dBASE III PLUS

M.dePace

dBASE III PLUS

Mario de Pace

COLLINS
8 Grafton Street, London W1

Collins Professional and Technical Books
William Collins Sons & Co. Ltd
8 Grafton Street, London W1X 3LA

First published in Great Britain by
Collins Professional and Technical Books 1987

British Library Cataloguing in Publication Data
De Pace, M.
dBase III Plus.
1. dBase III Plus (Computer program)
I. Title
005.75'65 QA76.9.D3/

ISBN 0–00–383378–X

Typeset by V & M Graphics Ltd, Aylesbury, Bucks
Printed and bound in Great Britain by
Mackays of Chatham, Kent

dBASE III PLUS

Other books of interest

Working with dBASE II
Mario de Pace
0 00 383251 1

The dBASE Programming Language
Mario de Pace
0 00 383267 8

IBM Personal Computer
Mario de Pace
0 00 383003 9

Office Automation Using the IBM Personal Computer Systems
Joseph St John Bate
0 00 383104 3

SuperCalc Prompt
Randall McMullan
0 00 383004 7

WordStar Prompt
Randall McMullan
0 246 12446 6

WordStar in Action
Randall McMullan
0 00 383107 8

Choosing Accounting Software For Your Microcomputer
G. Simon
0 00 383006 3

The Pick Operating System
Joseph St John Bate and Mike Wyatt
0 00 383160 4

MS/PC DOS Prompt
Randall McMullan
0 00 383288 0

Contents

Chapter One
Introduction

The importance of a database

A database package is essentially a piece of computer software that helps you to organise and use your information with as little effort on your part as possible. It is general purpose in the sense that it does not matter whether your information consists of a list of customers, an inventory of stock items or a collection of gramophone records. It will let you hold a large number of different bits of information about each item, and it will let you group, summarise, and sort your information in any way that you want. When you want to locate a specific item, it will find it for you almost immediately.

The term database package is a quick way of saying 'database management system'. Although it gives the impression of simplicity itself, it is really much more complicated than it sounds; such a system may be likened to a swan that appears to glide effortlessly over the water while beneath the surface it is paddling like fury. It is a complex program that performs the task of managing information contained in a database file. It provides the user with facilities for adding, amending and deleting data, producing information reports from more than one file and making enquiries that access more than one file. A database package also provides special file management facilities, such as merging database files and even amending the structure of a database file without any loss of data. A database manager of this kind differs from programming languages in that most of the basic functions such as creating a file, loading data, editing the data and so forth are ready made. There is thus an immediate and tremendous saving of development effort since the user does not have to spend several weeks writing programs to provide these basic functions.

An important feature of a database manager is its flexibility, both in allowing the user to create a database file tailored to his own needs, and in providing him with a trouble-free means of changing his mind about

what he wants his database to contain. The same database package can be used by a marketing department to maintain records of prospective customers and by a sales manager to keep track of the performance of his sales team. The domestic user will use it to maintain an index of his books or do his accounts while the shopkeeper will use it for stock control. In fact, anybody who has to maintain information which is voluminous, or changes quickly, or is complex, will benefit from a database package. It enables the user to control information and, just as important, to obtain quick and correct answers to questions.

Personal computer databases

There are a large number of software packages on the market which are listed under the heading Data Management. Some of these are powerful database managers, while others are simpler file maintenance and report producing packages. Some are based on the card indexing method of storing and retrieving information and do not include functions such as totalling a numeric item, for example, the cost of a stock item.

Until recently, the most successful database package in any category was Ashton-Tate's dBASE II which was used by hundreds of thousands of computer users. It is a complete database manager which has, in addition, its own programming language. Indeed it proved to be such a successful basis for developing applications that a large number of software packages were developed using the dBASE II programming language.

The success of dBASE II led Ashton-Tate to develop a follow-on product using the same approach but taking advantage of the power and capacity of the new generation of personal computers, like the IBM PC, with 16-bit microprocessors able to run far larger programs than their 8-bit predecessors. This product, dBASE III, required a 256K memory but its specification was quite impressive. It allowed up to a billion records in each database file, with a single record size of up to 4000 characters. It let you use ten database files at once, using its relational features to obtain information from all ten.

It provided efficient multi-indexing on each database file so that an item of information could be looked up from more than one starting point, for example, Surname as well as Customer Code. Apart from being more powerful than dBASE II, it also had several new features

such as the ability to produce name and address labels, or the means to store word processing text in conjunction with database information.

dBASE III soon reached the number one spot in the database software charts, toppling its predecessor dBASE II from that position. Meanwhile, Ashton-Tate continued developing their new product and late in 1985 they replaced dBASE III with a completely new version, dBASE III PLUS. Any programs written in the dBASE III programming language will work unchanged with dBASE III PLUS: the latter simply represents a more advanced version. Although the dBASE III PLUS specification in terms of maximum number of database files, maximum records and so on does not exceed that of dBASE III it does have a large number of extra features. It also has the means of working on a multi-user computer system.

To new users, who may not be too concerned with such refinements, it continues the dBASE tradition of being an extremely powerful computer tool that is easy to use and that offers the means of deriving substantial benefit from a personal computer without having to wait for expensive and time-consuming programs to be written.

Users of dBASE II

dBASE III PLUS treads ground similar to that of dBASE II and many existing users of the latter will be interested in learning about dBASE III PLUS. The packages are similar, yet there are enough differences to require a relearning exercise on the part of those who have experience of dBASE II. Chapter 12 discusses some of the important points to be observed in changing over from dBASE II to dBASE III PLUS. Included in that chapter is a description of a conversion program, dCONVERT, which is contained in the dBASE III PLUS package and which may be used to do much of the work of converting dBASE II programs, database files, and so forth, to run under dBASE III PLUS.

The exercise of converting from dBASE II to dBASE III PLUS is well worth the effort, not only for the extra features and extra power of dBASE III PLUS but because the latter is continually being improved while work on developing dBASE II as a software product has already ceased. Also, for those who want to write programs in the dBASE programming language, it is possible to buy a compiler for dBASE III PLUS which will convert dBASE III PLUS programs into freestanding programs that require neither dBASE III PLUS nor the compiler.

Preliminaries

Before you load dBASE III PLUS there is one bit of preparation that will come in very useful as you use more and more of the features of dBASE III PLUS. It is mainly a technical item that will not affect you during your initial use of the package and it is unfortunate that a technical item should intrude at this early stage of explaining the use of the software. Unless you are aware of this point, however, you may be confused by a certain reluctance on the part of dBASE III PLUS to let you use as many database files as you would expect, or even to let you do certain things such as creating an extra printed report. These are unlikely events while you are using the package as described in Chapters 2 and 3 and you may, therefore, safely skip to the next section and return to this section at a later stage but by explaining it here you will at least be aware of such a possible complication.

The operating system reads a file called CONFIG.SYS when it starts up and this file may be used to tell the operating system the maximum number of files that it should allow to be in use at any one time. Since one of the strengths of dBASE III PLUS is its ability to handle a number of different files all at once, we do not want the operating system to restrict this in any way. We will use it to tell the operating system to allow up to 20 files to be in use at the same time. You can also use this method to set up a large number of buffers. These are the memory areas which are used to hold the data being read from or written to a disk file. The more buffers there are, the less time will be wasted in reading or writing. It is recommended that you set the CONFIG.SYS file to contain the following:

```
files=20
buffers=24
```

The method of creating the CONFIG.SYS file is described in Appendix A.

Getting started

dBASE III is loaded by entering DBASE against the DOS prompt. Since the software is protected against illegal copying, it is neccessary to have the System Disk 1 in the drive from which the software is being loaded. Once loaded, you will be asked to replace System Disk 1 with System Disk 2. On a hard disk computer the software may be installed onto the hard disk and subsequently loaded from there without

requiring the use of either system disk. The install procedure is described in one of the booklets that accompany the software but really consists of no more than inserting the System Disk 1 and typing in INSTALL C:. Thereafter you simply follow the instructions which appear on the screen.

If you have loaded the program correctly, it will display a copyright screen which also contains the dBASE III PLUS version number. If you wait for a few seconds or if you press the Return key, the dBASE III PLUS Assistant screen will appear. This is a screen that allows you to use dBASE III PLUS by means of 'menus' (lists of options) from which you choose the task or operation that you want to carry out. We will return to this menu screen in a moment but let us first look at another way of starting up dBASE III PLUS operations. This is done by means of dBASE III PLUS commands.

dBASE III PLUS is command driven in the same way that DOS is. It has a large number of commands giving the user a very powerful and flexible tool with which to organise and reproduce information. A command processor needs a prompt that invites you to enter a command, and you are no doubt familiar with the DOS prompt that allows you to enter a command such as DIR to list the files on a diskette. dBASE III PLUS has a similar prompt, the full stop ('dot'), which allows you to enter one or more of its large number of keywords as a communication medium. To move from the Assistant menu screen to the dot prompt, you press the Escape key. The screen will clear and the dot prompt will appear at the foot of the screen with the cursor blinking beside it – this is your invitation to type in a dBASE III PLUS command.

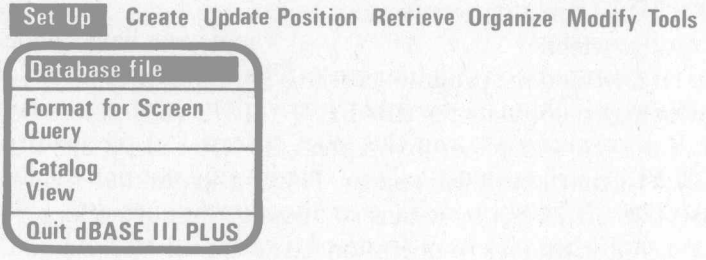
On the line beneath the dot prompt is a Status line which, as we will see later, serves to provide useful information about the activities of dBASE III PLUS. The command for returning to the Assistant menu screen would be typed in as follows:

```
. assist
```

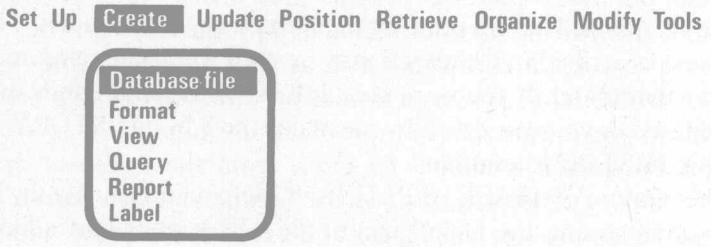
You then press the Return key and the original menu screen will re-appear. It should be clear from the above that you can use either the menu approach or the command approach in working with dBASE III PLUS. You can even use a mixture by flicking from one to the other. If you would rather work with commands and you do not want the Assistant menu to appear when you start dBASE III PLUS, you can prevent it doing so as explained in Appendix C.

Let us next look at how you would work using the menu screen. The top line contains a set of eight main menu options, and each of these

has a sub-menu which is listed underneath it: this appears as shown below.



The first of the main menu options is highlighted by means of a highlight or cursor bar. The Left and Right Arrow keys will move this cursor bar from side to side on the top line. As the cursor bar moves from main menu option to main menu option, the vertical list of sub-menu options changes too, as shown below.



The top option in each sub-menu is marked out by a second cursor bar which may be moved up or down by means of the Up and Down Arrow keys. You choose a sub-menu option by moving the cursor bar up or down to the required option and then pressing the Return key. There are nearly 50 options available on the Assistant menu screen and it would be confusing to try and explain them at the outset. Instead we will cover each as we follow through the examples contained in this book, usually as part of explaining the equivalent dBASE III PLUS commands.

Returning to the dBASE III PLUS prompt, when you enter a command incorrectly, dBASE III will inform you by means of its error messages where you have not followed the rules. It will also offer you a 'help' facility which explains the syntax of the command, provides a description of the command, and allows you to browse through like descriptions of associated commands. Examples of error messages include the following:

```
File does not exist
Unrecognised command
Unrecognised phrase
Syntax error
```

Certain error messages try to show you where you have made a mistake. For example, you will find out in Chapter 3 how to list your information by specifying the names of your various information items or fields. If you specify an item that does not exist in the database, dBASE III PLUS will not only tell you that it could not find the item, it will also place a question mark over the item in question. This is especially useful when you have specified a whole list of items.

To save you unnecessary typing, dBASE III PLUS saves earlier commands and allows you to retrieve and correct or change them. Only the most recent 20 commands are saved but this is more than enough to save lengthy retyping. Later we will see how to increase this number if we need to do so. A saved command is retrieved by pressing the Up Arrow key once for each previous command. These will appear, one by one, at the dot prompt where a command may be corrected or changed before pressing the Return key to execute it. The control keys described in Chapter 2 may be used when changing such a retrieved command. If you want see a full list of the commands that have been saved, you can do so by means of the LIST HISTORY or DISPLAY HISTORY command.

Another feature of dBASE III PLUS is its Help menu which may be called up by pressing the Help function key. The latter varies from computer to computer, for example, on an IBM PC the F1 function key is used as the Help key. The Help menu contains options such as 'What is a ...?' or 'How do I ...?' to take you to Help screens that provide the appropriate explanations. It is possible to browse about from screen to screen so that the Help feature can also be used to provide an introduction to the software.

dBASE III PLUS commands come in many sizes. Some are short, such as USE, while others are long, such as DISPLAY STRUCTURE. Once you have become familiar with the commands, you may find it tedious to enter the longer commands in full. You may, therefore, prefer to use the abbreviated form which consists of the first four letters of each word, for example DISP STRU.

With these minor preparations out of the way, you may proceed to use the program which you will find far more interesting than reading about rules as you have just been doing. In the next chapter you will be shown how to create a database file, add information to it, change the information afterwards, and then list the details you have entered.

Chapter Two

Creating and Using a Database

Designing a database

Although dBASE III PLUS is capable of working with ten different database files at once, we are going to start with just one. To illustrate the use of dBASE III PLUS in its basic mode we are going to create a database file that will allow us to maintain a record of a fleet of vehicles belonging to a company called Swiss Windows. For the purposes of the illustration, it does not really matter what the business of Swiss Windows is but let us assume that they manufacture and install windows, and that apart from the directors' cars they also have a large fleet used by the sales team. The database will be a simple one purely to illustrate the first steps of using dBASE III PLUS, so do not be surprised if obvious details such as the vehicle registration details are not recorded.

The database will help us to keep a record of each vehicle, showing the make and model, its type (for example Saloon or Estate), its value, and which member of staff is using the vehicle. The records will be used to: list all the vehicles of a certain make, say all the Volvos; to find out which vehicle is being used by a certain person; to find out whether there is a certain model in the fleet; to provide a full list of the entire fleet; to determine how much money we have invested in estate cars; and so forth.

The first task is to decide what information we want to store and the maximum size of each item of information. It looks like this:

```
Make       : 10 characters
Model      :  8 characters
Type       :  8 characters
Driver     : 17 characters
Value      :  8 characters

Total size  52 characters
```

Each item of information is known as a field, and the collective name for the fields representing a single entry, i.e. one vehicle in this illustration, is a record. You will see that the total size of the information for each vehicle is one character larger than the sum of the individual field sizes. This is because dBASE III PLUS needs the extra character to store information about the record. Except for the Value field, the information to be loaded into each field will be descriptive text such as the name of a make of car. This kind of information is described as character data or text data. The Value field will be a numeric field which may be used in totalling or arithmetic operations.

The next thing to consider is the size of the database file in relation to the capacity of the diskette that will hold the file. We can estimate how many records will be needed to store the details of all the vehicles and then multiply that number by the size of the individual record. In this case we have a fleet of 245 vehicles, so the total file size will only be around the 12K mark, i.e. 52 times 245 giving 12740. On top of that there will be a further 195 characters which are used to store the details of the database structure. The size of the structure is not fixed at 195; it is dependent on the number of fields in the database, but it is, in any case, a very small part of the database file and may be ignored for purposes of trying to determine whether your file will fit on the diskette.

Our database file is so small that there is clearly no problem, but in another situation you may find that a database requires a much larger record and you will have to store thousands of records. For example, if you need to store 10000 records each 200 characters in length, you will have to consider using a hard disk giving, say, ten million, characters of available space. It is important to have spare space available on a diskette to allow for additions to the file, or you might decide to add another item of information to the record. As you will discover later, there will also be a need for indexes and other files in addition to the main database file. Finally, the dBASE III PLUS software files take up nearly 360K. For the sake of simplicity, we will assume throughout the rest of this book that we are using a hard disk computer and that all our files will be held on the hard disk which is known as Drive C:.

Creating the database

Now that we have designed our database, we need to tell dBASE III PLUS about it. This can be done either from the Assistant menu screen or the dot prompt. Let us look at each in turn;

Assistant menu: Here we select the *Create* menu option on the top line, and the *Database file* option on the sub-menu. We will then be shown a list of disk drives from which we choose one by moving the cursor bar to the appropriate drive letter (in our case this will be C:) and pressing the Return key. Next we will be asked for the name of our database file. We have already decided to call it MOTOR so we type that in. dBASE III PLUS allows enough space to type in 20 characters but remember that DOS restricts filenames to eight characters. Once we have typed in the name of our database there is no difference between using the Assistant menu or the dot prompt.

Dot prompt: As explained in the previous chapter, the dBASE III PLUS prompt consists of a dot. Against this prompt we will now enter the command CREATE. We could have added the name of the file to be created but since we did not, we will next be asked for the name of the file. Remember that DOS restricts filenames to eight characters. Having decided to call our database file MOTOR, we enter that. From this point onwards there is no difference between using the Assistant menu or the dot prompt.

dBASE III PLUS will next clear the screen and display a box containing explanations of some of the control keys (which we will ignore for now) and, in tabular form, an area for you to enter the field name, type, and size of each field to be contained in a database record. We have already determined most of this information, so we can start entering immediately underneath the captions. The line immediately beneath the captions will be highlighted and the cursor will be blinking in the first of the highlighted areas underneath the fieldname caption to show you where to start.

When you have entered a fieldname, you press the Return key and the highlight cursor will jump to the next point on the line and wait for you to enter the fieldtype. The latter will already be set to *Character* and unless you want to change this description, as you will shortly on the Value field, you simply press the Return key again to move on to next point on the line which has the caption Width. This is where you enter the size of the field. There is another caption on the line, the Dec or number of decimals, but having described the field as Character, the cursor will bypass this point and come to rest on the line below ready for you to enter the details of the next field.

When you come to enter the field type of the Value field, you will only need to enter the N of the word 'numeric' and the *Character* description will instantly change to *Numeric*. An alternative method of

choosing a field type other than Character consists of pressing the space bar to revolve the preset field type description until you see the type you want. After one depression of the space bar you will see the word Numeric appear in place of Character. You then select it by pressing the Return key. Next you enter the Width of the numeric field, after which the cursor will wait underneath the Dec caption for you to enter the number of decimals required. As you enter the field details, you will see at the top of the screen a running account of the number of characters remaining out of the 4000 character maximum that may be used in a single record. The status bar shows the number of fields defined so far. When the five fields of our MOTOR database have all been entered, the screen will contain the following:

	Fieldname	Type	Width	Dec
1	MAKE	Character	10	
2	MODEL	Character	8	
3	TYPE	Character	8	
4	DRIVER	Character	17	
5	VALUE	Numeric	8	2

If you want to change any of the above, you can move the highlight cursor back up to the required line to make your changes. If you want to insert a field, say between fields 2 and 3, you would place the highlight cursor on line 3 and create a blank field by holding down the CTRL key and pressing the letter N. Lines 3 and onwards would be moved down one line, leaving a blank line in which to enter the additional field. You may similarly delete a field by placing the highlight cursor on the field in question and pressing CTRL and U.

Let us assume, however, that we do not want to make any changes and that we want to move on to the next step. To finish, you press the Return key on a new line. The database file will now be created and you will be asked whether you want to start entering information immediately. If you reply No to this, you will be returned to the Assistant menu screen or dot prompt, depending on where you started from. Let us assume that we do not want to type in any information at this stage so that we can spend a bit more time looking at the CREATE command.

The fieldnames may be up to ten characters long. There are five different field types: Character, Numeric, Logical, Date, and Memo. Character or text fields are used for any textual information and may be up to 254 characters in size. Numeric fields are a bit more complicated. You will see that the Value field was entered with a width