an SFORM screen sysrecev displays before the user program is started. This entry is optional.

DIRECTORY

For custom programs, this is the name of the TRS-XENIX directory that contains the source code for the program. The Unify program loading shell script (uld) expects the archive containing the program to be in this directory also. The relative path ../src/ is prefixed to the directory name entered here. If no entry is made, uld uses the first three characters of the program name for the directory name. For example, if the program name is ord1ØØ, uld looks in ../src/ord for the archive containing ord1ØØ.

To add an executable, type **<a>** in response to the prompt at the bottom of the screen. The prompt clears and the cursor moves to the EXECUTABLE'S NAME prompt to let you add a new executable. Add ITM3ØØ as follows:

```
[execmnt]                          UNIFY SYSTEM
[A]DD                            5 OCT 1982 - 15:25
                               Executable Maintenance

EXECUTABLE'S NAME: ITM3ØØ
USES SYSRECEV ? N
PROGRAMS:
CMD NAME       HEADING                          SCREEN    DIRECTORY
m_Ø itm3ØØ
  1
  2
  3
  4
  5
  6
  7
  8
  9
```

Executable Maintenance puts in the first program name by default, giving it the same name as the executable file but mapped to lower-case letters. You may change the name of a program by deleting and reinserting it.

To put in a heading, type <m> under the CMD column as shown. <ENTER> moves the cursor forward; <F1> moves it backward. Type <F2> when the cursor is in the HEADING column to move the cursor back out to the CMD column. Finish the entry for ITM3ØØ as follows:

```
[execmnt]                          UNIFY SYSTEM
[A]DD                            5 OCT 1982 - 15:25
                               Executable Maintenance

EXECUTABLE'S NAME: ITM3ØØ
USES SYSRECEV ? N
PROGRAMS:
CMD NAME       HEADING                              SCREEN    DIRECTORY
  Ø itm3ØØ     Item Listing
  1
  2
  3
  4
  5
  6
  7
  8
  9
```

Type <F2> to store the entry for the program. Then type <Q> to move back to the USES SYSRECEV prompt. Now type <F1> three times to return to the Main Menu. You can now execute the program itm3ØØ by typing its name at the selection menu prompt, but for ease of use, put it on a menu. Select menu maintenance by typing its name, **menumnt**, as follows:

```
+-------------------------------------------------------------------+
|[entmenu]                          UNIFY SYSTEM                    |
|                                 5 OCT 1982 - 15:25                |
|                                     Main Menu                     |
|                                                                   |
|1. Manufacturer Maintenance                                        |
|                                                                   |
|2. Model Maintenance                                               |
|                                                                   |
|3. Item Maintenance                                                |
|                                                                   |
|4. SQL - Query/DML Language                                        |
|                                                                   |
|5. System Menu                                                     |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: menumnt_                                                |
|                                                                   |
+-------------------------------------------------------------------+
```

You may also select Menu Maintenance as an option from within the MENUH Screen Menu. The following screen appears:

```
+-------------------------------------------------------------------+
|[menumnt]                        UNIFY SYSTEM                      |
|                              5 OCT 1982 - 15:25                   |
|                               Menu Maintenance                    |
|                                                                   |
| NAME:            HEADING:                                         |
| CMD   LINE   MENU/PROG   M/P    PROMPT                            |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE  _                            |
+-------------------------------------------------------------------+
```

Select modify mode, and call up entmenu on which to put the new selection. Type <ENTER> repeatedly until the first blank line appears as shown on the following pages. Then type **<a>** to add a new menu line. The screen appears as follows just before you type <ENTER> to add the new line:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD  LINE  MENU/PROG  M/P   PROMPT
        1   sman1ØØ     E    Manufacturer Maintenance
        2   smod1ØØ     E    Model Maintenance
        3   sitm1ØØ     E    Item Maintenance
        4   sql         P    SQL - Query/DML Language
        5   sysmenu     M    System Menu
  a_
```

Add the new line as Line 4 so that the new entry follows Item Maintenance. The screen appears as follows just before you type <F2> to finish the entry and reorder the lines on the screen.

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD  LINE   MENU/PROG   M/P    PROMPT
        1    sman1ØØ      E     Manufacturer Maintenance
        2    smod1ØØ      E     Model Maintenance
        3    sitm1ØØ      E     Item Maintenance
        4    sql          P     SQL - Query/DML Language
        5    sysmenu      M     System Menu
        4    itm3ØØ       P     Item Listing
```

Now type <F2> and the menu is rearranged as follows:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD   LINE   MENU/PROG   M/P    PROMPT
         1    sman1ØØ      E     Manufacturer Maintenance
         2    smod1ØØ      E     Model Maintenance
         3    sitm1ØØ      E     Item Maintenance
         4    itm3ØØ       P     Item Listing
         5    sql          P     SQL - Query/DML Language
         6    sysmenu      M     System Menu
  _
```

Now type <Q> to return to the HEADING prompt, and then type <F1> repeatedly until you have exited Unify and then reexecute **ututorial** to see the results of the change to the menu, as the current menu is displayed from a copy kept in memory. Exiting Unify forces the menu to be read from the data dictionary again. After you reexecute **ututorial**, the screen shows the following menu:

```
[entmenu]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                    Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. SQL - Query/DML Language

6. System Menu




SELECTION: 4_
```

Select 4, Item Listing, and fill in the entries as follows. The report then prints out.

```
[itm3ØØ]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                   Item Listing


STARTING ACQUISITION DATE: 1/1/82

ENDING ACQUISITION DATE:   3/1/82

[running]
-
```

To give the data entry supervisor access to ITM3ØØ, execute empmnt and add itm3ØØ to the list of allowable programs. The following screens show the steps in the process.

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. SQL - Query/DML Language

6. System Menu




SELECTION: empmnt_
```

```
+-------------------------------------------------------------------+
|[empmnt]                        UNIFY SYSTEM                       |
|                              5 OCT 1982 - 15:25                   |
|                            Employee Maintenance                   |
|                                                                   |
|LOGIN ID:       NAME:                                              |
|GROUP ID:       NAME:                                              |
|                    SYSTEM ENTRY PT:                -              |
|                                                                   |
|                                                                   |
|ACCESS LEVELS DIFFERENT FROM GROUP:                                |
|  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL                      |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE m_                            |
|                                                                   |
+-------------------------------------------------------------------+
```

```
[empmnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                    SYSTEM ENTRY PT: entmenu  - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL


   1  smanlØØ    E     y       y   y   y   y
   2  smodlØØ    E     y       y   y   y   y
   3  sitmlØØ    E     y       y   y   y   y
   4  sql        P     y


[N]ext page, [P]rev page, [A]dd line, or number: a
```

```
[empmnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                     SYSTEM ENTRY PT: entmenu  - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL
      itm3ØØ     P     y

   1  sman1ØØ    E     y       y   y   y   y
   2  smod1ØØ    E     y       y   y   y   y
   3  sitm1ØØ    E     y       y   y   y   y
   4  sql        P     y
```

Type <F2> to complete the above entry. The following screen appears:

```
[empmnt]                              UNIFY SYSTEM
[M]ODIFY                          5 OCT 1982 - 15:25
                                Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                     SYSTEM ENTRY PT: entmenu  - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL

   1  sman1ØØ    E     y       y   y   y   y
   2  smod1ØØ    E     y       y   y   y   y
   3  sitm1ØØ    E     y       y   y   y   y
   4  sql        P     y
   5  itm3ØØ     P     y
```

Now type <F1> five times, and the entry menu, entmenu, appears. Call up execmnt again so that you can register ITM31Ø using Executable Maintenance, as you did with ITM3ØØ. After that, use Menu Maintenance and Employee Maintenance to give other users access to the program. Only the three final screens are shown here. If necessary, refer to the previous sections to see how to make the entries.

```
[execmnt]                          UNIFY SYSTEM
[A]DD                            5 OCT 1982 - 15:25
                               Executable Maintenance

EXECUTABLE'S NAME: ITM31Ø
USES SYSRECEV ? N
PROGRAMS:
CMD NAME       HEADING                          SCREEN    DIRECTORY
  Ø itm31Ø     Inventory Aging Report
_ 1
  2
  3
  4
  5
  6
  7
  8
  9
```

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD   LINE   MENU/PROG   M/P    PROMPT
         1    sman1ØØ      E    Manufacturer Maintenance
         2    smod1ØØ      E    Model Maintenance
         3    sitm1ØØ      E    Item Maintenance
         4    itm3ØØ       P    Item Listing
         5    itm31Ø       P    Inventory Aging Report
         6    sql          P    SQL - Query/DML Language
         7    sysmenu      M    System Menu
  -
```

```
[empmnt]                              UNIFY SYSTEM
[M]ODIFY                          5 OCT 1982 - 15:25
                                Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                     SYSTEM ENTRY PT: entmenu  - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL


   1  sman1ØØ    E     y       y   y   y   y
   2  smod1ØØ    E     y       y   y   y   y
   3  sitm1ØØ    E     y       y   y   y   y
   4  sql        P     y
   5  itm3ØØ     P     y
   6  itm31Ø     P     y
```

Chapter 14
QUERY BY FORMS

ENTER lets you query the database using screen forms. It also gives you the data entry capability illustrated in the section "Adding Data with ENTER." If you fill in the fields on the form with the values you want, ENTER constructs the optimal query to find the data. Once the data is found, you can update or delete it. In addition, you can associate ENTER with a report format so that a report can be produced from the retrieved data. You can use any SFORM screen that ENTER can drive in this manner.

The Item Maintenance screen form provides a view of the database that includes most of the database fields. You can use it to perform some interesting queries.

Start up Item Maintenance as follows:

```
[entmenu]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                    Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: 3_
```

The following screen appears:

```
+-------------------------------------------------------------------+
|[sitml00]                          UNIFY SYSTEM                    |
|                                 5 OCT 1982 - 15:25                |
|                                  Item Maintenance                 |
|                                                                   |
|ITEM NUMBER    :                                                   |
|MANUFACTURER   :                                                   |
|MODEL          :                                                   |
|                                                                   |
|                                                                   |
|ADD DATE       :                                                   |
|PRICE          :                                                   |
|ORDER NUMBER   :                                                   |
|PURCHASE PRICE:                                                    |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE m_                            |
+-------------------------------------------------------------------+
```

Type <M> <ENTER> to select modify mode as shown above. The following screen appears:

```
[sitml00]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                 Item Maintenance

ITEM NUMBER     :_
MANUFACTURER    :
MODEL           :


ADD DATE        :
PRICE           :
ORDER NUMBER    :
PURCHASE PRICE:


Begin search [CTRL E], Clear field [CTRL Z], Exit [CTRL X]
```

The prompt at the bottom lists the operations available at this time.

CTRL E - Starts the search with the selection options you enter. Alternatively, you can type <ENTER> until you pass the last screen field, and the search begins.

CTRL Z - Clears the current selection specification (that is, "zaps" it). For example, if you enter an ADD DATE specification and decide the date does not matter, position the cursor at the ADD DATE prompt and type <CTRL> <Z>. The specification is erased, and ADD DATE is not used in the query.

CTRL X - Terminates the current ENTER screen and returns you to the Main Menu. In contrast to the way you use the two previous CTRL characters, you may use CTRL X any time an ordinary character can be accepted. This gives you a quick way to exit an ENTER screen.

The simplest query is to find a particular item by its primary key, the serial number. Enter the serial number 1ØØ2 as shown next and the following screen appears:

```
[sitml ØØ]                          UNIFY SYSTEM
[M]ODIFY                          5 OCT 1982 - 15:25
                                   Item Maintenance

ITEM NUMBER    :1ØØ2
MANUFACTURER   :1ØØ      RH Smith Manufacturing
MODEL          :1ØØ1     1/2" socket wrench

ADD DATE       :Ø2/15/82
PRICE          :     9.75
ORDER NUMBER   :3
PURCHASE PRICE:      7.35
```

Since this is a primary key access, the record is found immediately with no searching required. You can now type <ENTER> to get to any desired field on the screen and update the record by typing

in new data. (Press <ENTER> to move down the screen or to the right; press <F1> to move up or to the left.) Type <ENTER> to get to the PRICE field and change the price to 9.5Ø as shown next:

```
[sitml ØØ]                        UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Item Maintenance

ITEM NUMBER    :1ØØ2
MANUFACTURER   :1ØØ      RH Smith Manufacturing
MODEL          :1ØØ1     1/2" socket wrench


ADD DATE       :Ø2/15/82
PRICE          :     9.5Ø_
ORDER NUMBER   :3
PURCHASE PRICE:      7.35
```

Type <ENTER> three times to complete the entry, and then type <F2> to clear the screen so that you perform a more complex query.

ENTER provides two basic query capabilities -- exact matching (you type exactly what you want the selected records to contain) and inexact, or generic, matching (you provide special characters that have different meanings during the matching process).

To specify an exact match, you fill in the field or fields on the screen with exactly what you want to see. To specify an inexact match on string fields, you use the same set of special characters that SQL and the Listing Processor use. The special characters and their meanings are as follows.

? - Wild character (question mark) matches any single character. For example, if you want to find all the Smiths but do not know if they are spelled "Smith" or "Smyth," you can use the specification "Sm?th".

* - Wild string (asterisk) matches any string of characters of any length, including zero length strings (also called "null strings"). ENTER automatically appends an asterisk to the end of all string specifications, since this is what is most commonly meant.

[...] - Three dots stand for a set of characters that define a "character class." The character class matches any single character that is a member of the class. You specify ranges of characters by separating two characters with a dash (-). For example, you can use all upper-case letters to represent the class

[ABCDEFGHIJKLMNOPQRSTUVWXYZ]

Or, more conveniently you can use [A-Z]. You can use [a-zA-Z] to represent all letters, upper- and lower-case. You can construct other classes in a similar manner.

You can construct inexact matches on nonstring fields by using the following set of expressions. f1 and f2 refer to the field values you supply.

> f1 - Greater than, matches all fields with values greater than the entered value.

< f1 - Less than, matches all fields with values less than the entered value.

!f1 - Logical "not," matches all fields that do not match the entered value.

f1-f2 - Range, matches all fields that match the entered values or that are between the entered values. This is equivalent to >= f1 AND <= f2.

!f1-f2 - Matches all fields that are outside the range of entered values. This is equivalent to < f1 OR > f2.

You can fill in any number of fields on the screen by using the above expressions. The result is to AND all of the qualifications together to produce a query.

The first example query is an exact match on two different fields. Find all items in inventory made by RH Smith Manufacturing (#1ØØ) that were acquired on 2/15/82. To do this query, fill in the MANUFACTURER and ADD DATE prompts as shown next.

```
+-----------------------------------------------------------------+
|[sitml00]                        UNIFY SYSTEM                    |
|[M]ODIFY                      5 OCT 1982 - 15:25                 |
|                               Item Maintenance                  |
|                                                                 |
|ITEM NUMBER    :                                                 |
|MANUFACTURER   : 100    RH Smith Manufacturing                   |
|MODEL          :                                                 |
|                                                                 |
|                                                                 |
|ADD DATE       :2.15.82                                          |
|PRICE          :_                                                |
|ORDER NUMBER   :                                                 |
|PURCHASE PRICE:                                                  |
|                                                                 |
|                                                                 |
|                                                                 |
|                                                                 |
|                                                                 |
|                                                                 |
|Begin search [CTRL E], Clear field [CTRL Z], Exit [CTRL X]       |
+-----------------------------------------------------------------+
```

Type <CTRL> <E> to begin the search. The following screen appears:

```
[sitml00]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Item Maintenance

ITEM NUMBER    :1001
MANUFACTURER   :100   RH Smith Manufacturing
MODEL          :1001  1/2" socket wrench


ADD DATE       :02/15/82
PRICE          :     9.75
ORDER NUMBER   :1
PURCHASE PRICE:      7.25









[N]EXT, [P]REVIOUS, [M]ODIFY, [S]TOP _
searched:       4 selected:       3 current:      1
```

Only four records were examined to satisfy the query, since there is an explicit relationship defined through the REF column in the fields screen of Schema Maintenance between manufacturers and items that identifies all the items made by a particular manufacturer. The performance advantages of explicit relationships are not immediately apparent in a database this small, but you can easily see them in a database that contains tens of thousands of records in a file. In such a case, using explicit relationships lets you access data ten times faster than using indexes.

The first record in the set is displayed as shown. The first prompt at the bottom of the screen shows what operations are valid at this time. Six operations are possible, depending on the current mode (inquire, modify, or delete) and whether or not a report format is associated with the screen.

[N]EXT, [P]REVIOUS, [M]ODIFY, [D]ELETE, [R]EPORT, [S]TOP

The options that are displayed at this time will be chosen from this set of operations. [N]EXT, [P]REVIOUS, and [S]TOP are displayed for all modes. [M]ODIFY and [D]ELETE are displayed only when in modify or delete modes, respectively. [R]EPORT is displayed in inquire mode, provided a report format is associated with the ENTER screen. The meaning of the various responses is as follows:

N - Displays the next record in the set of retrieved records. <ENTER> also performs this task.

P - Displays the previous record in the set of retrieved records.

M - Moves the cursor to the first screen field so that you can modify the data in the current record.

D - Deletes the current record from the database.

R - Takes the current set of retrieved records and formats them for output according to the associated report script. You can associate as many as four report scripts with a screen form. If only one script exists, that script is started immediately. Otherwise, the screen displays the available scripts, and you can select one.

S - Clears the current set of retrieved records and clears the screen of data so that you can

perform a different query. <F1> also performs this task.

searched: nnnnnn

Shows how many records were examined to satisfy the query.

selected: nnnnnn

Shows how many records were selected.

current: nnnnnn

Shows which record within the set is currently displayed on the screen. This number ranges from 1 to the number of selected records.

Type <ENTER> or <N> <ENTER> to advance to the next record in the set. The following screen appears:

```
[sitml00]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Item Maintenance

ITEM NUMBER    :1002
MANUFACTURER   :100   RH Smith Manufacturing
MODEL          :1001  1/2" socket wrench


ADD DATE       :02/15/82
PRICE          :    9.50
ORDER NUMBER   :3
PURCHASE PRICE:    7.35









[N]EXT, [P]REVIOUS, [M]ODIFY, [S]TOP
searched:      4 selected:      3 current:      2
```

Notice that the "current" count is updated to show that you are on the second record in the set. Type <M> <ENTER> to modify this record as shown, and the cursor moves to the ITEM NUMBER prompt. Type <ENTER> to get to the PRICE prompt. Then change the price back to 9.75 as shown next.

```
+-------------------------------------------------------------------+
|[sitm1ØØ]                        UNIFY SYSTEM                      |
|[M]ODIFY                      5 OCT 1982 - 15:25                   |
|                               Item Maintenance                    |
|                                                                   |
|ITEM NUMBER    :1ØØ2                                               |
|MANUFACTURER   :1ØØ      RH Smith Manufacturing                    |
|MODEL          :1ØØ1     1/2" socket wrench                        |
|                                                                   |
|                                                                   |
|ADD DATE       :Ø2/15/82                                           |
|PRICE          :     9.75                                          |
|ORDER NUMBER   :3                                                  |
|PURCHASE PRICE:      7.35                                          |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|searched:       4 selected:       3 current:       2              |
+-------------------------------------------------------------------+
```

Now type <F2> to redisplay the prompt at the bottom. To perform another query, type <S> <ENTER> as shown next to stop examining records and clear the screen and current set.

```
[sitm1ØØ]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Item Maintenance

ITEM NUMBER     :1ØØ2
MANUFACTURER    :1ØØ      RH Smith Manufacturing
MODEL           :1ØØ1     1/2" socket wrench


ADD DATE        :Ø2/15/82
PRICE           :     9.75
ORDER NUMBER    :3
PURCHASE PRICE:     7.35









[N]EXT, [P]REVIOUS, [M]ODIFY, [S]TOP s_
searched:       4 selected:       3 current:       2
```

The screen clears, and the prompts at the bottom change to show the options that are now valid.

```
+-------------------------------------------------------------------------+
|[sitml00]                          UNIFY SYSTEM                          |
|[M]ODIFY                        5 OCT 1982 - 15:25                       |
|                                  Item Maintenance                       |
|                                                                         |
|ITEM NUMBER    :_                                                        |
|MANUFACTURER   :                                                         |
|MODEL          :                                                         |
|                                                                         |
|                                                                         |
|ADD DATE       :                                                         |
|PRICE          :                                                         |
|ORDER NUMBER   :                                                         |
|PURCHASE PRICE:                                                          |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|Begin search [CTRL E], Clear field [CTRL Z], Exit [CTRL X]               |
+-------------------------------------------------------------------------+
```

The next query uses the wild string character in the model description field. To find all pliers in inventory, type <ENTER> until the cursor is at the field immediately to the right of the MODEL prompt. Then enter the data as shown next:

```
[sitml00]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Item Maintenance

ITEM NUMBER    :
MANUFACTURER   :
MODEL          :      * pliers_


ADD DATE       :
PRICE          :
ORDER NUMBER   :
PURCHASE PRICE:




Begin search [CTRL E], Clear field [CTRL Z], Exit [CTRL X]
```

Type <ENTER> and the model selection is performed.

```
+-------------------------------------------------------------------+
|[sitml00]                        UNIFY SYSTEM                      |
|[M]ODIFY                      5 OCT 1982 - 15:25                   |
|                               Item Maintenance                    |
|                                                                   |
|ITEM NUMBER    :                                                   |
|MANUFACTURER   :                                                   |
|MODEL          :     * pliers                                      |
|                                                                   |
|                                                                   |
|ADD DATE       :                                                   |
|PRICE          :                                                   |
|ORDER NUMBER   :                                                   |
|PURCHASE PRICE:                                                    |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|Begin search [CTRL E], Clear field [CTRL Z], Exit [CTRL X]         |
|Selecting model_                                                   |
+-------------------------------------------------------------------+
```

This specification selects all models the description of which contains the string "pliers". ENTER automatically puts an asterisk (*) at the end of all string specifications.

To find models with a given description requires a scan of the model file. If you perform this kind of query often and the model file is large, you can significantly improve response time by establishing an index on the description field. ENTER looks to see what indexes exist in order to most efficiently perform the requested query. (For further information, see the Reference Manual, "Secondary Index Maintenance.")

Now type <CTRL> <E> to start the item search. The following screen appears:

```
[sitml00]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Item Maintenance

ITEM NUMBER    :1006
MANUFACTURER   :101      Precision Tool Co.
MODEL          :55271    combination pliers


ADD DATE       :02/15/82
PRICE          :    6.89
ORDER NUMBER   :1
PURCHASE PRICE:    6.00




[N]EXT, [P]REVIOUS, [M]ODIFY, [S]TOP _
searched:      2 selected:      2 current:      1
```

ENTER uses the model-item relationship to visit only the items for the indicated model. Type <ENTER> to display the next record in the set. Then type <S> <ENTER> to clear the screen. Now to perform a range query, find all items that cost between $2.00 and $7.00. Enter the specification as shown next:

```
[sitml00]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Item Maintenance

ITEM NUMBER     :
MANUFACTURER    :
MODEL           :


ADD DATE        :
PRICE           :2.00-7.00_
ORDER NUMBER    :
PURCHASE PRICE:




Begin search [CTRL E], Clear field [CTRL Z], Exit [CTRL X]
```

Type <ENTER> three times or type <CTRL> <E> to complete the specification and the search begins. The following screen appears:

```
+-------------------------------------------------------------------+
|[sitml00]                          UNIFY SYSTEM                    |
|[M]ODIFY                        5 OCT 1982 - 15:25                 |
|                                 Item Maintenance                  |
|                                                                   |
|ITEM NUMBER    :1006                                               |
|MANUFACTURER   :101      Precision Tool Co.                        |
|MODEL          :55271    combination pliers                        |
|                                                                   |
|                                                                   |
|ADD DATE       :02/15/82                                           |
|PRICE          :    6.89                                           |
|ORDER NUMBER   :1                                                  |
|PURCHASE PRICE:     6.00                                           |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|[N]EXT, [P]REVIOUS, [M]ODIFY, [S]TOP                               |
|searched:     14 selected:      6 current:      1                  |
|                                                                   |
+-------------------------------------------------------------------+
```

In this case, ENTER scans the item file to satisfy the request, as the "searched" count of 14 records indicates. Again, if you perform this type of query often and the file is large, indexing the sale price field significantly improves performance speed.

Type <ENTER> a few times to step through the selected records. Now that you know how to use ENTER to display and modify a set of records on the screen, let's produce a report. Type <CTRL> <X> to return immediately to the Main Menu. At the prompt, select 7, System Menu.

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: 7
```

The following screen appears:

```
[sysmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                    System Menu

 1.  Schema Maintenance                9. MENUH Screen Menu

 2.  Schema Listing                    1Ø. MENUH Report Menu

 3.  Create Database                   11. Reconfigure Database

 4.  SFORM Menu                        12. Write Database Backup

 5.  ENTER Screen Registration         13. Read Database Backup

 6.  SQL - Query/DML Language          14. Database Maintenance Menu

 7.  Listing Processor

 8.  Database Test Driver

SELECTION: 5
```

Select ENTER Screen Registration to register a report format with the Item Maintenance screen. The following screen appears:

```
+-------------------------------------------------------------------------+
|[entmnt]                          UNIFY SYSTEM                           |
|                                5 OCT 1982 - 15:25                       |
|                           ENTER Screen Registration                     |
|                                                                         |
|SCREEN NAME:                                                             |
|                                                                         |
|TARGET RECORD:                                                           |
|                                                                         |
|HEADING:                                                                 |
|                                                                         |
|REPORT SCRIPT 1:                                                         |
|              2:                                                         |
|              3:                                                         |
|              4:                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE m                                   |
|                                                                         |
+-------------------------------------------------------------------------+
```

Type <M> <ENTER> to select modify mode. The prompt at the bottom disappears, and the cursor moves to the SCREEN NAME prompt. Respond with sitml00 to call up the Item Maintenance screen. You see the following:

```
[entmnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                          ENTER Screen Registration

SCREEN NAME:    sitml00

TARGET RECORD: item

HEADING:        Item Maintenance

REPORT SCRIPT 1:
              2:
              3:
              4:
```

Type <ENTER> to get to the REPORT SCRIPT 1 prompt. Here you may enter the name of a TRS-XENIX file that contains a report script designed for the target record of the ENTER screen. For this example, enter the names of two report scripts -- one to go to the printer and one to go to the screen. Enter the report script names as follows:

```
[entmnt]                          UNIFY SYSTEM
[M]ODIFY                      5 OCT 1982 - 15:25
                          ENTER Screen Registration

SCREEN NAME:    sitml00

TARGET RECORD: item

HEADING:        Item Maintenance

REPORT SCRIPT 1: itm300.r
              2: itm320.r
              3:
              4:
```

The first script, itm300.r, is the same one created in Section 12 to do the Item Listing. It works here without any modification. The second script is used to call customers when defective items are discovered in an order that was shipped to them. This report lists the item serial number and description and the customer information, including the phone number.

Type <F2> to store the change and clear the responses. Now type <F1> until you get back to the Main Menu. Exit to the shell as before so that you can create the report script itm320.r. (See "Creating Canned Queries" in this tutorial. For further information on the vi text editor, see the Reference Manual, Appendix B.) The text of both files is given below. Note that only reporting commands need be given, since ENTER performs the selection processing.

File: itm3ØØ.r

```
sort DESC imodel.mdes, "MANF NO" imodel momano,
"MODEL NO" imodel_monum end
list "ITEM NO" sno, "MANF NO", "MODEL NO", "ADD DATE"
iad, PRICE isal end
total TOTAL 1 by DESC end
total TOTAL end
print "ITEM LISTING - ITM3ØØ"
```

File: itm32Ø.r

```
sort SERIAL sno end
list column 1 line 1 SERIAL end
list column 3Ø line 1 CUSTOMER iorder.ocust.cname,
     PHONE iorder.ocust.cphone end
list DESCRIPTION imodel.mdes under SERIAL end
list ADDRESS iorder.ocust.caddr under CUSTOMER end
list " " iorder.ocust.ccity, "  " iorder.ocust.cstate
     under ADDRESS end
list column 1 line 4 "   " "   " end
go
```

After you create the itm32Ø.r file, type <CTRL> <D> to return to the menu. Now start up Item Maintenance as follows to try out the reports.

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: 3_
```

The following Item Maintenance screen appears. Type <I> <ENTER> to select inquire mode.

```
+-------------------------------------------------------------------------+
|[sitml00]                          UNIFY SYSTEM                          |
|                                 5 OCT 1982 - 15:25                      |
|                                  Item Maintenance                       |
|                                                                         |
|ITEM NUMBER    :                                                         |
|MANUFACTURER   :                                                         |
|MODEL          :                                                         |
|                                                                         |
|ADD DATE       :                                                         |
|PRICE          :                                                         |
|ORDER NUMBER   :                                                         |
|PURCHASE PRICE:                                                          |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE i_                                  |
|                                                                         |
+-------------------------------------------------------------------------+
```

To try out the Item Listing, select all items in inventory by typing <CTRL> <E> with no selection criteria. The following screen appears:

```
[sitml00]                          UNIFY SYSTEM
[I]NQUIRE                       5 OCT 1982 - 15:25
                                  Item Maintenance

ITEM NUMBER    :1001
MANUFACTURER   :100      RH Smith Manufacturing
MODEL          :1001     1/2" socket wrench


ADD DATE       :02/15/82
PRICE          :     9.75
ORDER NUMBER   :1
PURCHASE PRICE:      7.25









[N]EXT, [P]REVIOUS, [R]EPORT, [S]TOP r_
searched:      14 selected:      14 current:       1
```

Type <R> as shown above to select a report. The following screen appears:

```
+-------------------------------------------------------------------------+
|[sitml00]                          UNIFY SYSTEM                          |
|[I]NQUIRE                       5 OCT 1982 - 15:25                       |
|                                 Item Maintenance                        |
|                                                                         |
|ITEM NUMBER    :1001                                                     |
|MANUFACTURER   :100      RH Smith Manufacturing                          |
|MODEL          :1001     1/2" socket wrench                              |
|                                                                         |
|                                                                         |
|ADD DATE       :02/15/82                                                 |
|PRICE          :    9.75                                                 |
|ORDER NUMBER   :1                                                        |
|PURCHASE PRICE:     7.25                                                 |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|searched:     14 selected:     14 current:      1                       |
|1. itm300.r  2. itm320.r   1_                                            |
+-------------------------------------------------------------------------+
```

You can select either report by entering the appropriate number. If only one report is associated with the current screen, then type <R> as above to start it up immediately. Type <1> as shown to start the item listing to the printer. The following screen appears:

```
+-------------------------------------------------------------------+
| [Report running]                                                  |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
| Report complete ->->_                                             |
+-------------------------------------------------------------------+
```

The prompt appears at the bottom of the screen when the report is ready to be printed. This report looks exactly like the ITM3ØØ report of Section 12. Type <ENTER>, and the screen is redisplayed as shown next.

```
+-----------------------------------------------------------------------+
|[sitml00]                          UNIFY SYSTEM                        |
|[I]NQUIRE                        5 OCT 1982 - 15:25                    |
|                                  Item Maintenance                     |
|                                                                       |
|ITEM NUMBER    :1001                                                   |
|MANUFACTURER   :100      RH Smith Manufacturing                        |
|MODEL          :1001     1/2" socket wrench                            |
|                                                                       |
|                                                                       |
|ADD DATE       :02/15/82                                               |
|PRICE          :    9.75                                               |
|ORDER NUMBER   :1                                                      |
|PURCHASE PRICE:     7.25                                               |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|[N]EXT, [P]REVIOUS, [R]EPORT, [S]TOP r_                                |
|searched:     14 selected:     14 current:      1                      |
+-----------------------------------------------------------------------+
```

Type <R> <ENTER> again, and then select 2 to see the screen report.

```
[sitml00]                          UNIFY SYSTEM
[I]NQUIRE                       5 OCT 1982 - 15:25
                                 Item Maintenance

ITEM NUMBER    :1001
MANUFACTURER   :100      RH Smith Manufacturing
MODEL          :1001     1/2" socket wrench


ADD DATE       :02/15/82
PRICE          :    9.75
ORDER NUMBER   :1
PURCHASE PRICE:    7.25









searched:     14 selected:     14 current:      1
1. itm300.r  2. itm320.r   2_
```

The report appears on the screen as follows:

```
[Report running]
   SERIAL                    CUSTOMER                    PHONE
DESCRIPTION                  ADDRESS

-------------------------------------------------------------------

     1ØØ1                    Smith & Sons Hardware       916-924-ØØ75
1/2" socket wrench           1234 State Street
                             Wheatville          CA

     1ØØ2                    Creative Manufacturing      916-91Ø-9999
1/2" socket wrench           9124 Industrial Blvd.
                             Redding             CA

     1ØØ3                    Smith & Sons Hardware       916-924-ØØ75
1/2" socket wrench           1234 State Street
                             Wheatville          CA

     1ØØ4                    Smith & Sons Hardware       916-924-ØØ75
1/2" socket wrench           1234 State Street
                             Wheatville          CA

Continue? y_
```

This is the first page of the report. You can continue by typing <Y> <ENTER> as shown above. The second page appears as follows:

```
+-------------------------------------------------------------------+
|    SERIAL                    CUSTOMER                    PHONE    |
| DESCRIPTION                  ADDRESS                               |
|-------------------------------------------------------------------|
|                                                                   |
|     1ØØ5                     Creative Manufacturing      916-91Ø-9999
| 1/2" socket wrench           9124 Industrial Blvd.                |
|                              Redding              CA               |
|                                                                   |
|     1ØØ6                     Smith & Sons Hardware       916-924-ØØ75
| combination pliers           1234 State Street                    |
|                              Wheatville           CA              |
|                                                                   |
|     1ØØ7                     Smith & Sons Hardware       916-924-ØØ75
| leather mallet               1234 State Street                    |
|                              Wheatville           CA              |
|                                                                   |
|     1ØØ8                     Reliable Construction Co.   916-945-3434
| 3" Phillips screwdriver      2113 Folsom Blvd.                    |
|                              Sacramento           CA              |
|                                                                   |
|Continue? n_                                                       |
+-------------------------------------------------------------------+
```

Type <N> <ENTER> as shown above, and type <ENTER> to the Report complete >> prompt. The Item Maintenance screen is redisplayed.

```
+-------------------------------------------------------------------------+
|[sitml00]                          UNIFY SYSTEM                          |
|[I]NQUIRE                       5 OCT 1982 - 15:25                       |
|                                 Item Maintenance                        |
|                                                                         |
|ITEM NUMBER    :1001                                                     |
|MANUFACTURER   :100      RH Smith Manufacturing                          |
|MODEL          :1001     1/2" socket wrench                              |
|                                                                         |
|                                                                         |
|ADD DATE       :02/15/82                                                 |
|PRICE          :    9.75                                                 |
|ORDER NUMBER   :1                                                        |
|PURCHASE PRICE:     7.25                                                 |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|[N]EXT, [P]REVIOUS, [R]EPORT, [S]TOP _                                   |
|searched:      14 selected:      14 current:       1                     |
+-------------------------------------------------------------------------+
```

Now type <CTRL> <X> to return to the current menu.

Chapter 15
ENTERING HELP DOCUMENTATION

You can enter help documentation to describe any menu, program, or ENTER screen by using the Enter Help Documentation function, enthdoc. To illustrate this, enter a description of the Item Listing report created earlier. Select the program as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: enthdoc_
```

The following screen appears:

```
[enthdoc]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                              Enter Help Documentation

MENU/PROGRAM: itm3ØØ_
```

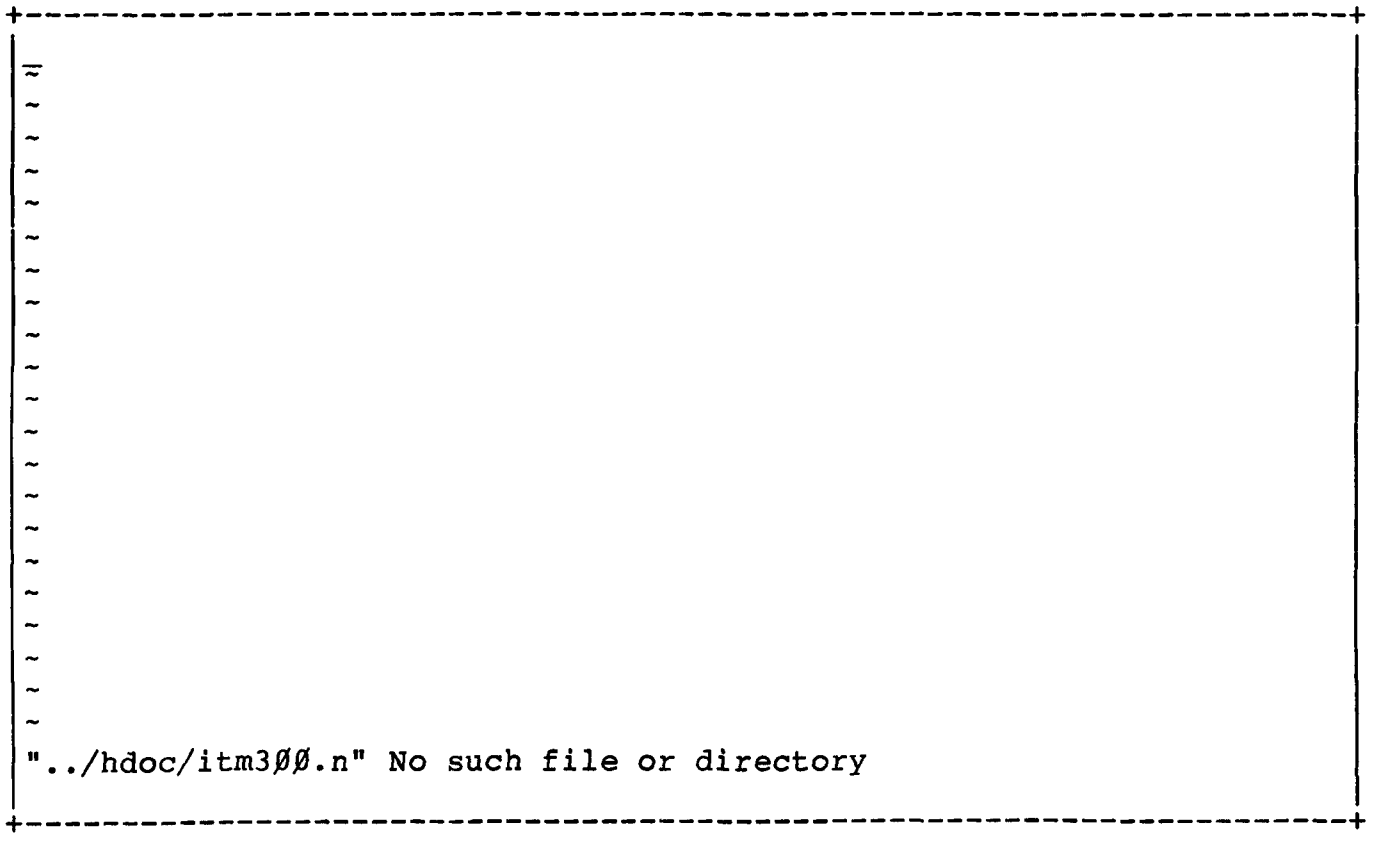
The meaning of the prompt is as follows.

MENU/PROGRAM

Name of a menu, program, or ENTER screen. Enthdoc starts vi, the text editor, with the file name "../hdoc/xxxxxxxx.n", where xxxxxxxx is the name you entered. You can specify a different text editor by using the environment variable EDIT. (For more information, consult the Reference Manual.)

When you exit the editor, the screen displays the prompt above, and you can enter a new file name. Type <F1> to return to the menu. Enter the program name itm3ØØ in response to the prompt, and the

following screen appears. The particular prompts you see depend on the editor you specify. This screen illustrates what vi does.



Type <I> to insert text and then enter the following text to describe the report:

```
Item Listing

This report lists items in inventory, sorted by description.
There is a subtotal for each type of item, and a total count
of items at the end.
The selection options are a range of dates during which the items
were added to inventory.
~
~
~
~
~
~
~
~
~
~
~
~
~
"../hdoc/itm3ØØ.n" [New file] 8 lines, 253 characters
```

Write the helpfile and exit the editor by typing <ESC>:wq, and the initial screen appears. Type <F1> to return to the Main·Menu, and select help as follows:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: help 4_
```

The screen shows the help documentation entered previously.

```
+-------------------------------------------------------------------+
|                                                                   |
|                                                                   |
|                                                                   |
| Item Listing                                                      |
|                                                                   |
| This report lists items in inventory, sorted by description.      |
| There is a subtotal for each type of item, and a total count      |
| of items at the end.  The selection options are a range of        |
| dates during which the items were added to inventory.             |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
| --> --> _                                                         |
+-------------------------------------------------------------------+
```

Type <ENTER> in response to the --> --> prompt, and the screen displays the menu. You can also obtain help by typing

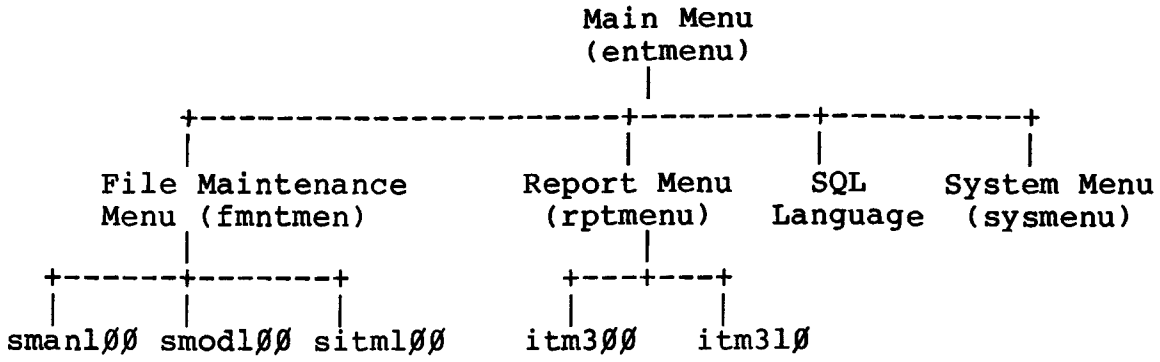
help menu or program name

in response to the menu SELECTION prompt. Try the following response to bring up the documentation just entered:

```
[entmenu]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                    Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: help itm3ØØ_
```

Chapter 16
MODIFYING MENUS

You now have two distinct sets of programs -- the file maintenance programs (smanlØØ, smodlØØ, and sitmlØØ) and the report programs (itm3ØØ and itm31Ø). Let's modify the current single-level menu structure to produce a hierarchical menu tree. The structure is as follows:



Select Menu Maintenance (menumnt) as follows:

```
[entmenu]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                    Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: menumnt_
```

The menu maintenance screen appears. To add two new menus, select add mode as follows:

```
[menumnt]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                  Menu Maintenance

 NAME:              HEADING:
 CMD   LINE   MENU/PROG   M/P    PROMPT









[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE a_
```

Then add the File Maintenance Menu, fmntmen. You must select ADD mode after entering the heading in order to enter the **ENTER** screens, one at a time. The screen appears as follows when you are finished:

```
[menumnt]                          UNIFY SYSTEM
[A]DD                            5 OCT 1982 - 15:25
                                  Menu Maintenance

 NAME: fmntmen  HEADING: File Maintenance Menu
 CMD  LINE  MENU/PROG  M/P   PROMPT
        1   smanlØØ     E    Manufacturer Maintenance
        2   smodlØØ     E    Model Maintenance
        3   sitmlØØ     E    Item Maintenance
  q_
```

Type <Q> <ENTER> and then type <F1> to clear the screen in preparation for adding the Report Menu, rptmenu. Add this menu as shown.

```
[menumnt]                          UNIFY SYSTEM
[A]DD                          5 OCT 1982 - 15:25
                                Menu Maintenance

 NAME: rptmenu  HEADING: Report Menu
 CMD  LINE  MENU/PROG  M/P   PROMPT
        1   itm3ØØ      P    Item Listing
        2   itm31Ø      P    Inventory Aging Report
  q_
```

Type <Q> <ENTER> and then type <F1> twice to get back to the mode prompt. Select modify mode as shown next.

```
[menumnt]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                 Menu Maintenance

 NAME:              HEADING:
 CMD   LINE   MENU/PROG   M/P    PROMPT









[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE m_
```

Now call up entmenu to make the required changes.

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                      5 OCT 1982 - 15:25
                               Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD  LINE  MENU/PROG  M/P   PROMPT
  d_    1   sman1ØØ     E    Manufacturer Maintenance
        2   smod1ØØ     E    Model Maintenance
        3   sitm1ØØ     E    Item Maintenance
        4   itm3ØØ      P    Item Listing
        5   itm31Ø      P    Inventory Aging Report
        6   sql         P    SQL - Query/DML Language
        7   sysmenu     M    System Menu
```

Delete Line 1 of the menu by typing <D> <ENTER> in the CMD column. Then delete two more lines the same way for a total of three lines deleted. The screen then appears as follows:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD  LINE  MENU/PROG  M/P   PROMPT
  m_    1   itm3ØØ      P    Item Listing
        2   itm31Ø      P    Inventory Aging Report
        3   sql         P    SQL - Query/DML Language
        4   sysmenu     M    System Menu
```

Now modify the first line by typing <M> <ENTER> in the CMD column, and then change MENU/PROG to fmntmen. The cursor is now in the LINE column. Type fmntmen <ENTER> <F2>, and the screen appears as follows:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD   LINE   MENU/PROG   M/P    PROMPT
         1    fmntmen      M     File Maintenance Menu
  _      2    itm31Ø       P     Inventory Aging Report
         3    sql          P     SQL - Query/DML Language
         4    sysmenu      M     System Menu
```

Now change Line 2 to rptmenu. The final configuration is as follows:

```
[menumnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                                 Menu Maintenance

 NAME: entmenu  HEADING: Main Menu
 CMD  LINE  MENU/PROG  M/P   PROMPT
        1   fmntmen     M    File Maintenance Menu
        2   rptmenu     M    Report Menu
  q_    3   sql         P    SQL - Query/DML Language
        4   sysmenu     M    System Menu
```

Now type <Q> as shown and then type <F1> three times to return to the Main Menu. Note that the menu displayed does not reflect the modifications, since it is displayed from a copy in memory.

You must now update the user's security privileges. Since the data entry clerks only have access to the programs on the File Maintenance Menu, use Group Maintenance (grpmnt) to change their system entry point to fmntmen. Select Group Maintenance as follows:

```
+-------------------------------------------------------------------+
|[entmenu]                        UNIFY SYSTEM                      |
|                               5 OCT 1982 - 15:25                  |
|                                  Main Menu                        |
|                                                                   |
|1. Manufacturer Maintenance                                        |
|                                                                   |
|2. Model Maintenance                                               |
|                                                                   |
|3. Item Maintenance                                                |
|                                                                   |
|4. Item Listing                                                    |
|                                                                   |
|5. Inventory Aging Report                                          |
|                                                                   |
|6. SQL - Query/DML Language                                        |
|                                                                   |
|7. System Menu                                                     |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: grpmnt_                                                 |
|                                                                   |
+-------------------------------------------------------------------+
```

Select modify mode as shown.

```
[grpmnt]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                                 Group Maintenance

GROUP ID:          NAME:

SYSTEM ENTRY PT:

ACCESS LEVELS:
 LN  MENU/PROG M/P INQ ADD MOD DEL  LN  MENU/PROG M/P INQ ADD MOD DEL




[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE m_
```

Type DEC <ENTER> at the GROUP ID prompt to retrieve the clerk group, and then type <ENTER> until you are at the SYSTEM ENTRY PT prompt. Change it to fmntmen as shown.

```
[grpmnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                                Group Maintenance

GROUP ID: DEC    NAME: Data Entry Clerks

SYSTEM ENTRY PT: fmntmen  - M

ACCESS LEVELS:
 LN  MENU/PROG M/P INQ ADD MOD DEL  LN  MENU/PROG M/P INQ ADD MOD DEL

  1  smanlØØ    E   y   y   y   n
  2  smodlØØ    E   y   y   y   n
  3  sitmlØØ    E   y   y   y   n

[N]ext page, [P]rev page, [A]dd line, or number: _
```

Return to the menu by typing <F1>, and select Employee Maintenance (empmnt) as follows. You can now add the two new menus, fmntmen and rptmenu, to the supervisor's privileges and change the supervisor's system entry point to entmenu.

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. Manufacturer Maintenance

2. Model Maintenance

3. Item Maintenance

4. Item Listing

5. Inventory Aging Report

6. SQL - Query/DML Language

7. System Menu



SELECTION: empmnt_
```

Select modify mode as shown.

```
+-------------------------------------------------------------------------+
|[empmnt]                          UNIFY SYSTEM                           |
|                                5 OCT 1982 - 15:25                       |
|                              Employee Maintenance                       |
|                                                                         |
|LOGIN ID:       NAME:                                                    |
|GROUP ID:       NAME:                                                    |
|                    SYSTEM ENTRY PT:                                     |
|                                                                         |
|ACCESS LEVELS DIFFERENT FROM GROUP:                                      |
|  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL                            |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE m_                                  |
|                                                                         |
+-------------------------------------------------------------------------+
```

Type sss at the LOGIN ID prompt to retrieve the supervisor. Change the SYSTEM ENTRY PT to entmenu, and then select add mode as shown to add the two new menus.

```
[empmnt]                          UNIFY SYSTEM
[M]ODIFY                       5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                     SYSTEM ENTRY PT: entmenu  - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL


   1  sman1ØØ    E     y       y   y   y   y
   2  smod1ØØ    E     y       y   y   y   y
   3  sitm1ØØ    E     y       y   y   y   y
   4  sql        P     y
   5  itm11Ø     E     y       y   y   y   y
   6  itm3ØØ     P     y
   7  itm31Ø     P     y


[N]ext page, [P]rev page, [A]dd line, or number: a_
```

Add the two new menus, fmntmen and rptmenu, to the list of access privileges. When finished, the screen appears as follows:

```
[empmnt]                          UNIFY SYSTEM
[M]ODIFY                        5 OCT 1982 - 15:25
                              Employee Maintenance

LOGIN ID: sss  NAME: Samuel S. Spade
GROUP ID: DEC  NAME: Data Entry Clerks
                    SYSTEM ENTRY PT: entmenu    - M

ACCESS LEVELS DIFFERENT FROM GROUP:
  LN  MENU/PROG M/P  ACCESS?  INQ ADD MOD DEL


   1  smanlØØ    E     y       y   y   y   y
   2  smodlØØ    E     y       y   y   y   y
   3  sitmlØØ    E     y       y   y   y   y
   4  sql        P     y
   5  itmllØ     E     y       y   y   y   y
   6  itm3ØØ     P     y
   7  itm3lØ     P     y
   8  fmntmen    M     y
   9  rptmenu    M     y
```

Now type <F1> repeatedly to return to the TRS-XENIX prompt and see the results of these changes. Log in as the supervisor by typing **login sss.** Then reexecute the Unify tutorial by typing **ututorial** <ENTER>. The following menu appears:

```
+-------------------------------------------------------------------+
|[entmenu]                        UNIFY SYSTEM                      |
|                              5 OCT 1982 - 15:25                   |
|                                  Main Menu                        |
|                                                                   |
|1. File Maintenance Menu                                           |
|                                                                   |
|2. Report Menu                                                     |
|                                                                   |
|3. SQL - Query/DML Language                                        |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: _                                                       |
+-------------------------------------------------------------------+
```

Type <F1> to return to the system prompt. Log in as the data entry clerk, **jhd**, and reexecute the Unify tutorial to see the effects of the changes to the group. The following menu appears:

```
+-------------------------------------------------------------------+
|[fmntmen]                          UNIFY SYSTEM                    |
|                                 5 OCT 1982 - 15:25                |
|                               File Maintenance Menu               |
|                                                                   |
|1. Manufacturer Maintenance                                        |
|                                                                   |
|2. Model Maintenance                                               |
|                                                                   |
|3. Item Maintenance                                                |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: _                                                       |
+-------------------------------------------------------------------+
```

Now go back out one more time, log in as root, and once again reexecute **ututorial.** The following menu appears:

```
[entmenu]                          UNIFY SYSTEM
                                 5 OCT 1982 - 15:25
                                     Main Menu

1. File Maintenance Menu

2. Report Menu

3. SQL - Query/DML Language

4. System Menu









SELECTION: parmnt_
```

You can change the prompt that appears at the top of each screen and report to one more descriptive of your particular system. The standard configuration comes with the prompt set to UNIFY SYSTEM. To change it, select System Parameter Maintenance (parmnt) from the system menu as above. The following screen appears:

```
+-------------------------------------------------------------------+
|[parmnt]                          UNIFY SYSTEM                     |
|                                5 OCT 1982 - 15:25                 |
|                          System Parameter Maintenance             |
|SUPER USER ID   : su                                               |
|ENTRY POINT     : entmenu                                          |
|SYSTEM HEADING  : UNIFY SYSTEM                                     |
|LANGUAGE        : EN                                               |
|MONTH MNEMONICS: JANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC              |
|BLOCKS / VOLUME: 849                                               |
|                                                                   |
|                                                                   |
|                                                                   |
+-------------------------------------------------------------------+
```

The above entries are system parameters that you can change.

SUPER USER ID

Login id of the super user. It can be any 8-character string.

ENTRY POINT

Name of the menu the super user sees after logging in.

SYSTEM HEADING

A 5Ø-character string that is displayed on Line 1 of all screens and menus in the application system.

LANGUAGE

A 2-character code that MENUH makes accessible to host language programs for testing the language character set used.

MONTH MNEMONICS

A 36-character string, containing 3 characters for each month of the year. This string is used to display Line 2 on all screens and menus in the application system.

BLOCKS / VOLUME

An integer that tells Write and Read Database Backup and Reconfigure Database how many 4k (4Ø96 bytes) blocks fit on a volume of the system backup device. The system backup device is a diskette.

The cursor is initially at the SUPER USER ID prompt. <ENTER> moves the cursor down the screen; <F1> moves it up. Typing <F1> at the first prompt returns you to the menu. To change an entry, type the new data and <ENTER>.

Change the system heading to WAREHOUSE INVENTORY SYSTEM as follows:

```
[parmnt]                          UNIFY SYSTEM
                                5 OCT 1982 - 15:25
                            System Parameter Maintenance

SUPER USER ID   : su

ENTRY POINT     : entmenu

SYSTEM HEADING  : WAREHOUSE INVENTORY SYSTEM

LANGUAGE        : EN

MONTH MNEMONICS: JANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC

BLOCKS / VOLUME: 849
```

The system parameters are stored in memory when you start up Unify, so you cannot see changes until you exit and reenter Unify. Type <F1> repeatedly to exit to the shell, and reexecute **ututorial** from root.

```
+-------------------------------------------------------------------+
|[entmenu]                WAREHOUSE INVENTORY SYSTEM                |
|                            5 OCT 1982 - 15:27                     |
|                                Main Menu                          |
|                                                                   |
|1. File Maintenance Menu                                           |
|                                                                   |
|2. Report Menu                                                     |
|                                                                   |
|3. SQL - Query/DML Language                                        |
|                                                                   |
|4. System Menu                                                     |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: parmnt_                                                 |
|                                                                   |
+-------------------------------------------------------------------+
```

Chapter 17
WRITING C PROGRAMS

Although you can implement many programs in an application system with ENTER and SQL, frequently you need to tailor data entry programs for operator efficiency. For example, consider the job of order entry. You can create an ENTER screen to allow entry of a single item at a time on an order, but that is very slow. A more efficient method is to have an order form with space for the customer, several items, and total order amount. You can create data entry programs such as this quickly and easily using the applications library functions supplied with Unify.

This example uses the programming conventions established for Unify. Since the program deals mainly with order records, it is called ord1ØØ and resides in the executable file ORD1ØØ. The source code is kept in the directory src/ord, which contains an archive named ord1ØØ.a. You build the program in a top-down, structured manner, with a single C function per source file. It is not the intent of this tutorial to teach C programming, so the source code is presented without comment in Appendix A.

The first step in designing a data entry program is to design the screen. Here is the screen form design for order entry, followed by a functional description of the program to drive it.

Note: You must be operating with the TRS-XENIX Development System in order to do the examples in this chapter.

```
+------------------------------------------------------------------------+
|[ordl00]              WAREHOUSE INVENTORY SYSTEM                         |
|                         5 OCT 1982 - 15:25                              |
|                         Order Maintenance                               |
|                                                                         |
| ORDER NO:'''''''''                                        DATE:''''''''|
| CUSTOMER:'''''                                                          |
|         :''''''''''''''''''''''''''''''                                 |
|         :''''''''''''''''''''''''''''''                                 |
|         :'''''''''''''''''''' '' '''''                                  |
|                                                                         |
| CMD ITEM NO   MODEL NO DESCRIPTION                          PRICE      |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|  "   """"""""""" """""""" """""""""""""""""""""""""""""""""" """"".""  |
|                                                                         |
|                                              TOTAL====>"""""".""      |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE: _                                 |
|                                                                         |
+------------------------------------------------------------------------+

     '''''''' => order data entry area
     """""""" => line item data entry area
```

This screen lets you inquire, add, modify, and delete orders and associated line items. There is a data entry area for orders and a multiline data entry area for line items.

[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE

This prompt lets you choose an operational mode for the order data entry area of the screen.

I - Inquire, lets you see the line items for an existing order.

A - Add, lets you add new orders and associated line items.

M - Modify, lets you add, modify, and delete line items for an existing order.

D - Delete, lets you delete an existing order and all its line items.

ORDER NO

A 9-digit order number that uniquely identifies the order. This is the primary key of the record.

DATE

Date the order was placed.

CUSTOMER

A 5-digit customer id number that identifies the customer for this order. The next three lines are display-only areas that list the customer name, street address, city, state, and zip code to assist the user in making sure the customer id number is correct.

CMD

You use this area to enter a command to perform an operation on the current line item. Use it in the same way as you use the CMD column on other Unify screens. The cursor moves up and down in the multiline data entry area in response to <F1> and <ENTER>. The position of the cursor marks the current line. To simplify the programming, only nine line items are allowed on a single order. Of course, you can add any number of line items with the appropriate coding. The following are valid entries:

a - Add a new line item to the order. The cursor moves to the first empty line to let you enter an item number. This response is only valid if there are fewer than nine items on the order already.

d - Delete the current line item from the order. This response is only valid if the current line contains an item.

q - Leave the multiline data entry area and position the cursor at the DATE prompt.

ITEM NO

A 9-digit serial number of an item in inventory that is not already on an order. If the item is already on an order, the screen displays an error message.

MODEL NO

A display-only field that shows the model number of the current item.

DESCRIPTION

A display-only field that shows the description of the current item.

PRICE

A display-only field that shows the sale price of the current item.

TOTAL

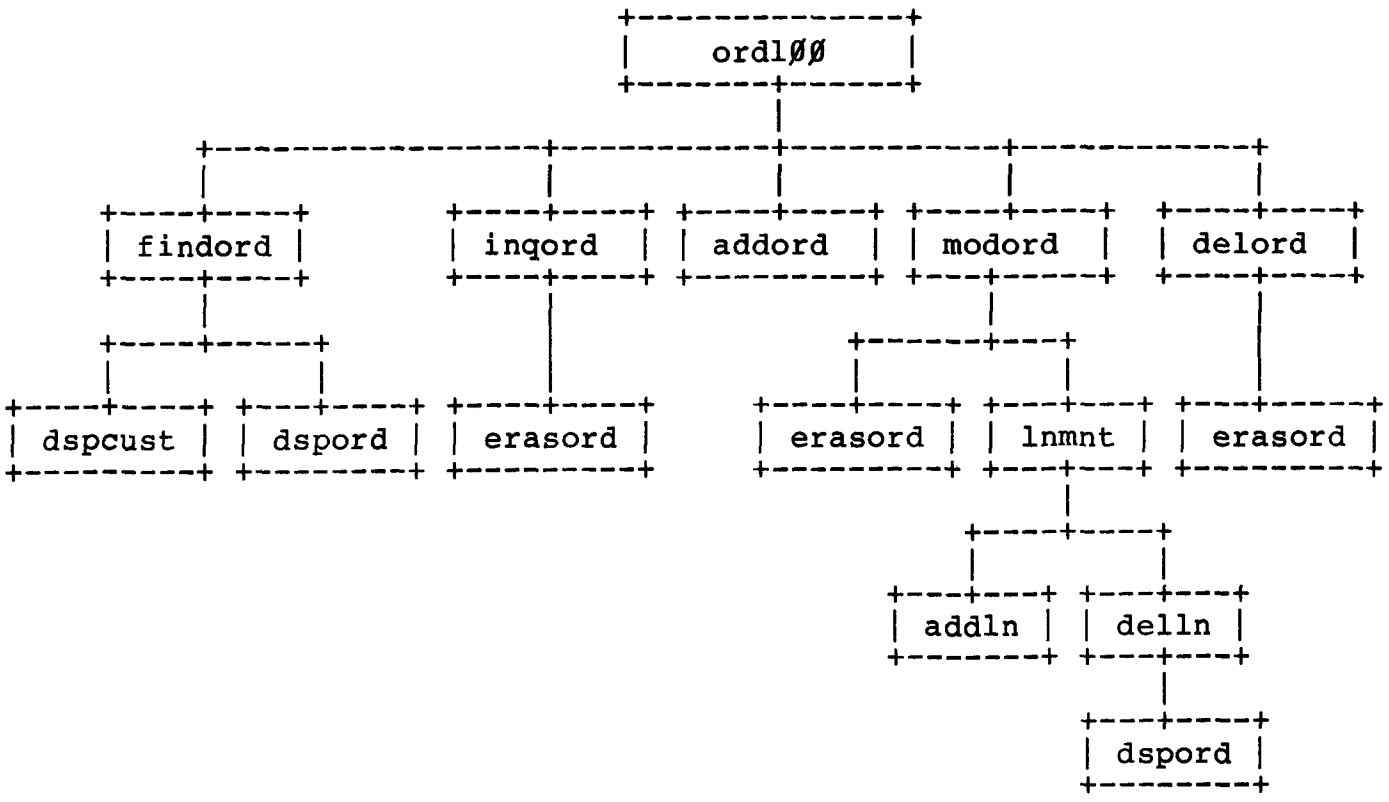
A display-only field that shows the sum of all the sale prices for items on the order.

The program operates as follows. An in-core table keeps the addresses of all item records for an order, to simplify the

manipulation of the records. For this reason, you must lock the current order record so that only one user can work with an order at a time. The database must contain a dummy order number Ø so that item records can be on order Ø if they are not on a real order.

You use both the SFORM runtime functions, in the upper part of the screen, and the database terminal I/O functions, in the line item area of the screen. These functions work together to create customized programs. You also use the function **priamd** to interface with the MENUH security capability.

In addition to the above Unify functions, the example program, listed in Appendix B of this tutorial, contains twelve functions. Following is a chart that shows what functions call each other.



The following list, describing the functions, is in alphabetical order.

addln - adds new items to the current order. If the item is already on an order, the add is not allowed. The total amount is updated to reflect the new item.

addord - adds new order records to the database. It gets the key of the order from the user.

delln - deletes the current item from the order. This function does not actually delete the record but sets the relational field iorder to Ø. This requires the "dummy" record with order number Ø. The total amount on the screen is updated.

delord - deletes the current order record from the database. It first asks for confirmation from the user and then sets all the item record's iorder field to Ø before deleting the order.

dspcust - displays data on the top half of the order screen.

dspord - displays all items for the current order in the item area of the screen. The total amount is calculated and displayed also.

erasord - erases data from the entire screen.

findord - accepts an order number from the user and looks up the record. If it finds the order, it fills in the table of item record addresses and displays the data on the screen.

inqord - waits for the user to finish looking at the order and then clears the screen.

lnmnt - controls the line item area of the screen. The user is prompted to add or delete items on the order.

modord - permits the user to modify data on the current order. It allows changing the date, customer number, and items on the order.

ordlØØ - the control function and entry point for the program. It gets the operational mode from the user and calls the appropriate routine to perform the function.

In addition to the functions described above, the order entry program includes the database defined file, file.h, which is created when the database is reconfigured in Chapter 9, and a file named ordlØØ.h that has been provided under the directory tutorial/def and includes the global data and defines for the program.

To get the program running, perform the following steps. Use SFORM to enter and process the screen form, sordlØØ. Start up Screen Entry as follows:

```
[entmenu]              WAREHOUSE INVENTORY SYSTEM
                           5 OCT 1982 - 15:25
                               Main Menu

1. File Maintenance Menu

2. Report Menu

3. SQL - Query/DML Language

4. System Menu




SELECTION: sfmaint_
```

The following screen appears:

```
+-----------------------------------------------------------------------+
|[sfmaint]              WAREHOUSE INVENTORY SYSTEM                      |
|                           5 OCT 1982 - 15:25                          |
|                              Screen Entry                             |
|                                                                       |
|SCREEN:                                                                |
|                                                                       |
|LN SFIELD   DFIELD  TP LEN FX FY PROMPT                          PX PY |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|                                                                       |
|[I]NQUIRE, [A]DD, [M]ODIFY, [D]ELETE: a_                               |
+-----------------------------------------------------------------------+
```

Select add mode as shown above, and then enter the following screen form for the order maintenance program.

SCREEN: sordlØØ

SFIELD	DFIELD	TYPE	LEN	FX	FY	PROMPT	PX	PY
sonum	onum	N	9	1Ø	4	ORDER NO:	1	4
sodate	odate	D	2	59	4	DATE:	54	4
socust	ocust	N	5	1Ø	5	CUSTOMER:	1	5
scname	cname	S	3Ø	1Ø	6	:	1	6
scaddr	caddr	S	3Ø	1Ø	7	:	1	7
sccity	ccity	S	2Ø	1Ø	8	:	1	8
scstate	cstate	S	2	31	8		31	8
sczip	czip	N	5	34	8		34	8
samd		S	1	2	11	CMD ITEM NO MODEL NO DESCRIPTION	1	1Ø
samdl			Ø	54	11	PRICE	57	1Ø
stotal		A	6	53	21	TOTAL====>	43	21

Now type <F1> to return to the menu and then select Test Screen as shown next to verify that the design is entered correctly.

```
[entmenu]                WAREHOUSE INVENTORY SYSTEM
                              5 OCT 1982 - 15:25
                                  Main Menu

1. File Maintenance Menu

2. Report Menu

3. SQL - Query/DML Language

4. System Menu




SELECTION: sfsamp_
```

```
+-------------------------------------------------------------------+
|[sfsamp]                WAREHOUSE INVENTORY SYSTEM                 |
|                            5 OCT 1982 - 15:25                     |
|                               Test Screen                         |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SCREEN: sordlØØ_                                                   |
+-------------------------------------------------------------------+
```

Enter sordlØØ as shown above to display the screen form just entered. The following display appears:

```
+-------------------------------------------------------------------------+
|[sfsamp]                WAREHOUSE INVENTORY SYSTEM                       |
|                            5 OCT 1982 - 15:25                           |
|                               Test Screen                               |
|                                                                         |
| ORDER NO:x                                                 DATE:x       |
| CUSTOMER:x                                                              |
|         :x                                                              |
|         :x                                                              |
|         :x                      x  x                                    |
|                                                                         |
| CMD ITEM NO   MODEL NO DESCRIPTION                            PRICE     |
|  x                                                            x         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                TOTAL====>x              |
|                                                                         |
|SCREEN: _                                                      sordlØØ   |
+-------------------------------------------------------------------------+
```

Now type <F1> to return to the menu and process the screen by selecting Process Screen as shown next.

```
+-------------------------------------------------------------------+
|[entmenu]              WAREHOUSE INVENTORY SYSTEM                  |
|                           5 OCT 1982 - 15:25                      |
|                               Main Menu                           |
|                                                                   |
|1. File Maintenance Menu                                           |
|                                                                   |
|2. Report Menu                                                     |
|                                                                   |
|3. SQL - Query/DML Language                                        |
|                                                                   |
|4. System Menu                                                     |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: sfproc_                                                 |
|                                                                   |
+-------------------------------------------------------------------+
```

The following screen appears:

```
+-------------------------------------------------------------------+
|[sfproc]                WAREHOUSE INVENTORY SYSTEM                 |
|                             5 OCT 1982 - 15:25                    |
|                               Process Screen                      |
|                                                                   |
|SCREEN: sordlØØ_                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
+-------------------------------------------------------------------+
```

Enter sordlØØ as shown above to process the screen. This creates the file sordlØØ.q in the tutorial/bin directory and the file sordlØØ.h in the tutorial/def directory. The .q file contains a binary screen description that the program reads at run time. The .h file contains the screen field names that are used in the source code of the program.

To compile the source code for ordlØØ, exit to the shell as follows:

```
+-----------------------------------------------------------------+
|[entmenu]              WAREHOUSE INVENTORY SYSTEM                |
|                           5 OCT 1982 - 15:25                    |
|                               Main Menu                         |
|                                                                 |
|1. File Maintenance Menu                                         |
|                                                                 |
|2. Report Menu                                                   |
|                                                                 |
|3. SQL - Query/DML Language                                      |
|                                                                 |
|4. System Menu                                                   |
|                                                                 |
|                                                                 |
|                                                                 |
|                                                                 |
|                                                                 |
|                                                                 |
|SELECTION: sh_                                                   |
|                                                                 |
+-----------------------------------------------------------------+
```

You are now in the tutorial/bin directory and can use the normal shell commands. The source code for each of the ordlØØ program functions is located within the tutorial/src/ord directory. Move this directory by entering:

```
cd ../scr/ord
```

Then compile the source as follows:

```
ucc -c -Ø *.c
```

Create an archive named ordlØØ.a by entering ar q ordlØØ.a and the list of the object files shown below:

```
ordlØØ.o
addord.o
delord.o
findord.o
dspcust.o
inqord.o
modord.o
erasord.o
lnmnt.o
addln.o
delln.o
dspord.o
```

Note: Do not press <ENTER> until all files have been entered.

The ordering of program functions within the archive can be verified by typing **lorder ordlØØ.a**. (For more information on **ar** and **lorder** functions, refer to AR(1S) and LORDER(1S) in your TRS-XENIX System Reference Manual.)

After you create the archive, reenter Unify by typing <CTRL> <D>. Then use Executable Maintenance (execmnt) to add the executable ORDlØØ. Start up execmnt as shown:

```
+-------------------------------------------------------------------+
|[entmenu]              WAREHOUSE INVENTORY SYSTEM                  |
|                            5 OCT 1982 - 15:25                     |
|                                Main Menu                          |
|                                                                   |
|1. File Maintenance Menu                                           |
|                                                                   |
|2. Report Menu                                                     |
|                                                                   |
|3. SQL - Query/DML Language                                        |
|                                                                   |
|4. System Menu                                                     |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: execmnt_                                                |
+-------------------------------------------------------------------+
```

Your screen appears as follows:

```
[execmnt]                WAREHOUSE INVENTORY SYSTEM
[A]DD                         5 OCT 1982 - 15:25
                           Executable Maintenance

EXECUTABLE'S NAME: ORD1ØØ
USES SYSRECEV ? Y
PROGRAMS:
CMD NAME       HEADING                          SCREEN    DIRECTORY
  Ø ordlØØ     Orders Maintenance               sordlØØ   ord
  1
  2
  3
  4
  5
  6
  7
  8
  9
```

Now you can load the program using the Program Loading utility (lfilegen). Type <F1> to return to the menu and then select Program Loading as shown following:

```
+-------------------------------------------------------------------+
|[entmenu]              WAREHOUSE INVENTORY SYSTEM                  |
|                            5 OCT 1982 - 15:25                     |
|                                Main Menu                          |
|                                                                   |
|1. File Maintenance Menu                                           |
|                                                                   |
|2. Report Menu                                                     |
|                                                                   |
|3. SQL - Query/DML Language                                        |
|                                                                   |
|4. System Menu                                                     |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|                                                                   |
|SELECTION: lfilegen_                                               |
+-------------------------------------------------------------------+
```

Enter the program name, ordlØØ, in response to the prompt when the screen is displayed. The screen appears as follows when the program is successfully loaded:

```
[lfilegen]              WAREHOUSE INVENTORY SYSTEM
                            5 OCT 1982 - 15:25
                             Program Loading

 PROGRAM NAME: ordlØØ


 load ../bin/ORDlØØ
 xxxx+xxxx+xxxx= xxxxxb = xxxxxxxb
 ../bin/ORDlØØ loaded as ../bin/ORDlØØ
```

In the process of program loading, conflicts may result between TRS-XENIX and Unify symbol names. These problems appear as warning messages about the multiply defined symbols. In general, you can ignore these messages, because the symbol names used by Unify are the correct names to load.

Another warning message may appear during this process when the modification date of a TRS-XENIX library archive is more recent than the date of its associated dictionary. This can occur if a library archive required by the program loading process has been copied since its dictionary was created. The resulting out of date warning message is of little consequence and may be ignored. If you want to remedy the

situation, you may execute the function **ranlib** described in the TRS-XENIX Reference Manual under RANLIB(1S).

When program loading is complete, type <F1> to return to the menu. Now you can add access privileges for ordlØØ and put it on the menus you desire. Here is a sample order screen with data on it:

```
[ordlØØ]                WAREHOUSE INVENTORY SYSTEM
[A]DD                        5 OCT 1982 - 15:25
                             Orders Maintenance

ORDER NO:1                                              DATE:2/18/82
CUSTOMER:1
        :Smith & Sons Hardware
        :1234 State Street
        :Wheatville              CA 95817

CMD  ITEM NO     MODEL NO  DESCRIPTION                      PRICE
     1ØØ1        1ØØ1      1/2" socket wrench                9.75
     1ØØ2        1ØØ1      1/2" socket wrench                9.75
     1ØØ3        1ØØ1      1/2" socket wrench                9.75
     1ØØ4        55Ø71     1/2" socket wrench               1Ø.25
     1ØØ5        55Ø71     1/2" socket wrench               1Ø.25
     1ØØ6        55271     combination pliers                6.89
     1ØØ7        1ØØ2      leather mallet                    8.75
     1ØØ8        61117     3" Phillips screwdriver           2.35
     1ØØ9        61117     3" Phillips screwdriver           2.69

                                        TOTAL===>             7Ø.43
```

APPENDIX

Order Maintenance - ORD1ØØ

```
/*****  ord1ØØ.h
       Global variables and defines
       for order maintenance - ord1ØØ
                *****/

#define MAXI 9          /* maximum number of items per order */
#define FIRSTLN 11      /* first line of item area on screen */

long iaddrs[MAXI+1];    /* table of item record addresses */
long total;             /* total price of all items on order */
#
/*************************************************************************
*
*       addln.c
*
*       Function to add item records to the current order.  It does
*       this by simply pfield-ing the current order number into the
*       item order number field.  Note however that if iorder is
*       already nonzero (i.e., this item is on some order), the add is
*       not allowed.  Either the other order must be deleted, or item
*       maintenance (itm1ØØ) must be used to take the item off the
*       other order.
*
*       arguments:
*               none
*
*       returns:
*               line number of line just added
*
*       author: wao
*************************************************************************/
$include "../../def/file.h"
#include "../../def/sord1ØØ.h"
#include "../../def/ord1ØØ.h"
```

```
addln ()
{
    int i,              /* for indexing */
        j;
    long xonum,         /* buffer for order number */
         xisal,         /* for sales price */
         xsno;          /* for item serial number */
    char msg[45];       /* message buffer */

    for (;;)
    {
        for (i = Ø; i < MAXI; i++)
            if (iaddrs[i] == Ø)
                break;
        if (i == MAXI)
        {
            prtmsg (1, 23, "no more items can be added");
            return MAXI-1;
        }
        while ((j = gtube (5, FIRSTLN + i, sno, &xsno))  != -2)
        {
            if (j == -3)
                continue;       /* item number must be entered */
            if (access (item, &xsno))
            {
                prtmsg (1, 23, "item not found");
                continue;
            }
            gfield (iorder, &xonum);
            if (xonum != Ø)
            {
                prtmsg (1, 23, "item already used");
                continue;
            }
            gfield (onum, &xonum);
            pfield (iorder, &xonum);
            pdata (15, FIRSTLN + i, imodel_monum);
            faccess (model, imodel);    /* get related model */
            pdata (24, FIRSTLN + i, mdes);      /* and display description*/
            gfield (isal, &xisal);
            ptube (54, FIRSTLN + i, isal, &xisal);      /* display price */
```

```
            total += xisal;                         /* add in price to total */
            outbuf (stotal, &total);    /* and update the screen */
            loc (item, &iaddrs[i]);                 /* save new record address */
            break;                                  /* remain in add mode */
        }
        if (j == -2)
        {
            eras_ln (l);
            return i;
        }
    }
}
#
/*****************************************************************************
*
*       addord.c
*
*       Function to add order records.  It accepts the order from the
*       user and attempts to insert the record.  If successful,
*       the table of item record addresses is cleared.
*
*       arguments:
*               none
*
*       returns:
*               Ø - new order not added
*               1 - order added
*
*       arthor: wao
*****************************************************************************/
$include "../../def/file.h"
#include "../../def/sordlØØ.h"
#include "../../def/ordlØØ.h"

addord ()
{
    int i;              /* for indexing */
    long xonum;         /* order number */

    while ((i = inbuf (sonum, &xonum)) != -2)
    {
```

```
        if (i == -3)
            continue;   /* order number must be entered */
        if (addrec (order, &xonum))
        {
            prtmsg (1, 23, "order already exists");
            erasprmp (sonum, sonum);    /* erase invalid order number */
            continue;
        }
        lockrec (order);                        /* lock the record */
        total = Ø;      /* total amount is zero */
        cfill (Ø, iaddrs, sizeof iaddrs);      /* zero address table */
        return 1;       /* new order added */
    }
    return Ø;           /* new order not added */
}
#
/*****************************************************************************
*
*       delln.c
*
*       Function to delete the current item from the current order.
*       It does this by setting the item order number field (iorder)
*       to Ø.  Note that there must be a dummy order in the database
*       with key = Ø for this to work.  Also it cleans up the table
*       of record addresses to reflect the deleted record.
*
*       arguments:
*               iax - index into iaddrs table of item to delete
*
*       returns:
*               none
*
*       author: wao
*****************************************************************************/
$include "../../def/file.h"
#include "../../def/ordlØØ.h"

delln (iax)
int iax;
{
```

```
    int j;
    long zero;

    zero = Ø;
    setloc (item, iaddrs[iax]); /* get item to delete current */
    pfield (iorder, &zero);     /* set order number field to Ø */
    for (j = iax; j < MAXI; j++)
        iaddrs[j] = iaddrs[j+1];        /* reshuffle address table */
    dspord ();                  /* redisplay item area on screen */
}
#
/*************************************************************************
*
*       delord.c
*
*       Function to delete order records.  Note that before the order
*       can be deleted, all items related to the order must be
*       taken off.  This maintains logical database integrity.
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*************************************************************************/
$include "../../def/file.h"
#include "../../def/ordlØØ.h"

delord()
{
    int i;              /* for indexing */
    long zero;          /* buffer for dummy record id */

    zero = Ø;
    if (yorn ("DELETE?") == 1)
    {
        for (i = Ø; i < MAXI; i++)
        {
```

```
            if (iaddrs[i] == Ø)
                break;  /* no more items for order */
            setloc (item, iaddrs[i]);    /* make item current */
            pfield (iorder, &zero);      /* move it to dummy order #Ø */
        }
      if (delete (order) == -1)
            prtmsg (1, 23, "order not deleted");
    }
    else
        prtmsg (1, 23, "order not deleted");
    erasord ();                          /* erase the screen */
    ulockrec (order);                    /* release the record */
}
#
/*********************************************************************
*
*       dspcust.c
*
*       Function to display the data in the top half of the order
*       screen.
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*********************************************************************/
$include "../../def/file.h"
#include "../../def/sordlØØ.h"

dspcust ()
{
    dsply (sonum, sodate);
    if (faccess (customer, ocust) == Ø)
        dsply (socust, sczip);
}
#
```

```
/*****************************************************************
*
*       dspord.c
*
*       Function to display all the items for the current
*       order record on the screen.
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*****************************************************************/
$include "../../def/file.h"
#include "../../def/sordlØØ.h"
#include "../../def/ordlØØ.h"

dspord ()
{
    int i;              /* for indexing */
    long xisal;         /* buffer for current item amount */

    total = Ø;
    for (i = Ø; i < MAXI; i++)
    {
        if (iaddrs[i] == Ø)
        {
            mv_cur (1, 2, FIRSTLN + i);
            eras_ln (1);                /* erase empty entries */
        }
        else
        {
            setloc (item, iaddrs[i]); /* make item current */
            pdata (5, FIRSTLN + i, sno);        /* display item data */
            pdata (15, FIRSTLN + i, imodel_monum);
            faccess (model, imodel);    /* get related model */
            pdata (24, FIRSTLN + i, mdes);      /* display description */
            gfield (isal, &xisal);      /* get sale price from db */
```

```
            ptube (54, FIRSTLN + i, isal, &xisal);      /* display it */
            total += xisal;             /* and add it in to the total */
        }
    }
    outbuf (stotal, &total);    /* display total on the screen */
}
#
/*****************************************************************
*
*       erasord.c
*
*       Function to erase data from the entire order entry screen.
*       The item area must be erased separately in case the terminal
*       doesn't have clear foreground
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*****************************************************************/
#include "../../def/ordlØØ.h"

erasord ()
{
    int i;       /* for indexing */

    cleancrt ();        /* clears the foreground */
    for (i = Ø; i < MAXI; i++)
    {
        mv_cur (1, 5, FIRSTLN + i);
        eras_ln (1);    /* erase the line */
    }
}
#
```

```
/*********************************************************************
*
*       findord.c
*
*       Function to accept an order number from the user and look it
*       up in the database.  If the order is found, the table of
*       item record addresses is filled in.  The data is then displayed.
*
*       arguments:
*               none
*
*       returns:
*               Ø - order record not found
*               1 - order found, iaddrs table filled in
*
*       author: wao
*********************************************************************/
$include "../../def/file.h"
#include "../../def/sordlØØ.h"
#include "../../def/ordlØØ.h"

findord ()
{
    long xonum;          /* buffer for order number */
    int i,               /* for indexing */
        j;               /* for return values from db functions */

    while ((i = inbuf (sonum, &xonum)) != -2)
    {
        if (i == -3)   /* <cr> was entered */
            continue;
        if (access (order, &xonum))
        {
            prtmsg (1, 23, "order not found");
            erasprmp (sonum, sonum);       /* erase bad order number */
            continue;
        }
        if (lockrec (order))
        {
            prtmsg (1, 23, "order in use at another terminal");
            continue;                      /* someone else is using it */
```

```
        }
        cfill (Ø, iaddrs, sizeof iaddrs);       /* zero address table */
        makeset (onum, iorder, last);           /* identify the set of
                                                   items for current order */
        for (i = Ø; i < MAXI; i++)
        {
            if ((j = prevrec (onum, iorder)) == -2)
                break;  /* no more items for current order */
            loc (item, &iaddrs[i]);             /* save record address */
        }
        dspcust ();                             /* display top half */
        dspord ();                              /* display the items */
        return 1;
    }
    erasprmp (sonum, sonum);
    return Ø;
}
#
/*********************************************************************
*
*       inqord.c
*
*       Function to perform order inquiry.  It waits for an operator
*       response before erasing the screen.
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*********************************************************************/
$include "../../def/file.h"

inqord ()
{
    prtmsg (1, 23, ""); /* wait for user response */
    erasord (); /* erase data from the screen */
    ulockrec (order);   /* release the record */
```

```
}
#
/*****************************************************************************
*
*       lnmnt.c
*
*       Function to maintain item records for the current order.
*       A maximum of 9 items can appear on a single order, for
*       simplicity.  The screen could be made to scroll if desired.
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*****************************************************************************/
$include "../../def/file.h"
#include "../../def/sordlØØ.h"
#include "../../def/ordlØØ.h"


lnmnt ()
{
    int i,              /* for indexing */
        j;              /* for status returns */
    char xchr[2];       /* buffer for operational mode */

    for (i = Ø;;)
    {
        if ((j = gtube (2, FIRSTLN + i, cstate, xchr)) == -2)
        {
            if (i)
            {
                i--;    /* move up a line */
                continue;
            }
            else
                return -2;      /* return to upper part of screen */
        }
```

```
        if (j == -3)
        {
            if (i == MAXI-1 || iaddrs[i] == Ø)
                i = Ø;  /* loop back to top of item area */
            else
                i++;    /* move down a line */
            continue;
        }

        prmp (2, FIRSTLN + i, "  ");     /* erase the mode character */
        xchr[Ø] |= Ø4Ø;                  /* map to lower case */
        switch (xchr[Ø])
        {
          case 'a':     /* add */
            if (iaddrs[MAXI-1] == Ø)
                i = addln ();
            else
                prtmsg (1,23,"no more items can be added");
            break;

          case 'd':     /* delete */
            if (iaddrs[i] != Ø)
                delln (i);
            else
                prtmsg (1,23,"can't delete empty line");
            break;

          case 'q':     /* quit */
            return Ø;

          default:
            prtmsg (1, 23, "Enter <cr>, a, d, or q");
        }
    }
}
#
```

```
/*************************************************************************
*
*       modord.c
*
*       Function to modify fields in the current order record.
*       The key field is not modifiable in this function.  When the
*       item entry area is reached, another function is called to
*       handle the processing.
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*************************************************************************/
$include "../../def/file.h"
#include "../../def/sordlØØ.h"

modord ()
{
    int i,      /* for indexing */
        j;      /* for status returns */

    i = sodate; /* set to first modifiable field */
    while (1)
    {
        switch (i)
        {
            case sodate:
                if (input (sodate) == -2)
                {
                    erasord (); /* erase the data on the screen */
                    ulockrec (order);   /* release the record */
                    return;
                }
                i = socust;     /* next modifiable field */
                break;
```

```
            case socust:
                if ((j = input (socust)) == -2)
                {
                    i = sodate; /* back to previous field */
                    break;
                }
                if (faccess (customer, ocust) == Ø)
                    dsply (scname, sczip);      /* display customer data *
                i = samd;       /* next modifiable field */
                break;

            case samd:
                if (lnmnt () == -2)
                {
                    i = socust; /* back to previous field */
                    break;
                }
                i = sodate;     /* beginning of screen */
                break;
        }
    }
}
#
/*************************************************************************
*
*       ordlØØ.c
*
*       This is the control routine for order entry.  It gets the mode
*       of operation from the user and calls the correct routine to
*       perform the indicated operation.
*
*       arguments:
*               none
*
*       returns:
*               none
*
*       author: wao
*************************************************************************/
$include "../../def/file.h"
#include "../../def/sordlØØ.h"
```

```
ordlØØ ()
{
    int mode;

    while ((mode = priamd (1)) != -2)
    {
        switch (mode)
        {
            case Ø:     /* inquire mode */
                while (findord ())
                    inqord ();
                break;

            case 1:     /* add mode */
                while (addord ())
                    modord ();
                break;

            case 2:     /* modify mode */
                while (findord ())
                    modord ();
                break;

            case 3:     /* delete mode */
                while (findord ())
                    delord ();
        }
    }
}
```

INDEX