



esm
Software for Schools

Single User Publisher
ESM
Duke Street
Wisbech
Cambs
PE13 2AE

ACORNSOFT
The choice of experience
in software.

Multi User Publisher
Acorn Computers Ltd
Cambridge Technopark
645 Newmarket Road
Cambridge
CB5 8PD

The Database for the
British Broadcasting
Corporation
Microcomputer

User Guide by Jim Milne

SupaStore SupaStore
SupaStore SupaStore
SupaStore SupaStore
SupaStore SupaStore

SupaStore User Guide

Published and distributed by

ESM

Duke Street,
Wisbech PE13 2AE

© Council for Educational Technology 1986

First published 1986

ISBN 0 905114 17 5

Other than as permitted by the Copyright Act 1956 no part of this publication may be photocopied, recorded or otherwise reproduced, stored in a retrieval system or transmitted in any form by any electronic or mechanical means without the prior permission of the copyright owner.

Printed in Great Britain by
Ebenezer Baylis & Son Ltd
The Trinity Press, Worcester, and London

Library Reference Details

Program Name : SupaStore — An Information Management System for the
British Broadcasting Corporation Computer

Publisher : Acorn/ESM

Software Authors : Andrew Jacobs and David Tee.

Documentation Author : Jim Milne

Design and Development : Bob Coates, Andrew Jacobs, Jim Milne and David
Tee.

ISBN : 0-905114017-5

Dewey Number : 025.04

Contents

The SupaStore guide

1 : How to use this guide	9
---------------------------	---

Introduction to SupaStore

2 : Information management with SupaStore	11
3 : Overview of SupaStore features	14

Using a database

4 : The example database	16
5 : Getting started	18
6 : Defining queries	23
7 : Defining report formats	27
8 : Sorting report records	30
9 : Revising queries and formats	33
10 : Deleting queries and formats	35
11 : Changing the information stored in records	36
12 : Adding new records to the database	39

Creating a new database

13 : SupaStore utilities	42
14 : New databases — preliminary considerations	44
15 : Creating new databases	49

Utilities & advanced features

16 : Database description	53
17 : Record selection	54
18 : Archiving and retrieving queries and formats	57
19 : Labelling facilities	60
20 : Creating sub-databases	63

Printing and saving reports

21 : Printer output	65
22 : Spool file reports	67

Network use

23 : Network user considerations	69
----------------------------------	----

Appendices

Appendix A : Botanical information	77
Appendix B : Error messages	83
Appendix C : Query reference information	85
Appendix D : Command reference information	87
Appendix E : Format reference information	88
Appendix F : Sorting reference information	90
Appendix G : Database reference information	92
Appendix H : Technical details	95
Appendix I : Screen design sheet	96

Glossary	97
----------	----

Index	101
-------	-----

List of Illustrations

2.1	Managing a Paper Database
3.1	Screen Features — Main Menu
3.2	Screen Features — Query Selection
4.1	Example Record — The Sessile Oak
5.1	Main Menu Screen — British Trees
5.2	Query Selection Screen — British Trees
5.3	Format Selection Screen — British Trees
5.4	First Page of List all Records Output
5.5	Searching the Database Screen
5.6	Query Selection Screen after Searching
6.1	Define a New Query Screen
6.2	Query Selection Screen with a New Query Installed
7.1	Define a New Format Screen
8.1	Report sorted by Botanical Family
8.2	Query Selection Screen with a Sort Flag
8.3	Format Selection Screen with a Sort Flag
12.1	Adding Records to a Database
12.2	Unused Records in a Database
17.1	Select Records Screen
18.1	Archive/Retrieve Queries Screen
19.1	Definition of Label Terms
19.2	Label Definer Screen

SECTION 1 : How to use this guide

Organisation

The numbered sections in this User Guide are intended to be instructional. They encourage practical use of the system to familiarise new users with its basic operation.

Sections 2-4 provide introductory information and a description of the example database supplied with the SupaStore pack. However, practical experience begins at Section 5 which helps you to get started through the production of *reports* from the example database.

Sections 6-12 progress through the definition of SupaStore *queries* and *report formats* to using the *editor* to change and add to the information in the example database.

The *Utilities* Sections (13-20) can then be followed to access advanced features including the creation of new *databases* and *libraries*, and print labels.

The detailed appendices are provided as a reference source for experienced users.

Considerations for users with different storage systems

SupaStore operates as a *standalone* system with databases stored on *floppy* or *hard disk*, or as a *multi-user* version within *Econet*. Separate discussion is used to make clear operational differences between different storage systems. *Network* users can also share databases with other *Econet* users; a special Section (23) deals with this and other network features.

Symbols and terms used in this guide

Words which have a specific meaning in SupaStore are defined in the Glossary. The first occurrence of these words in each section appears in italic type (eg *query*).

All references to actual keys on the British Broadcasting Corporation micro keyboard are represented by special symbols. For example **RETURN**, **11**, **1**.

When an entry from the BBC keyboard is required, the documentation uses the terms 'enter' and 'press' consistently to distinguish between two forms of input. When instructed to 'enter' something, the user must type a series of characters and then press the **RETURN** key. If the characters are specified, they appear in extra bold (eg **★SUPASTORE**). When asked to 'press' a certain key, that key should be pressed once only and without use of the **RETURN** key.

All SupaStore screen titles are distinguishable from the rest of the text. For example «Main menu».

References to *options* the user must select from screen menus appear as Reports.

SECTION 2 : Information management with SupaStore

Why is information management by computer a useful thing? For the purpose of discussion, consider that you are interested in the trees commonly seen in Britain and want to collect, store and retrieve information about them.

One way of proceeding without a computer would be to store your information on paper. You could make out a sheet like the one in Figure 2.1a. In this there is a blank space, or *field*, next to each of the tree characteristics you would use to distinguish between tree types. If sheets are filled in for a number of species they can 'store' this information in an organised way because the internal layout of each sheet is identical.

With a stack of completed sheets you have a paper *database* to explore questions that interest you about these trees.

For example, you might want to know which trees are evergreen. You could go about answering this by looking at each of the completed sheets in turn. If the relevant field contained 'Evergreen' you would put the sheet on a 'yes' pile; if it did not, you would put it on a separate 'no' pile (Figure 2.1b).

After all the sheets are checked, the 'yes' pile would represent the result of *searching* your database. You might then follow this up by looking at the 'yes' sheets to see what other attributes Evergreen trees have in common. For example, do they reach similar mature heights, are their leaves similar, or do they grow in similar locations?

This approach has one immediate problem however. If you want to ask another question you will have to sort through all the sheets again and establish a new 'yes' pile. In doing so you lose the result of your last question.

In reality, storage and retrieval on paper is only convenient for small numbers of sheets. Even if your paper database is small, it becomes increasingly difficult manually to check more complex questions. Consider how long it might take to find the sheets corresponding to tree types that were evergreen, not native to Britain and had mature heights between 30 and 40 metres.

Another problem with paper storage relates to the difficulty in changing written records when necessary.

In the examples above, you can only look at all the information stored on the sheets answering a particular question. What if you want to reproduce only selected fields for different audiences? Clearly, you will have to rewrite the stored information each time to achieve this censorship.

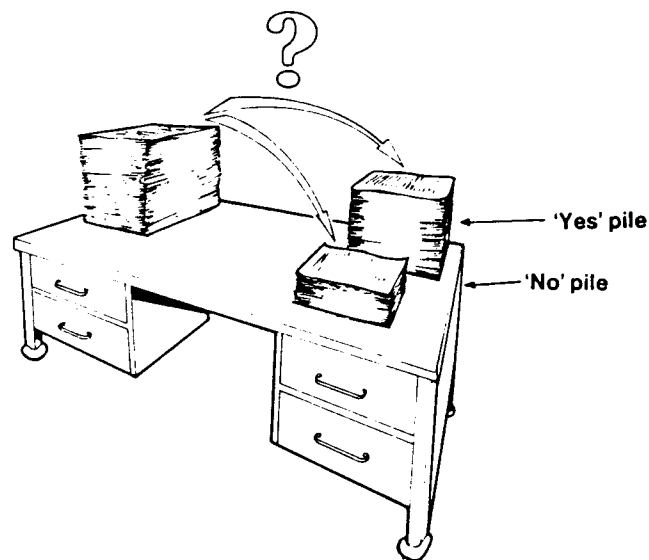
All these comments point to the problems that an information management system like SupaStore can overcome.

In SupaStore, information is entered from the keyboard onto screens organised into fields of fixed length, much like the paper sheet in Figure 2.1a.

a)

TREE DATA RECORD SHEET	
Sheet No	
Pop Name	
Other Name	
Botanical Name	
Family	
Decid/Ever	
Native	
Mature Height	
Leaf Description	
Fruit	
Location	
Use	

b)



2.1 Managing a Paper Database

The basic unit of a SupaStore *database* is called a *record*. Each record corresponds to a single object measured (eg a type of tree). As with paper equivalents, the fields in a record can be stored on a single screen page, or laid out over up to eight screens.

However, use of a SupaStore database is much more effective. Unlike paper sheets the contents of each record are easily changed.

Questions asked of SupaStore databases are called *queries*. After queries have been put to each record in turn, the queries themselves, and lists of the records that answer their question, are saved in the system. You can then re-call the results of any search by referring to the query name.

Information from the records matching queries is presented in SupaStore *reports*. Reports can include all or any of the fields by using a flexible *report format* facility and can be displayed on a variety of media.

Possible applications

Because database organisation and contents are independent of the system itself, potential uses of SupaStore provide a wide range of opportunities.

Education

- i) Individual students can store and analyse information collected from libraries, in laboratories or during fieldwork.
- ii) Equally, the system can be used to teach information studies itself.
- iii) As well as allowing cross-curricular applications, databases can be tailored to a wide range of age groups.
- iv) SupaStore would also be a powerful administrative tool in the school office or library.

Business and Commerce

Wide applications in any office or shop needing to store, update and access information that is continually changing. For example, personnel records, sales information.

Home and Hobbyist

There are database opportunities in any area which is currently supported by reference books. For example, naturalist, gardening, recipes, stamps, books, addresses, gazeteers, holiday guides.

SECTION 3 : Overview of SupaStore features

Organisation

SupaStore is a system for the British Broadcasting Corporation Microcomputer that allows you to manage information stored in *databases*.

SupaStore databases consist of a finite number of *records*. Each record is organised into *fields* prefixed by a field name. The fields within a record can be of variable length up to a maximum of 255 characters. Other limitations on record and database size are listed in the Database Appendix.

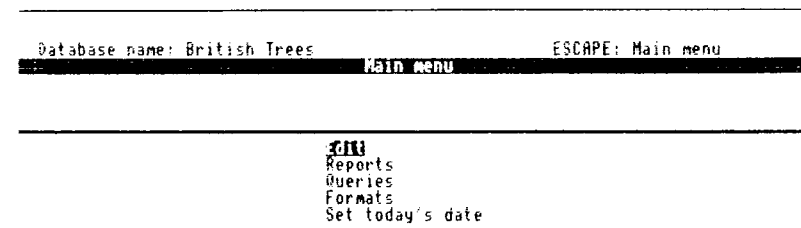
You can store any information in fields, or you can restrict the type of characters that can be entered to just words, numbers, or dates. Other special types of field are described in Section 14.

Menu features

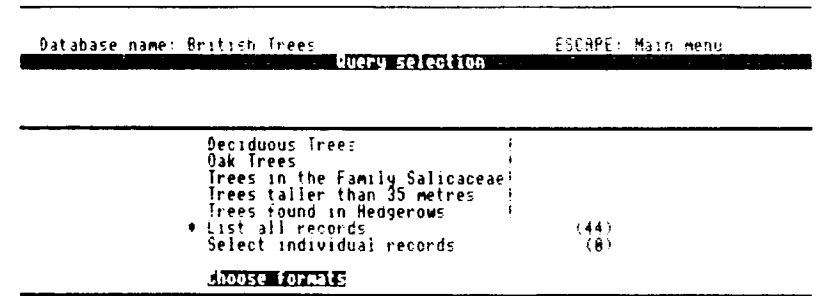
Operations within SupaStore mostly require you to make choices from screen menus. Apart from initial data entry and *editing*, a *cursor selection* feature makes keyboard entries largely unnecessary.

Nonetheless, different forms of cursor selection menus will be encountered as you progress through the different facilities available.

Most menus display a vertical list of *options* bounded by two horizontal lines (Figure 3.1). Each option lies on a separate line and a *cursor* initially covers the topmost option. The user can move the cursor up and down the list of options with the **↑** and **↓** arrow keys. An option is selected by locating the cursor over it and pressing **RETURN**.



3.1 Screen Features — Main Menu



3.2 Screen Features — Query Selection

Some menus have a detached option at the bottom (Figure 3.2). Here the user must first cursor-select one option from the main list and then proceed by selecting the detached option.

Dealing with mistakes

If you cursor-select the wrong option, or make other mistakes, you can press the **ESCAPE** key to get yourself out of trouble.

Pressing **ESCAPE** in SupaStore has a consistent meaning; when pressed, all the decisions made at the current screen are ignored and control is passed to the screen advised by the escape message in the top right of the current screen. This is usually the last screen you interacted with.

SECTION 4 : The example database

The early sections in this User Guide are intentionally linked to an example database, **trees**, supplied with the pack. Practice with this database will allow new users to become familiar with basic operations without having to be concerned initially with the more complex tasks of creating databases and defining queries and report formats.

Database of British Trees

The subject of the example database is identical to the paper database described in Section 2. It consists of 44 records each organised into 12 fields.

Each record represents a single tree species that is commonly found in Britain. The records cover a one screen 'page'. An example record, for the Sessile Oak, (Figure 4.1) shows that it has been possible to organise the information in much the same way as on the hypothetical sheet in Figure 2.1a.

Record no. 4 Date: 04/09/85 Page: 1

Pop. Name: [Sessile Oak]

Other Name: [Durmast Oak]

Bot. Name: [Quercus petraea]

Family: [Fagaceae]

Decid/Ever: [Deciduous]

Fruit: [Y]

Mature Ht: [40]

Leaf Descr: [Simple Pinnately-lobed]

Fruit: [Huts]

Location: [Woods Hedgerows]

Use: [Joinery Shipbuilding]

Keywords: [

4.1 Example Record — The Sessile Oak

The Fields

The fields used are not meant to be exhaustive, but cover the characteristics that can be used to distinguish between different species. Moreover, although every attempt has been made to ensure field contents are accurate, definitions and information may not match all sources. The actual fields are as follows; numbers in parenthesis indicate their length in characters.

Field	Explanation
Pop Name (19) :	Popular name.
Bot Name (24) :	Botanical name.
Other Name (22) :	Other name in common use
Family (16) :	Botanical Family.
Decid/Ever (9) :	Is the tree type deciduous or evergreen?
Native (1) :	Is the tree type native to Britain?
Mature Ht (2) :	The mature height in metres.
Leaf Descr (25) :	Botanical description of the leaf shape.
Fruit (7) :	Type of fruit used for propogation.
Location (35) :	Where the trees are most commonly seen.
Use (35) :	The use made of the tree type.
Keywords (50) :	Field for individualisd use.

A Botanical Appendix provides full explanations of the database fields, a botanical glossary, leaf silhouettes, and a list of useful reference sources on trees.

Example Queries and Formats

A small number of pre-defined queries and report formats are also provided with the example database. These are used in the next section to ask relevant questions and produce reports which illustrate the main features of the system.

SECTION 5 : Getting started

After working through this section the user will be able to produce SupaStore *reports* from the example *database* and should be familiar with the basic operation of the system. Please note that your British Trees disk may contain additional queries and formats to the examples below if someone else has worked through these sections before you.

Things to do first

It is assumed that the SupaStore EPROM is installed in your microcomputer. If this is not the case, please refer to the installation instructions on the separate sheet provided with the pack.

The example database disk supplied is 40 track. If you will be working with 80 track *floppy disks*, *hard disk* or *Econet* you will need to convert the disk supplied, or install the data on your system. See the separate sheet supplied with the SupaStore pack.

Getting started

Before starting, refer to Section 1, where the terms and symbolism used below are explained. Remember, 'enter' means that you need to type the characters specified and press the **RETURN** key. If you encounter any unexpected problems refer to the Errors Appendix.

Starting procedures differ slightly according to the storage system you are using.

FLOPPY DISK

Insert the disk labelled 'Database of British Trees' into your disk drive (this should be drive 0 if you have a dual disk drive). Enter **★SUPASTORE**.

HARD DISK and ECONET

Ensure you are in your own SUPASTORE sub-directory. Enter **★SUPASTORE**, then enter **Trees** when prompted for a database name.

All users should have reached the «Main menu» screen (Figure 5.1).

```
Database name: British Trees          ESCAPE: Main menu
Main menu

Edit
Reports
Queries
Formats
Set today's date

Enter today's date:
```

5.1 Main Menu Screen — British Trees

Setting today's date

It is good practice to start every SupaStore session by using the «Main menu» option Set today's date. This is because some options use this value, and *reports* are always labelled with the date stored. The date is set initially to when the database was created.

To *cursor select* Set today's date simply move the *cursor* over that option and press **RETURN**. (Remember, if you accidentally select the wrong option, simply press **ESCAPE** and you will be returned to the «Main menu»).

A prompt for the date will appear below the «Main menu» (Figure 5.1). You must enter today's date in a precise day/month/year format (eg 01/02/85). Single digit days and months must be prefixed by a zero; only / can be used as a separator.

The date entered is then saved and you are left at the «Main menu» to select a new option.

Producing a Report

From the «Main menu» *cursor select* the Reports option. You will be presented with a «Query selection» menu (Figure 5.2).

```
Database name: British Trees          ESCAPE: Main menu
Query selection

LOGICNAME: Trees
Day Trees
Trees in the Family Satocedex
Trees taller than 15 metres
Trees found in hedgerows
List all records
Select individual records
Choose formats
```

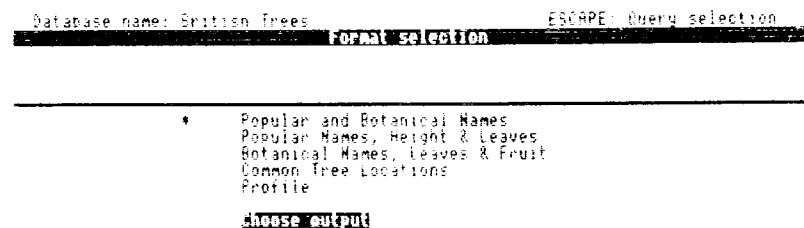
5.2 Query Selection Screen — British Trees

The «Query selection» screen has a number of noteworthy features. There is a list of five *query* names each followed by an ! *new query* flag signifying the database has not been *searched* for those queries yet.

The list also contains two standard queries List all records and Select individual records which are provided with the system. The numbers 44 and 0 after these indicate the number of *records* currently matching those special queries.

It would be useful to get a picture of the whole database from your first report. So, cursor select List all records then move to Choose formats and press **RETURN**. Note, your selection decision is flagged by a ★ symbol to the left of the query chosen.

The next screen is «Format selection». This presents a list of four report formats defined for the example database, plus a standard Profile option (Figure 5.3). Cursor select the first format Popular and Botanical Names and then leave this screen by selecting the detached Choose output option.



5.3 Format Selection Screen — British Trees

Cursor select the Screen option from the next menu, «Output Selection». As all the database records will match the query List all records, the database is not searched, and the report is presented immediately on the screen.

You will find your report is divided up into three 'pages' over three screens. The first page is reproduced as Figure 5.4. Each page is composed of a header followed by the information recovered from each of the records.

Database: British Trees		Page: 1
Query date: 04/09/85		Total: 44
Query: List all records		
Format name: Popular and Botanical Names		
Pop. Name	Bot. Name	Family
1 Ash	Fraxinus excelsior	Oleaceae
2 Beech	Fagus sylvatica	Fagaceae
3 Pedunculate Oak	Quercus robur	Fagaceae
4 Holm Oak	Quercus ilex	Fagaceae
5 Sessile Oak	Quercus petraea	Fagaceae
6 English Elm	Ulmus procera	Ulmaceae
7 Hornbeam	Carpinus betulus	Carpinaceae
8 Alder	Alnus glutinosa	Betulaceae
9 Crack Willow	Salix fragilis	Salicaceae
10 Goat Willow	Salix caprea	Salicaceae
11 Silver Birch	Betula pendula	Betulaceae
12 Common Birch	Betula pubescens	Betulaceae
13 Walnut	Juglans regia	Juglandaceae
14 Small leaved Lime	Tilia cordata	Tiliaceae
15 Sycamore	Acer pseudoplatanus	Aceraceae
16 Grey Poplar	Populus canescens	Salicaceae

Press SPACE bar

5.4 First Page of List all Records Output

The header contains the report date which should match that set by you, the query and format name, the total records in the database matching that query, and a page number.

The information is organised into a number of columns. Each column represents the information stored in a particular *field* and the field name appears above it. In this case the popular, botanical and family names are the fields requested by the *report format*. Reading across the report, each row of data represents a single record/species.

Each page of the report is held on the screen until you press the **SPACE BAR** as instructed. When the last page is cleared you are returned to the «Main menu».

Searching the Database

Following the same sequence of operations as above, cursor select Deciduous Trees from the «Query selection» menu.

Note, if you select a query in error just re-select the correct one; the ★ moves accordingly. The query Deciduous Trees aims to find all the records containing information about the different types of deciduous tree.

Proceed to «Format selection» and select Popular Names, Height & Leaves, then request Screen output again from «Output selection».

Because Deciduous Trees is newly-defined the database records must be searched before a report can be displayed. As the query is put to each record in turn, the search progress is shown by a count of the number of records retrieved (Figure 5.5). Note, a search is made for all the currently defined queries not just Deciduous Trees.

As it checks each record the system will keep an account of each one that satisfies the conditions specified by each query. The record lists matching each query are stored under the query name.

Searching the database. Record 44 of 44	
Deciduous Trees	(35)
Oak Trees	(3)
Trees in the Family Salicaceae	(5)
Trees taller than 35 metres	(12)
Trees found in Hedgerows	(16)

Press SPACE bar

5.5 Searching the Database Screen

The searching screen is held until you press the **SPACE BAR** as instructed. This allows you to leave the system during long searches and still get a picture of the success of the different queries before proceeding.

After searching you will find 35 records represent Deciduous trees. These will be displayed in the new format covering popular name, leaf description, mature height and the Decid/Ever field which should confirm that the correct records have been retrieved.

Profile Reports

The final example illustrates the system provided report Profile.

You should be at the «Main menu» after viewing the last report. Move to «Query selection» by choosing Reports. Note that some features of this screen have changed. Because all the queries were searched in the last operation the ! symbols have been replaced by numbers in parenthesis which signify the numbers of records matching each question (Figure 5.6).

Database name: British Trees	ESCAPE: Main menu
Query selection	
<hr/>	
Deciduous trees	(35)
Oak trees	(3)
Trees in the Family Salicaceae	(5)
Trees taller than 35 metres	(12)
Trees found in Hedgerows	(16)
List all records	(44)
Select individual records	(8)
Choose formats	

5.6 Query Selection Screen after Searching

Select Oak trees then proceed to «Format selection». Although you can only carry one query forward to this stage, you can display it in more than one format. Select Popular Names, Height and Leaves and Profile in turn. Note both of these have a ★ placed against them.

If you request screen output Oak trees will be displayed in the formats selected. Note Oak trees is reported without searching; the database is only searched when you report a new query.

Profile is a standard format supplied with the system which prints out all the fields of each record. Every page has a header, and each field appears on separate lines.

Further examples

After working through the examples above you should now have a feel for how SupaStore works through cursor selection from screen menus, and of the way reports are produced.

If you feel you need further practice at report generation try some different combinations of queries and formats provided with the example database.

SECTION 6 : Defining queries

This section explains how you can define your own *queries* for the example database. Up to ten queries may be defined at one time.

SupaStore Queries

The term query is used to describe a question that you want to put to each *record* of the *database*. Because the information within each record is organised into *fields* (see Section 2), a query always involves a check on the contents of one or more fields.

You have to translate your question into a form that the system can understand. SupaStore uses a special *command language* to define queries.

The simplest query consists of a FIELD COMMAND STRING sequence.

Oak trees used in Section 5 in fact takes this form. In command language it looks like 'Pop Name Contains Oak'. This query checks each record to see if the popular name field 'Contains' the *string* Oak.

Contains is only one of a number of *commands* available. It means that the string specified may occur anywhere within the field. The full command set is explained in the Command Appendix. When you begin to create your own queries it is important that you use this appendix to gain understanding of the differences between commands.

Clearly, the key to defining queries that successfully retrieve the records you want means; using the correct field/s, choosing appropriate commands, and care in specifying the string to be checked for.

Defining a simple Query

Consider now a new question related to the example database. It would be interesting to find all the trees that are native to Britain.

The translation of this question into a query is straightforward. The 'Native' field is clearly the one to use. Information stored in this field is coded into Y for native trees, or N for all others. Thus our query will need to check that 'Native Contains Y'.

Follow the starting procedure appropriate to your storage system (Section 5). Use the «Main menu» option Queries to reach a «Query menu» and then *cursor select* the Define a New Query option.

You must enter a query name within the [] symbols provided. This can be composed of any keyboard characters, but **Native Trees** is used and referred to here.

Enter the name suggested. The «Define a New Query» screen follows. This is divided horizontally into five screen areas (Figure 6.1). The topmost is a display area for the query as it is built up. The next two areas contain cursor-selection menus of the fields in the example database and all the SupaStore commands. The fourth area provides a location for string entry, while the bottom section contains *operators* to build more complex queries, and the option Save query.

Database name: British Trees		ESCAPE: Query menu	
Define a new query			
Query: Native Trees			
Fields			
Top. Name	Other Name	Bot. Name	Family
Decid/Ever	Native	Mature Ht	Leaf Descr
Fruit	Location	Use	Keywords
RECORD NO	DATE	(
Command			
contains	starts with	equal to	greater than
less than	doesn't contain	not equal to	
String			
[]		
and	or)	save query

6.1 Define a New Query Screen

The different screen areas are active in a precise order, following the FIELD COMMAND STRING rule. Initially, the cursor will be in the top right of the field area; you cannot move out of this area without cursor-selecting a field. Move to the Native field and press **RETURN**. 'Native' will appear in the display area and the cursor is positioned over the first command.

If you now select Contains this will be added to the display area while the cursor moves down for string entry. Enter **Y** in the space available. The string may be typed in upper or lower case as both string and field contents are converted to upper case when being matched during searching.

The query is now complete with the cursor in the bottom screen area. Now cursor-select the Save query option. This saves the definition in the system under the name 'Native Trees', adds the name to the current list of queries and returns you to the «Query menu» screen.

Finally, use the Return to main menu option; do not press **ESCAPE** for the «Main menu» because your defined query will be lost.

Using the Query

If you select Reports from the «Main menu» you will see Native Trees is installed in the «Query selection» menu with an ! *new query flag* beside it (Figure 6.2). You can proceed as in Section 5 if you wish to search and report the new query.

Database name: British Trees	ESCAPE: Main menu
Query selection	
Rectangular Trees (45)	
Oak Trees	(3)
Trees in the Family Salicaceae	(5)
Trees taller than 35 metres	(12)
Trees found in Hedgerows	(16)
Native Trees !	(44)
List all records	(8)
Select individual records	
Choose formats	

6.2 Query Selection Screen with a New Query Installed

Complex Queries

More complex queries consist of FIELD COMMAND STRING sequences joined by AND or OR *operators*. For example, to find all trees less than 40 metres high commonly found in hedgerows the query should read

Mature Ht Less than 40 and Location Contains hedgerows

To build a complex query, select either AND or OR after defining a simple sequence and the cursor will move to the field area to start another sequence. Terminate complex queries by selecting Save query.

In long complex queries you can ensure certain FIELD COMMAND STRING sequences are evaluated first during searching by enclosing them in brackets (see Appendix C).

Before starting a FIELD COMMAND STRING sequence you can select the opening bracket in the field area; the cursor will remain in the field area to allow you to select a field, command and string in turn. You can then use the closing bracket in the bottom area to close the sequence/s to be evaluated first.

The use of operators and brackets are illustrated by examples in the Command and Query Appendices.

Further examples

Suggestions for other questions that can be translated into queries and searched are included in the Botanical Appendix.

Query names

The query name is important. It will appear in the current list of queries in screen menus and on any reports based on that query. Accordingly, choose query names that suggest the nature of the enquiry and what the associated records represent.

Correcting mistakes

If you select the wrong field or command, or enter an incorrect string press the red **F2** function key. This will delete the last query element in the display area and the cursor will move back to the corresponding screen area to allow you to re-select that item. You can delete as many elements as necessary by continued use of **F2**.

SECTION 7 : Defining report formats

This section explains how you can define your own *report formats* for the example *database*. Up to ten formats can be defined at one time.

Advantages of Report Formats

From Section 5 you will know that *reports* can be displayed in a variety of ways. While only one *field* need be used in a query, reports can provide much more information. This flexibility stems from being able to select different fields for output by using pre-defined reports able to select different fields for output by using pre-defined report formats.

A report format defines the list of fields and other items to be displayed across the report. Each line in a report represents these fields for a single *record* while, looking down the report, fields appear as columns (Section 5, Figure 5.4). fields can be chosen in any order, and any field may be selected more than once.

Defining a new Report Format

A format covering the use of the trees will be defined as an example. Clearly this requires the *Use* field but, you might also include the record number and popular names for reference, and perhaps the height which may be related to commercial viability.

Starting at the «Main menu», *cursor select* the *Formats* option, then Define a New format from the following «Format menu» screen. Next, enter a format name between the [] symbols provided.

Database name: British Trees ESCAPE: Format menu

Define a new format

—Format name: Ways in which trees are used— Line length: 80

—Selected fields—

Pop. Name	Other Name	Bot. Name	Family
Decid/Ever	Native	Mature Ht	Leaf Descr
Fruit	Location	Use	Keywords
RECORD NO	DATE	SET TABS	EXIT

7.1 Define a New Format Screen

Enter the name Ways in which trees are used. You will encounter the «Define a New Format» screen next. This is divided into an upper display area where the format is built up, and a lower menu area containing the fields and two other options SET TABS and EXIT (Figure 7.1). In addition, RECORD NO and DATE allow you to include the record number and the date when the record contents were last updated.

Try cursor selecting the following fields; RECORD NO, Pop Name, Use and Mature Ht. As you press **RETURN** over each field a list of their names appears in the display area.

It is important to realise that, initially, only ten spaces are available to print the information contained in each field. This space is equal to the standard field name length. The total width of the report in characters is stated at the top of the screen; this includes a space between each name.

Because the relation between the field name length and the actual field width is clearly important, the latter is indicated in the display. If part of the field name is displayed in *inverse colours* it means the field is shorter than ten characters. In fact the actual field length is equal to the width of the inverse block. An entire field name displayed in inverse colours signifies that the actual field length is equal to, or more than, ten characters.

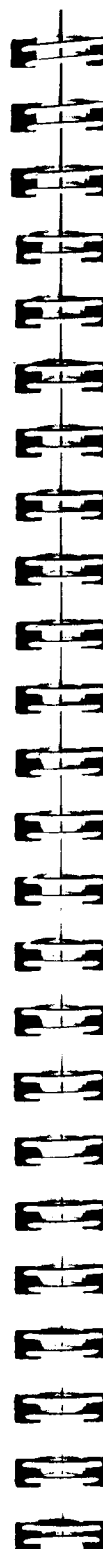
If all the fields in your format are shorter than ten characters it should be acceptable because no information will be lost when displaying the report. If this is so the user would select the Exit command to save the format and return to the «Format menu».

However, some fields (eg Pop Name, Use) will be wider than ten characters. To print the contents of these in full you have to adjust the space available for each field in the format.

Try selecting the SET TABS option. This transfers control to the display area itself where you can adjust the location of each field across the report like a typist sets tab positions on a typewriter.

You can move across the field list by pressing the **←** and **→** arrow keys. Try moving to the Pop Name field which is clearly longer than ten characters. If you now repeatedly press **SHIFT** and **→** together the space available for that field can be expanded up to the actual field width as indicated by the inverse block in the display. Note the total report width is increased accordingly.

If any fields in your format are less than ten characters there is some merit in reducing their width to make space for long ones. To do this simply move the cursor over a short field and press the **SHIFT** and **←** keys together. Try shrinking the RECORD NO and Mature Ht fields in your current list.



You should note however that the field names used for column titles in the report are as they appear in the display and reduction for very short fields will truncate field names with varying effect; for example RECORD NO is still sensible as REC, but Mature Ht will lose more meaning by reduction.

When you are happy with the layout of the report format press **RETURN**. The cursor will appear over the SET TABS option; move to and select EXIT to return to the «Format menu» then cursor-select Return to Main menu to reach that point. On doing so the format will be saved in the system and its name added to the current list of formats.

Don't use **ESCAPE** to reach the «Main menu» unless you want to prevent a format with mistakes in it being saved.

Report Format widths

When defining report formats you should always bear in mind the width of the screen or printer being used. Screen width is fixed at 79 characters but widths possible on printers will depend on printer type, paper width and the size of characters being printed (see Section 21).

Format names

Report format names should also be given some thought. With consideration they can summarize the fields selected for output. However, because formats may be used with any number of queries the name need not be linked with any one query.

Correcting mistakes

If you select a field in error in «Define a new format», press the red **f6** function key. This deletes the last field in the format list. Any number of fields can be deleted from the rightmost field backwards in this way.

SECTION 8 : Sorting report records

All reports so far have presented the records in the order they are stored in the database, viz by record number. On some occasions you may want to re-arrange or 'sort' the records in a report.

Sorting features

You can sort the records in a report according to the contents of one, or up to five, fields. To do this the user defines a new report format (Section 7) and assigns a sort code to the field that is to be used for sorting.

Whenever you report a query with this format the system will find the sort code message and use the contents of the encoded field to sort the records associated with that query.

Sorting works with both numerical and alphabetical fields. Fields containing both numbers and letters are sorted by numbers first. You may also specify sorting in ascending or descending order (See Appendix F).

Sorting is an especially useful feature when queries retrieve large numbers of records. It will organise your output so that a basis for revising your query to retrieve fewer records may be apparent.

Defining a Format with a simple sort.

As an example, it may be interesting to show how records can be sorted into groups with the same family name.

Proceed to the «Format menu» and select Define a new format. Enter a name along the lines of **Names by Botanical Family**. Then select the following fields RECORD NO, Family, Pop Name, Bot Name finishing with SET TABS. Note the format name indicates that the format involves sorting.

You should have control in the display area. Adjust the fields so that the format will print them in full (see Section 7).

Now move the cursor over the 'Family' field and press the **F4** function key to encode it with a sorting message. The system acknowledges this by placing a ^1 symbol under the field in the display.

If you encode the wrong field by mistake, press **F4** again and the ^1 symbol will disappear.

If you are happy with the layout of the format, and the correct field is encoded, press **RETURN** and select EXIT to save the format in the normal way.

Producing a Sorted Report

Try reporting the Deciduous Trees query with this new format following the steps in Section 5.

After selecting Screen from the «Output selection» menu you will get a 'Sorting, please wait' message. The sort message in the format has been found and the records are being sorted. The displayed report is organised in the usual manner, but the records are ordered according to the contents of the Family field. Note, the order is alphabetical, the trees in Aceraceae come first, followed by Aquifoliaceae and so on (Figure 8.1).

Database: British Trees			Page: 1
Query date: 04/09/85			Total: 44
Query: List all records			
Format name: Names by Botanical Family			
REC	Family	Pop. Name	Bot. Name
14	Aceraceae	Sycamore	Acer pseudoplatanus
17	Aceraceae	Field Maple	Acer campestre
21	Aquifoliaceae	Holly	Ilex aquifolium
31	Araucariaceae	Chile Pine	Araucaria araucana
7	Betulaceae	Alder	Alnus glutinosa
10	Betulaceae	Silver Birch	Betula pendula
11	Betulaceae	Common Birch	Betula pubescens
28	Caprifoliaceae	Elder	Sambucus nigra
6	Carpinaceae	Hornbeam	Carpinus betulus
18	Corylaceae	Hazel	Corylus avellana
32	Cupressaceae	Leyland Cypress	Cupressocyparis leylandii
1	Fagaceae	Beech	Fagus sylvatica
2	Fagaceae	Pedunculate Oak	Quercus robur
3	Fagaceae	Holm Oak	Quercus ilex
4	Fagaceae	Sessile Oak	Quercus petraea
20	Fagaceae	Sweet Chestnut	Castanea sativa

Press SPACE bar

8.1 Report sorted by Botanical Family

Sort flags

When the «Query selection» and «Format selection» menus are next used you will find that an 'S' sort flag appears next to Deciduous Trees and Names by Botanical Family (Figures 8.2 and 8.3). These are placed to remind you that these two options were the last to be combined in a sorting operation.

Database name: British Trees		ESCAPE: Main menu
Query selection		
1		
Deciduous trees (45)		
Oak trees (3)		
Trees in the Family Salicaceae (5)		
Trees taller than 35 metres (12)		
Trees found in Hedgerows (16)		
Native Trees (30)		
List all records S (44)		
Select individual records (0)		
Choose formats		

8.2 Query Selection Screen with a Sort Flag

```

Database name: British Trees      ESCAPE: Query selection
Format selection

Popular and Botanical Names
Popular Names, Height & Leaves
Botanical Names, Leaves & Fruit
Common Tree Locations
Ways in which trees are used
Names by Botanical Family      S
Profile

Choose output

```

8.3 Format Selection Screen with a Sort Flag

If you report Deciduous Trees again with the same format the system recognises the sort flags and does not sort the records again unnecessarily.

If you report any other query with the sorting format, the 'S' flag will be placed against that query. 'S' flags are cleared if either the associated query or format is revised (Section 9), or if records matching that query are added or changed (Sections 11 and 12).

Hierarchical Sorts

Up to 5 fields may be used for sorting the records in a report. After you have encoded the first sort field, another four fields can be specified by moving to them in turn and pressing **F4**. The symbols ^1, ^2, through ^5 are placed underneath to indicate the order in which the fields were selected. The full range of sorting symbols are listed in the Format Appendix.

Assume a report format encoded with Family then Mature Ht has been saved. Reports in this format will have their records sorted into family groups, and then the trees in each family will be sorted by their mature height.

Ascending and Descending Sorts

Whenever you encode fields for sorting it is assumed that you want to sort in ascending order (ie 0 to 9, A to Z). However, if the cursor is still over that field you can press the **F5** function key to change this to a descending sort. The symbols in the display change accordingly from ^1 to v1 and so on. Hierarchical sorts may use a mixture of ascending and descending fields.

Justification Codes

Justification is particularly useful for aligning columns of numbers and the Format Appendix provides an example. You can encode a field with justification codes. These are assigned to fields in the same way as sort codes but by pressing the **F3** function key. A second press of **F3** removes these set in error.

SECTION 9 : Revising queries and formats

Occasionally you may find that a *query* or *format* is inappropriate. The query may contain an unsuitable *field* or *string* (see Section 6) causing the wrong *records* to be retrieved. Equally, a format may need fields added, or even deleted if the *report* is too wide (Section 7). In either case you can revise the query or format instead of defining a new one.

Revising Queries and Formats

The steps in revising queries and formats are very similar. The table below shows the relevant menus and options you should *cursor select* to reach «Query revision» and «Format revision» screens respectively.

Queries		Formats	
«Main menu»	Queries	«Main menu»	Formats
«Query menu»	Revise a query	«Format menu»	Revise a format

The «Query revision» and «Format revision» screens both contain a list of the currently defined queries or formats.

If you cursor-select any query/format from this list the current name will be displayed in the lower part of the screen. You can retain the same name by pressing **RETURN**, or you can remove it using the **DELETE** key and enter a new name.

In query revision you will next see the «Define a new query» screen with the query definition already in it. You can now extend the query, or delete its elements (from the last backwards) with the **F2** function key then re-define the sequence removed (Section 6). Select Save Query last as usual.

If you are revising a format you will next see the «Define a new format» screen with the list of fields already in it. You can add new fields if you wish, or you can delete the existing ones first (from the last backwards) with the **F6** key. Then select SET TABS to adjust the field spacing of the revised list and alter *sorting/justification codes* as necessary (Section 8). Leave this screen by selecting EXIT.

After revising a query or format you will be returned to the «Query revision» or «Format revision» screens as appropriate, where you can revise another query/format, or select the detached option to return to the «Query menu» or «Format menu».

Remember, unless you want your changes to be ignored and the query/format saved as it was before, do not use the **ESCAPE** key to return to the «Main menu».

Changes in Menus after Revision

Note, if you revise a format or query that has been used in connection with a sorted report, the 'S' *sort flags* are removed from the «Query selection» and «Format selection» menus.

If you revise a query that has been searched, the record list associated with it is deleted and the query is marked as unsearched with an ! *new query flag* in the «Query selection» menu.

Recalling Query definitions

Because the query name is the only representation you have of a query, it is easy to forget its FIELD COMMAND STRING organisation. To recall this simply go through the actions of revising the query without making any changes.



SECTION 10 : Deleting queries and formats

Up to ten *queries* and *formats* can be stored in the system at one time. If you attempt to define any more you will get a message like 'All queries (formats) are used. Delete one and try again'.

This section explains how you can make room for new queries or formats by deleting existing ones from the current menus.

Deleting Queries and Formats

The steps in deleting a query or format are the same as for revising a query or format. Only the specific options required to reach «Query deletion» and «Format deletion» screens differ. These, and the screens they are selected from, are shown in the table below.

Queries		Formats	
«Main menu»	Queries	«Main menu»	Formats
«Query menu»	Delete a query	«Format menu»	Delete a format

The «Query deletion» and «Format deletion» screens both contain lists of the currently defined queries or formats. Note, the List all records and Select individual records queries, and the Profile format do not appear on these lists; they cannot be deleted.

If you *cursor select* a query (or format) it will dissappear from the list, but will not be deleted permanently until you use the Return to query menu OR Return to format menu menu option.

If you delete the wrong query by mistake, simply press **ESCAPE**. This will take you back one step and abandon all the deletions requested.

Note, you remain in the deletion screen so that further queries or formats can be deleted.

Query and Format Libraries

When you use the options above the queries and formats are permanently lost. However, you can remove queries (or formats) from the current lists temporarily by *archiving* them in a *library* and retrieve them from this when they are needed (Section 18).

SECTION 11 : Changing the information stored in records

Information stored in *databases* rarely stays the same; there is usually a continual need to keep the information up to date.

You can change or edit SupaStore *records* using an *editor* accessed by the *Edit option* on the «Main menu».

Before an editing session, it is good practice to use the «Main menu» option Set today's date (see Section 5).

Changing a record in the Example Database

The need to update the records in the British Trees database may not be so apparent. After all botanical names are unlikely to change.

However, mature heights are more open to debate since they can vary with soil type and location. As an example, assume that you have found a reliable source which indicates that the mature height for the Sessile Oak is 30 metres rather than the 40 stored. The way you can change this value is explained below.

The Editor screen

Cursor select the Edit option from the «Main menu». A new screen invites you to; 'Enter the number of the record you wish to edit or just press **RETURN** for the next free record'.

You know from *reports* that the Sessile Oak is stored in record 4. Enter 4 and this record will appear (see also Figure 4.1).

At the top of the screen are the record number, the date that this record's contents were last changed, and the screen page number. Below this are the *fields* prefixed by their names.

Making changes in Fields

A *field-cursor* in the editor text colour fills the first field on the screen. This can be moved between fields with the **↑** and **↓** arrow keys. Try moving down to the 'Mature Ht' field; note the cursor changes to fill each field on the way.

Within a field, the **←** and **→** keys move the small flashing *editor-cursor* to where unwanted information can be removed with the **DELETE** key, or new characters can be inserted.

Move the editor-cursor past the current value of 40 in the 'Mature Ht' field and delete it. Now enter 30; after you press **RETURN** the field cursor will move to the next field.

Saving your changes

You must leave the record you have been editing in one of three ways to permanently save the changes made.

i) Press **CTRL** and **↓** together to go to the next record. If you were changing the last record in the database the next record will be a blank *free record* (see Section 12).

ii) Press **CTRL** and **↑** together to go to the previous record. Note, you do not move back if you are at record 0.

iii) Press the **F0** function key for a record number prompt in the top left of the screen. Enter any valid number to move to that record in the database. If you enter an invalid number you will be re-prompted.

F0 is most useful if you have a number of records to change. If you are only editing one record you still must do one of these three actions to save the changes.

If you don't wish to save the changes you've made to the current record press **ESCAPE** which will return you to the «Main menu».

Leaving the editor

When you have finished editing, ensure you have left the last record you changed, and press **ESCAPE** to return the «Main menu».

Other Cursor Movement Features

By pressing **SHIFT** and the cursor arrow keys together you can move about a record more quickly. If you have a single-screen record, **SHIFT** **↓** will move the field cursor to the bottom field; **SHIFT** **↑** moves it to the first field.

If the record covers more than one screen **SHIFT** **↓** takes you to the first field in the next page, while **SHIFT** **↑** moves the cursor to the last field on the previous page.

Tracing Edits

It can be useful to trace records you've changed. When an edited record is saved, a date stored with the record is changed to the one currently stored in the system (this is why it is important to set today's date before edit sessions).

You can then define a query using DATE in the field menu to find records you edited on, before, after or between certain days.

Updating of Queries during Edits

A useful feature of SupaStore is the automatic updating of queries. If any changes made to a record answer an existing query the record is added to the list saved under the query name and the total in «Query selection» is increased. Equally, if the change means a record no longer satisfies a query it is deleted from saved lists.

Thus, there is no need to *search* the database after an editing session to ensure query results are accurate.

Network Users

Network users should note that they can only make full use of the features described in this section with databases they 'own'. If they access other user's databases as 'non-owners', they can use the editor to view the record contents but cannot make any changes (see also Section 23).

SECTION 12 : Adding new records to the database

New records are always added at the end of SupaStore *databases*. A blank *free record* is maintained for new data after the last complete *record*; after this is used and saved the database is extended by creating a new free record (Figure 12.1).

(a) Adding Records to a Database
Database of 20 Records

Next Free Record = 20

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----

Records Searched: = 20

12.1 Adding Records to a Database

There are 44 records in the British Trees database. However, because SupaStore databases start with record 0, the last record is in fact number 43, and the next free record is 44.

The Botanical Appendix at the end of this guide contains data for two species that aren't in the database. This section explains how these can be added as new records.

Accessing the next free record

Select Edit from the «Main menu» and press **RETURN** after the prompt 'Enter the number of the record you wish to edit or just press **RETURN** for the next free record'.

The free record 44 will appear with empty fields. As with complete records, a large *field-cursor* and a small flashing *editor-cursor* will be located in the Pop Name field (Section 11).

Completing Records

Anything entered from the keyboard will be put into the field occupied by the field-cursor. The amount of information that can be entered is limited to the field length indicated by the [] symbols.

Make record 44 the Aspen. Enter the details from the Botanical Appendix sheets into the fields. You can move the editor-cursor across the entry with the **←** and **→** keys to correct any mistakes.

After finishing a field entry press **RETURN** or **Y** and the field-cursor will move to the next field. See Section 11 for other field-cursor movement features.

Saving a new record

To save the new record and create a free record press **SHIFT** and **COPY** together. Your screen will display the new free record 45.

You can now press **ESCAPE** to return to the «Main menu», or go on to complete and add another new record.

Add the data for the White Poplar in record 45, save it with **SHIFT COPY** then press **ESCAPE**.

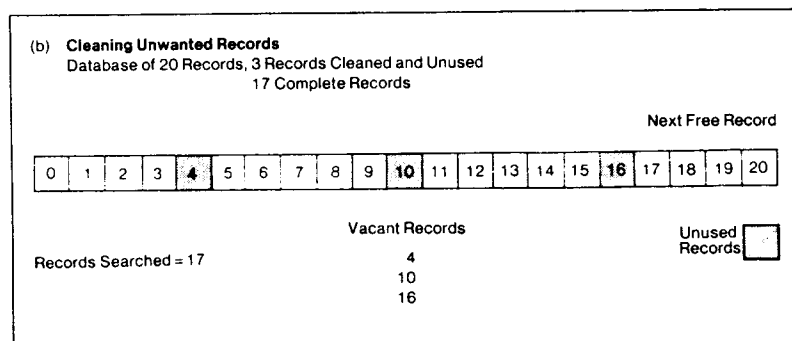
If you don't wish to save a new record full of mistakes, press **ESCAPE**. This will return you to the «Main menu» keeping that record as the next free one.

Re-using unwanted records

In some situations existing records may become redundant. If so, you can re-use them by entering completely new information.

To avoid tedious changes field by field, the entire contents of the current record may be wiped by pressing the red **f1** function key. To prevent accidental erasure, a prompt at the top of the screen will ask you to confirm your intention to clear the record with Y, or decline with N.

If you re-use the record immediately use **SHIFT COPY** to save the changes you have made.



12.2 Unused Records in a Database

If you don't re-use the record immediately the system will ignore it during searching. The record retains its number however, and exists as an empty record in that position in the database (Figure 12.2). When a number of records cleared by **f1** remain unused, the total records declared during searching (Figure 5.5) will be less than the number of the last complete record in the database.

You can find out what records are vacant by using the database description utility (Section 16).

Automatic updating of Queries

If a new record answers an existing query this is added to the record list saved under that query name and the total in «Query selection» is increased.

For example, the addition of both the Aspen and the White Poplar should have increased the record count next to the queries Deciduous Trees and Trees in the Family Salicaceae by two, increased Native Trees by one, and brought the total record count beside List all records to 46.

Equally, if you wipe a record's contents using **f1** this is removed from the query lists that it matches and totals in «Query selection» are reduced accordingly.

Network users

Network users should note that they can only make full use of the features described in this section with databases they 'own'. If they access other user's databases as 'non-owners', they can use the editor to view the record contents but cannot make add any new records (see also Section 23).

SECTION 13 : SupaStore utilities

The main objectives in using SupaStore are the *searching of databases* and the production of *reports*. These have been fully covered by the preceding sections.

However, the system is enhanced by a set of *utilities* which provide advanced features. These additional facilities are introduced here along with an explanation of the relationship between the utilities and the SupaStore software on EPROM. Most of the remaining sections in this Guide deal with utilities in detail.

Accessing the Utilities from BASIC

Access to the utility programs will vary according to the storage system in use.

FLOPPY DISK USERS — The SupaStore pack is supplied with utilities on a 40 track *floppy-disk*. If you will be working with an 80 track system you will need to create an 80 track Utilities Disk following the instructions in the separate sheet provided with the pack.

If you haven't been using SupaStore initially, floppy disk users can *autoboot* the Utilities Disk. Put this in the drive (use drive zero if you have a dual system) and press the **SHIFT** and **BREAK** keys together, releasing **BREAK** first. The «SupaStore utilities» menu will appear.

HARD DISK AND ECONET USERS — The network manager should have installed the utility programs in a \$.SUPASTORE directory. Ensure you are in your own SUPASTORE sub-directory and enter **★SSUTILS**. The «SupaStore utilities» menu will appear (see also Section 23).

Accessing the Utilities from SupaStore

If you have been working with a database, the utilities can be accessed from the SupaStore «Main menu».

HARD DISK and ECONET users can simply enter **★SSUTILS** from the keyboard to be transferred to «SupaStore utilities» directly after pressing **RETURN**.

FLOPPY DISK users must enter **★BASIC**, then autoboot the utilities as explained above.

The SupaStore Utilities menu

The «SupaStore utilities» screen asks you 'Which utility do you want?'. You respond by *cursor selecting* from the menu provided. The menu options are replicated below along with the section that deals with the use of that option in detail.

Option	Section
Define a new database	: 15
Modify configuration	: 15
Edit selected records	: 17
Archive or retrieve formats	: 18
Archive or retrieve queries	: 18
Display database description	: 16
Create a sub-database	: 20
Label definer and printer	: 19
Access another user's database†	: 23
Enter supastore	: 13
Quit	: 13

† Note; this option is only available to network users.

Leaving the Utilities

When you have finished a utilities option you will be returned to «SupaStore utilities». You can now; use another utility, leave the utilities by selecting Quit, or transfer directly to SupaStore by selecting Enter SupaStore. If floppy disk users choose the latter option, they should ensure they insert an appropriate database disk first. By default, selection of Quit will return all users to BASIC, although Econet users may find their network manager has arranged for direct transfer to other network facilities.

Special considerations for floppy disk users

When using the Utilities Disk you will need to have readily available a database disk or blank, *formatted disks* to receive newly-created databases, *libraries*, and label files. You should read the introductions to relevant sections carefully and ensure you have these disks prepared.

You will have to exchange the Utilities Disk with other disks from time to time. The system gives clear screen prompts when necessary and only allows you to insert a disk of the required type.

When exchanging disks wait for the 'Press SPACE bar' instruction and always wait until the disk drive has stopped spinning before removing a disk.

SECTION 14 : New databases – preliminary considerations

One of the most powerful SupaStore features is the ability to create any number of new *databases* that can be managed by the one system.

Before you use the database creation *utility*, you should however consider a number of factors in advance. This section is designed to advise on and encourage such preparations.

Record definition

With new databases you need to decide unambiguously what the *record* is to represent. When this is clear, you can decide upon the information you want to store and how this can be organised into *fields*.

The subject of the example database was 'British Trees'. Accordingly, each record corresponded to a single species.

Database on apparently the same subject may have different record definitions to be asked. For example, in comparison to the example database, a forester with commercial interests in trees might make each record a plantation, with fields covering its location, the species growing there, the number of trees, their age and so on.

However, it would be wrong to imply each record in a database has to measure the same thing. Record definition could be something general like 'An object found on the beach' so that one record might be a bucket, another a shell, another seaweed and so on. However, the fields used would have to be equally general (eg colour, size) covering things that all the objects can be measured by.

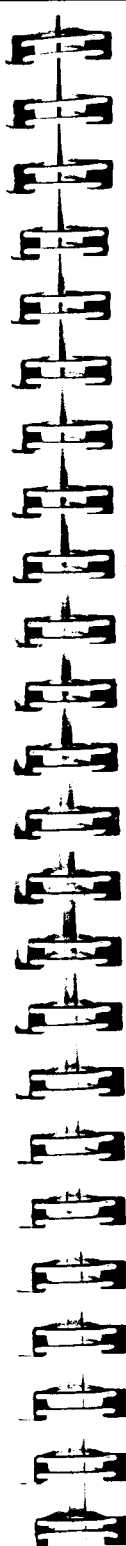
Field decisions

All SupaStore records must be organised into fields. Most of the advance preparations discussed below involve a consideration of fields.

The fields in the British Trees database arose from tree attributes that best illustrated the functioning of the system because this was the underlying aim.

Your aims for using any new database should guide decisions about what fields to use. In particular, think ahead to the sort of questions you want to ask.

Decisions involving fields concern their number, type, length and contents. The advice below covers these areas, but also stresses the need to balance record size against database size.



Pilot data

Before you create a database, you should assemble data for as many records as possible and use these to determine the type and maximum length of your field.

However, if you plan to add more records later, you should consider future entries and ensure fields are sufficiently long to accommodate them.

Deciding on a list of fields

Many simple fields are preferable to a small number of complex ones each trying to store too many things.

There is no advantage in storing information on a number of aspects in a single long field. Separate fields use the same storage space and make for flexible reporting because contents can be output separately.

Start with a list of the things describing the record subject which you would want to report separately. Make these your fields and decide on names (up to ten characters) which indicate the field's function.

Although it is not impossible to add overlooked fields later on (Section 20) it is an idea to include a 'Keywords' or 'Comments' field in your original list. This can be used to store general *keywords* for retrieving records by queries not covered by the other specific fields. Such a field was included in the example database.

Up to 250 fields are allowed per record, but limits on record size may reduce this if you have a number of large fields (see below and Appendix G).

Deciding on field types

You have to decide on an *input type* for each field. This defines what keyboard characters can be entered through the *editor*.

Eight input types are defined by the single-character *input codes* in the table below.

- A – Any keyboard characters.
- O – All characters except digits.
- L – Letters and space.
- P – Telephone Numbers.
- Y – Y or N for Yes or No responses.
- I – Integer numbers.
- F – Floating point numbers.
- D – Dates in dd/mm/yy format.

The simplest (A) allows entry of any keyboard character, and there is no reason why all the fields in a record shouldn't be of this type.

However, there is some merit in using input types which disallow certain characters (O, L, P, Y). Such data validation is useful if someone unfamiliar with expected field contents will be entering the information.

Other special types (I, F, D) benefit from storage advantage and are treated in a special way in operations such as sorting.

Field contents and field length

A field can be anything between 1 and 255 characters long. The length decided upon is obviously determined by what you intend to store in it. Careful consideration of alternative representations of field contents can contribute to significant economies in record size.

Some fields from the British Trees database can illustrate different field contents and how they influence field length.

The contents of Pop Name was clearly different in every record. To estimate its length in advance the longest name had to be determined. It is especially important with fields like this to think about the size of data to be collected and added in the future.

In contrast, Ever/Decid was designed as a keyword field. It contains only one of two keywords (Evergreen or Deciduous) which occur in every record. There were no other possible entries so the required length was obvious.

We were only interested in whether the tree type was native or not. Hence the Native field implied only two outcomes, Y or N. Making use of the Y input type meant this field only needed to be one character long.

Leaf shape was also a keyword field, but in this case the leaves of each tree type were distinguished by a combination of keywords; (see Appendix A). Here maximum field length had to accommodate the longest combination of shape keywords. In fields like this, keep the order of keywords consistent, otherwise you will get unexpected results when the field is used for sorting.

If you use keyword fields compile a list of keywords as you collect pilot data, and ensure entries are consistent throughout the database. Thinking about keywords assists database organisation, and helps in *searching* because keywords provide obvious strings in *queries*.

In reality keyword fields could contain full text. For example, entries such as Parks Gardens in the Location field could read as Found in Parks and Gardens. All words are treated as keywords and queries would still find the separate *strings* Gardens and Parks.

One advantage of using full text is increased readability when the field is included in a report. However, this has to be weighed against the increase in record size and the fact that long fields will reduce the number of fields possible in a report.



Thinking ahead to Queries and Formats

It was mentioned above that your aims should guide database definition. As these will be orientated towards asking questions and producing reports it is important to consider in advance the queries and formats required.

Firstly, ensure you don't disbar any questions. For example, the Y/N Native field was adequate for indicating whether a species was native or not, but what if the country of origin of alien trees was of interest? It would have been necessary to ensure the field was large enough for Native and other country name keywords, and to change the name to something like Origin.

Because query strings cannot be longer than 20 characters, avoid very long keywords and consider how you would retrieve any longer entries you have. In reality, twenty characters should be sufficient to define a unique string, and this didn't prove a problem even when considering the long botanical names in the example database.

Long, full-text fields may increase legibility, but shouldn't mean all your reports will be based on single fields. If some long fields have to be reported in isolation you should ensure there is room in the report for an index item like RECORD NO so that the information can be matched with other reports.

When deciding on field types and contents consider whether they will be primarily used for searching or reporting. If fields are only used for searching you can have very short fields containing codes instead of keywords. The more fields are expected to be used in report formats, the more readable their contents should be.

Design

Database creation also requires the design of field layout on the editor screen. This is also best done in advance. A sheet on which to try designs, which you can freely copy, is provided in Appendix I.

When designing the editor screen, it may be useful to bear the following points in mind.

Put fields that help to index the record, (eg names, your own index numbers), or fields that are likely to be updated frequently, at the top.

Group fields with common functions together and separate them with blank lines.

Use the space economically by putting several short fields on one line, but don't cram everything into one page. Use of more than one page doesn't take up any more storage space, so use the facility freely.

Database size

The following restrictions on database size apply only to users who will store databases on *floppy disks*.

Floppy disk drives may be single or double-sided, and 40 or 80 track; also users may have single or dual drives. These terms effectively measure storage capacity and the type of drives attached to your computer will determine the largest database you can create.

However, don't create larger databases than necessary that use all the storage capacity available. For instance, there is no reason why someone with a dual, double-sided 80 track drives cannot create single-sided databases if they provide sufficient records.

Free disk drive capacity is in fact required by the *spool file* output (Section 22) and sub-database creation (Section 20) facilities, and is only available to users with dual drives for this reason. If you have a dual system, and these options are useful to you, you should only create single-drive databases.

When starting a new database you are required to specify whether your disk drives are 40 or 80 track, and whether you want to use single or double-sided disks, and single or dual drives. The database created is then *configured* to that specification.

Deciding on database configuration

Floppy disk users only — The Database Appendix explains how to estimate record size, and a table indicating database size under different configurations for a range of record sizes. Calculate record size for your proposed database, and use the table or graph to estimate the smallest configuration which will give you sufficient records.

You may find that your field specification was ambitious and that the minimum configuration of your storage system will store a database with too few records. If so, you will have to reduce the length of some fields and re-think their contents.

Preparing disks (Floppy disk users only)

Standalone users should format in advance the blank floppy disk/s that will store the database. A table in Appendix G indicates which sides of the disks should be formatted for different configurations.

SECTION 15 : Creating new databases

Checklist

Before you create a new *database* you should have;

- i) Defined what a *record* represents.
- ii) Collected together some pilot data.
- iii) Decided on a list of *fields*.
- iv) Fixed the name, length and *input type* of each field.
- v) Decided on database size.
- vi) *Floppy disk* users should have formatted the required number of blank floppy disks.
- vii) Designed the layout of the *editor* screen.

Practical advice on making these decisions is given in Section 14.

Configuration and Design

The creation of a new database is a single operation which has two stages.

You must firstly specify the database *configuration* which is then saved in a configuration file. For floppy disk users this involves stating the disk drive characteristics which match your own and provide a database of the required capacity.

All users must specify the type of printer *interface* they have, printer paper page lengths, and the screen colours they want SupaStore to operate in. However, Hard disk and ECONET users should note that they only need to do this once. If a configuration file exists on their SUPASTORE sub-directory, database creation begins with the second stage. A facility is provided to allow all users to modify their configuration specification (see below).

A second design stage involves giving the database a name, specifying the field characteristics and deciding how the records will appear in the editor.

Configuration Menus

Access the utilities by a means appropriate to your storage system (see Section 13) then *cursor select* Define a new database from the «SupaStore utilities» menu.

You must specify the configuration by cursor selection from a succession of screen menus. Floppy disk users have to make decisions in all of the following areas. However, *hard disk* and *Econet* users are not asked questions a to c.

- One or two disk drives?
- Single or double sided drives?
- 40 or 80 track drives?
- A serial, parallel or network printer?
- Page length of printer paper (in lines)?
- Menu text colour?
- Menu background colour?
- Editor text colour?
- Editor background colour?

If you specify a serial printer, an additional menu will appear requiring you to cursor select its *baud rate*.

Note, the *cursor* lies over *default* options in each menu. If you make a wrong selection press **ESCAPE** to go back to the previous menu.

Saving the Database configuration

The following disk exchange instructions apply to floppy disk users only. The configuration is saved automatically for hard disk and Econet users.

After the last menu you must exchange the Utilities Disk for a blank formatted 'database' disk. If your database is to be stored on two disks, also put the other blank disk in drive 1. When you press the **SPACE BAR** as instructed, the configuration data will be saved.

Follow the instruction to return the Utilities Disk to drive 0. Design stage programs are loaded into the computer when you press the **SPACE BAR**. Finally, return the database disk to drive 0 and proceed by pressing the **SPACE BAR** again.

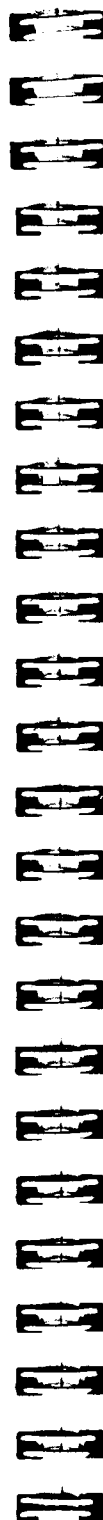
Creation Date and Database Name

A new screen will prompt you to enter the current date and the database name in turn. The date, which must be in dd/mm/yy format, is installed as a creation date. The database name can be up to 15 *alphanumeric* keyboard characters. Hard disk and Econet users are also required to enter a shorter 7 character name.

The Design Screen

The following design screen is blank but for statements of the database name, record size, the number of usable records and page number. There are also entry points for field length and input type.

As fields are defined the statement of record size increases, and usable records changes to the number of records of that size possible within your configuration. Thus the user always has an indication of the whether the pre-conceived fields will allow a sufficiently large database.



Field specification and the Record layout

The user defines each field in turn by positioning them, and their names, within the blank screen area. On completion this screen is saved as the input screen for the editor.

Following the screen design prepared (Section 14) move the small flashing cursor with the cursor arrow keys to where a field name is to begin and press **RETURN**.

A ten-character box will appear. Any field name you enter will be placed in the box.

You must now enter the values decided for the field length and *input codes* (see Section 14 or Database Appendix). Your entries will be confirmed at the top of the screen. After entering a field length a representation of the field between [and] symbols is placed after the field name.

Repeat these steps until all of your fields are specified.

Using more than one screen

There is no need to crowd a large number of fields into one screen. If you fill the first screen move to Page 2 by pressing **SHIFT** and **↓** together. To move back a screen use **SHIFT** **↑**. If necessary, up to seven screens can be used.

Making changes

Having completed the record, you may find you have been too ambitious, leaving the database with too few usable records. If so, you can change the size, type, or even position, of the existing fields.

To alter any field, move the cursor under the first character of the name and press **RETURN**. You can now change the existing name (by using **DELETE** and retyping), and re-enter the field length and input code.

If you want to delete a field (perhaps to redefine it elsewhere in the screen) move the cursor under the first character and press the red **f6** function key.

Saving the Database

When you are satisfied with the screen layout and field definitions press the **SHIFT** and **COPY** keys together. Floppy disk users should exchange the utilities and database disks in drive 0 as instructed.

All your definitions will be saved as part of the database, and a number of data files will be mapped over the database disk/s or hard disk/Econet directories. This takes a little while; do not remove the disk/s until you are returned to «SupaStore utilities» or are instructed to replace the Utilities Disk.

Adding Records

A newly created database will have a empty record 0 available. Use the editor to enter data into this and subsequent records (Section 12).

Modifying configurations

Users may need to alter configuration information relating to databases if they acquire a new printer, or find screen colours inappropriate. In addition, network users may need to switch between local and network printers. Selection of the Modify configuration option in «SupaStore utilities» allows you to reselect printer and screen colour options as above, but returning you to «SupaStore utilities» without having to redefine fields and layout. Note, the cursor lies over your previous settings; to keep some of these simply press **RETURN**.

SECTION 16 : Database description

The decisions made when *configuring* and designing a *database* (Section 15) are effectively hidden once it is in use. If you need to recall these definitions use the *utility* described below.

Other disks required

Floppy disk users should ensure they have the disk storing the database to be described ready. This is referred to as the 'database disk' below.

Entering the Utilities

Access the utilities by the means appropriate to your storage system (Section 13), then *cursor select* Display database description from «SupaStore utilities».

Floppy disk users should 'Put the database disk in drive 0' as instructed.

You are then asked, 'Do you want to print the database description?'. This refers to output on a printer. Enter **Y** only if you have a printer connected and want a description on paper. Enter **N** if you want screen output only.

Database description output

For an example database description see the Botanical Appendix. Descriptions occupy three screen pages covering ;

- i) The database name, its creation date, the current number of *records*, the maximum number of records possible, the size of the existing database as a percentage of the maximum possible, and the screen colour and printer *interface* settings.
- ii) Each *field's* name, length, input type, screen page number, and screen position. The latter is given as two numbers representing the column and line position on the screen respectively. The page numbers will all be the same unless the record extends over more than one screen page.
- iii) A list of the currently unused records wiped in the *editor* (Section 12). Note, this section is omitted if there are no unused records.

Each page is held on the screen until you press the **SPACE BAR** as instructed.

SECTION 17 : Record selection

In some situations, you may want to examine a miscellaneous group of records. For example, you might want detailed Profiles (Section 5) of a sub-set of records from a long list retrieved by a *query*.

If this sub-set is small, and the record numbers are known, it would be tedious to have to *search* a large *database* for them. Equally, you may have to define quite a complex query (Section 6) to recover just those records.

The *utility* described in this section allows you to construct a list of record numbers. These can then be used as if they were a query to produce a *report* in any existing *format*.

Other disks required

Floppy disk users will need to have a database disk ready because the record list you define is saved on this.

Accessing the Utilities

Refer to Section 13 for the means of access appropriate to your storage system. Select Edit selected records.

FLOPPY DISK USERS — You will be asked to 'Put the database disk in drive 0'. Insert the British Trees disk, and press the **SPACE BAR** as requested.

HARD DISK AND ECONET USERS — Enter **Trees** when prompted for a database name.

Defining a Selected Records list

A «Select records» screen will appear. You must enter a Heading within the [] symbols. This is equivalent to a query name (Section 6) and will appear in the *headers* of reports.

The «Select records» screen has a working area between two horizontal lines containing a *window* in the screen text colour (Figure 17.1).

Database name: British Trees

ESCAPE: utility menu

Select records

Heading: []

17.1 Select Records Screen

Enter **14**. As you type this in, the number appears in the left of the window; after you press **RETURN** the number moves to the right to signify it has been stored as part of the list.

If you type a second record number in, (try **6**), the first number moves up out of the window to make room for it.

It is then a simple matter to add further record numbers in this way. To delete a number entered by mistake, move the list with the **↑** and **↓** keys until it is in the window and then press the **F7** function key.

When the list is complete, a combined press of the **SHIFT** and the **COPY** keys will save it. Floppy disk users must replace the Utilities Disk and press the **SPACE BAR** as instructed to return to «SupaStore utilities».

Using Selected Record lists as Queries

If you run SupaStore with the example database and go to «Query Selection» you will find the Select individual records option now has a number in the parentheses after it. This corresponds to the number of records in the list you've just defined. You can simply select this option and report it in any format from the «Format selection» list in the usual manner. If the format contains a *sort code*, the list will be sorted accordingly.

Changing existing Record lists

Only one selected records list can be stored in the system. If you use the Edit selected records utility again, the 'Heading prompt' will contain the current name. You can retain this by pressing **RETURN** or make changes by using the **DELETE** key and typing new characters.

In «Select records», the existing list can be moved through the window as above so that new records can be added or deleted with **F7** at any point. Save any changes with **SHIFT COPY**.

Defining new lists

To define a new record list, enter a new 'Heading', and then press **F8** at the «Select records» screen. To avoid accidental erasure, you will be asked 'Are you sure? Y/N'. Only answer Y if you want the list deleted. Save the new list you enter with **SHIFT COPY**.

SECTION 18 : Archiving and retrieving queries and formats

Only ten *queries* and *report formats* can be stored at one time.

Two *utility* options overcome this restriction by allowing you to transfer current queries and formats to *libraries*, making room for you to define new ones in the usual manner.

Query and Format libraries

The libraries can store large numbers of queries and formats in a single library file.

The terms 'archive' and 'retrieve' are used respectively to describe the transfer of a query/format to and from the library.

Queries versus Formats

The creation of query and format libraries, and archive/retrieval operations, are identical. Discussion below consistently refers to queries, but applies equally to formats.

However, there is one major difference in retrieval. When returned to the current list queries appear on the «Query selection» menu with ! *new query flags* (Section 5). This is because queries can't be automatically updated while they are in the library (Sections 11 and 12). A retrieved query thus has to be *re-searched* to take account of any changes to records since it was archived. In contrast, formats are unaffected when they return to the current list.

Creating a Query library

Floppy disk users should format a disk to hold the library. You will also need the Utilities and database disks. Follow the disk exchange prompts carefully.

Access the utilities (Section 13) and *Cursor select* the Archive or retrieve queries option from the «Supastore utilities» menu.

If this is the first time you have used this utility no library file will exist. The program recognises this and will respond with 'There are no queries archived'. Press **SPACE BAR** as instructed and you will be asked 'Do you want to create a new query library?'. Enter Y and the library file will be opened.

Archiving Queries

Immediately after creating a library an «Archive/retrieve queries» screen will appear (Figure 18.1). The left side of this screen contains the 'Current queries' in the database and two other options. The right side provides space for a list of 'Archived queries'; as the library is new this is empty but for a Delete an archived query option. You can move about these lists with the cursor arrow keys.

Database name: British Trees	
Archive/retrieve queries	
Current queries	Archived queries
Deciduous trees Oak trees Trees in the Family Salicaceae Trees taller than 35 metres Trees found in Hedgerows Native trees DELETE a current query SAVE and exit from archive	DELETE an archived query

18.1 Archive/retrieve Queries Screen

To archive any query, cursor select it from the 'Current queries' list; it will move from that list to the 'Archived queries' list.

When you have finished archiving, select the Save and edit option on the left. The archived queries will now be saved in the library file. Floppy disk users should put the database disk back in drive 0 as requested, then press **SPACE BAR**. The new current list will be saved with the database.

You can now re-enter SupaStore and define any new queries required (Sections 6 and 13).

Retrieving Queries from the library

Retrieval from an existing library involves taking the same steps as above to reach the «Archive/retrieve queries» screen. However, you are not required to confirm the creation of a new library.

If necessary, you can now archive any current queries to make room for those being retrieved.

Any unwanted queries in the 'Current' list can be removed by selecting the Delete a current query option, again making room for retrieved queries.

If you cursor select any query from the righthand list it is added to the 'Current queries', but it is not removed from the 'Archived queries' list. This maintains the library, and allows you to delete the same query from within SupaStore later without having to archive it.

When you have retrieved all the queries required, use Save and exit as above to return to «SupaStore utilities», exchanging disks as instructed.

Deleting Queries from the library

Whenever you are in the «Archive/Retrieve queries» screen any query can be removed from the library by cursor selecting the Delete an archived query option and then the query itself.

Separate libraries

There is no reason to archive queries/formats in a single large library. You can create as many libraries as necessary storing queries and formats for different purposes.

SECTION 19 : Labelling facilities

This section explains an applications *utility* which allows the user to create and print labels based on the contents of SupaStore *database fields*. Label definitions are saved on the storage medium you are using and can be recalled at any time.

Labels can be printed using fields from every *record* in a database, or just for those records matching a pre-defined and *searched query*.

Label databases and Fields

Users should bear in mind the physical constraints of the label size that their printer handles, and the limitations of this utility (see below) when constructing label fields.

To generate labels the user must clearly have a database containing appropriate fields. However, the same database can usefully cover a number of other fields that can be used in queries to define sub-sets of labels for printing.

An obvious label application is addresses. However, addresses can vary in form considerably, so organise database fields flexibly to cover all events. An example covering most eventualities follows.

```
Name      [                . ]
Add 1     [                ]
Add 2     [                ]
Add 3     [                ]
County    [                ]
Post Code [                ]
```

Name, county, post code always occur so they are named separately. The more general fields Add 1, 2 and 3 allow for variation in detailed location.

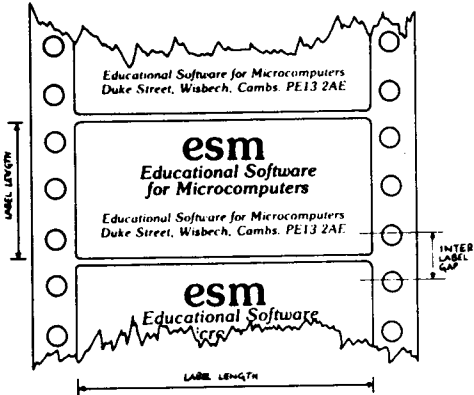
Creating New Labels

Floppy disk users should have prepared a blank formatted 'Labels disk' in advance, and have their database disk ready.

Obtain the «SupaStore utilities» menu by the means appropriate to your storage system (Section 13). On selecting the Label definer and printer *option* a «SupaStore labels» menu will be presented. Select Create a new label from this.

Floppy disk users should insert the database disk in drive 0 and press **SPACE BAR** as instructed.

After SupaStore information has been loaded, you are required to enter a label name and inter-label gap in turn. The latter is the number of lines between the labels to be fed through your printer (Figure 19.1). Label names may be up to 32 characters.



19.1 Definition of Label Terms

Defining Label Size

The «Label definer» screen has an upper area used to define the label size and a lower *cursor selection* menu of the fields in the database being used (Figure 19.2).

Label Definer

Label name: Tree names

Width: 32

Length: 5

Gap: 3

Pop. Name.....

Bot. Name.....

Pop. Name	Other Name	Bot. Name	Family
Decid/Ever	Native	Mature Ht	Leaf Descr
Fruit	Location	Use	Keywords
RECORD NO	FINISH		

19.2 Label Definer Screen

Initially, you should adjust the label size to match the physical size of your labels. The width of the label can be increased up to a maximum of forty characters by pressing the **→** key. Similarly, the **↓** key increases the label length up to a maximum of 10 lines. The **←** and **↑** keys reduce these dimensions if necessary. Note, statements of label width and length are maintained to the left of the upper screen area.

When the label is correct press **RETURN**. A cursor will appear in the field area. You now have to place the fields you want printing within the label.

Select the first field required. To place the field, move the small flashing cursor within the label using the cursor arrow keys to where you want the field to start on the label and press **RETURN**. The field will appear initially as ten characters long; you can increase or decrease its length with the **←** and **→** keys. To finally fix that field in position, press **RETURN** again.

You can now add remaining fields in the same way. In some situations you may want to place two fields on the same line, for example surname and forename if you have stored them separately. To do this follow the steps above, but finally fix the second field's position by pressing the red **F0** function key instead of **RETURN**.

Saving A Defined label

End label definition by selecting Finish. Floppy disk users will be asked to insert their Labels disk in drive 0.

If this is the first label you have created you will be advised that there is no label file and asked if you want to create one. Enter **Y** to save the label and return to the «SupaStore labels» menu (floppy disk users will need to re-insert their Utilities disk to do so).

Printing labels

Select the Print a label option from the «SupaStore labels» menu. Floppy disk users will be instructed to insert their Labels disk in drive 0. Then cursor select the label to be printed from the «Labels menu» that follows. Note, each label option contains a reminder of the database used to define it.

Floppy disk users must insert the disk storing the database used to define the label in drive 0 so that query and label data can be read.

Next, you have to choose a query from the database you are working with. A label will be printed for each of the records matching that query. To print labels for every record, select List all records.

The system then warns you to ensure that the label paper and printer are set up correctly, and asks you to specify how many spaces should be printed before each label (see Figure 19.1), and to enter **Y** or **N** to indicate whether printing should pause after each label.

After printing is finished, you will be returned to «SupaStore utilities». Again floppy disk users must return their Utilities disk to drive 0 first.

SECTION 20 : Creating sub-databases

Users with very large *databases* may only want to work with sub-sets of their *records* at one time to avoid lengthy *searches*. If such users define a *query* which retrieves the relevant sub-set of records, they can use the *utility* explained in this section to create a new smaller *sub-database* consisting of just those records.

This utility also permits users to include fewer *fields*, or adding new ones, to records in the sub-database. As the List all records query can be used to define the records making up a sub-database, complete databases can be copied with unanticipated fields added.

If you use List all records as the defining query, you will be asked whether you want any empty records (Section 12) copied or not. If you respond with **N**, empty records will not be copied to the new sub-database, but you should bear in mind transferred records will be renumbered.

Restrictions on use.

This option is only available to network users, or those with hard disks or dual *floppy disks*. This is because two floppy disk drives are required to handle the transfer of records from the main database to the sub-database on a new disk. The nature of operation also means this facility is only possible with single or double-sided databases stored on one floppy disk.

Creating a Sub-database

You must have defined a query and searched the database to retrieve the records to go into a sub-database. Floppy disk users should have a blank disk formatted ready to receive the new sub-database.

Obtain the «SupaStore utilities» menu (Section 13). *Cursor-select* the Create a sub-database option. Floppy disk users will be asked to put the existing database disk in drive 0.

Select from the list of the currently defined queries that follows the query that specifies the records to be copied to the sub-database.

FLOPPY DISK USERS ONLY — You are required to return the Utilities disk to drive 0, and put the existing database disk in drive 1. After SupaStore information is loaded, you must then put the blank sub-database disk in drive 0. It is important you ensure the correct disks are put in the drives specified and follow other disk exchange messages carefully.

A menu of the existing fields appears. Select in turn the ones to go into the sub-database ending with Finish. If you select New field you are prompted to supply its name, length and input type (see Sections 14 and 15). Note, the fields selected are ★ 'd and you cannot select a field twice.

As with new databases (Section 15) you must enter a creation date (in dd/mm/yy format) and database name. A screen similar to that used for defining the fields in a new database follows. However, as the field definitions are already known, all you have to do is locate them within this screen. Details of each field appears at the top if the screen in turn. All you need do is move the cursor to where the field should go press **RETURN**. Note, this also gives you an opportunity to re-design the field layout for the sub-database.

After the last field has been placed the data will be copied from the existing database to the new one.

A message 'Creating the new sub-database, Please wait' is followed by a screen which monitors the transfer of records. On completion the user is returned to «SupaStore utilities» (note, floppy disk users must insert the Utilities disk in drive 0 first).

SECTION 21 : Printer output

Because not all SupaStore users will have a printer, all examples in the sections above recommend screen output. However, it is an easy matter to produce *reports* on paper. Note, this section deals with printed output for standalone users, and network users with their own local printer. Section 23 describes how output can be sent to a network printer.

Printer selection

Ensure your printer is set up and switched on. Proceed with report production via «Query selection» and «Format selection» in the usual manner (Section 5), but cursor select Printer from the «Output selection» menu.

You will be asked how many copies you want, followed by the default value of 1. If you require only one copy, press **RETURN**. If you require more than one copy, delete the 1 and input the number required.

Note, printer output is paged and divided into *headers* and data in the same way as screen output. The number of lines in each printed page (including the header) will be equal to the value specified in your database configuration. Page length can easily be reset using the Modify configuration utility (Section 15).

Report widths

The potential width of printed reports will depend on make of printer, paper width and the size of characters being printed.

Wider reports are possible if you use a smaller character set supported by your printer. You can achieve this by using a 15 *pitch* wheel with a daisy-wheel printer, or by switching to *condensed print* mode if you have a dot-matrix printer.

For condensed print you will need to send special codes to the printer before your SupaStore session. These codes will be explained in your printer manual.

The maximum report width possible is 255 characters; this can be accommodated on 14" (35.5cm) wide paper in condensed print. In wide outputs you will find it useful to print items like the record number down both ends of the report.

Wide reports

If your report is going to be wider than that possible on your screen or printer you can do three things;

- a) You can save the format as it is. No error will occur; but each line will *wrap* making for an untidy layout.
- b) If you want to fit each record on a single line you must reduce the width of fields that will suffer least by not being printed in full (Section 7). Again, no error will occur; field contents will simply be truncated on the right-hand side.
- c) You could define two separate formats covering the fields require.

SECTION 22 : Spool file reports

The *spool file* option allows *reports* to be saved on disk.

Restrictions on use

Spool file output can only be used if you have dual *floppy disk* drives, a hard disk, or if you are an *Econet* user.

Because *records* are accessed during report production, the database disk must occupy a drive all the time. Spool files can thus only be saved on a free blank disk occupying the second drive in a dual system.

It follows that users working with large *databases* stored on two disks cannot use the spool file option either.

IMPORTANT: Under no circumstances should floppy disk users attempt to save a spool file on drives 0 or 2.

Saving spool file Reports

Choose a *query* and *format* as usual (Section 5) but select Spool file from the «Output selection» menu. A prompt for a Spool filename will appear in the lower half of this screen.

FLOPPY DISK USERS — Ensure a blank formatted disk is in drive 1. To save a report on drive 1 enter :1.filename. 'filename' may be any 7 character name valid under DFS. Users with double-sided drives can also save reports on drive 3 by entering :3.filename.

HARD DISK AND ECONET USERS — If you enter just a valid filename under ADFS, NFS or DNFS the spool file will saved in your own SUPASTORE directory. However, you can send the output to any other directory you have access to by entering Dir.filename substituting 'Dir' for the directory name.

A message 'Spooling, please wait' is displayed while the report is being saved. Spool file reports differ from screen and printer output by consisting of one long page with a single *header*.

Uses of Spool files

You may want to include SupaStore reports in other documents, essays, or articles. To avoid retyping them, reports saved on disk can loaded into a wordprocessor along with other explanatory text.

Spool file form is necessary if you want to distribute reports to other parties by electronic mail.

If you need multiple copies of a lengthy report, and have another computer system available, it would be convenient to print these using a spooled report on the second system, leaving SupaStore free for further searches.

The ability to load spooled reports into wordprocessors means that you can customize reports as necessary. For example it would be effective to print query, format and field names in bold type, underline important records and so on.

SECTION 23 : Network considerations

Section 1 noted that SupaStore can be run by users with different storage systems, namely *floppy disk*, *hard disk* and *Econet* file-server. However, SupaStore also operates under two modes of use, *standalone* and *multi-user*.

Standalone Usage

Standalone users have their own floppy or hard disk drive connected to their microcomputer and have complete control over all their *databases*. The *databases* and *utility* programs are stored on disks within their personal system and these are the only ones they can access.

SupaStore in a network environment

In the multi-user situation, *databases* created by individual users are stored on the network file-server in SUPASTORE sub-directories of their own network directory.

For example, user SUSAN will sign onto the network as ★ I AM SUSAN, then reach her database directory by entering

★ DIR SUSAN.SUPASTORE

However, facilities needed in common by network users, such as the SupaStore utilities, are stored in a root directory called \$.SUPASTORE created as if there was a user called SUPASTORE. Special programs in \$.LIBRARY ensure these can be accessed directly from the individual users sub-directory. For example to call upon the SupaStore Utilities, the user simply enters ★ SSUTILS.

Other *databases* that are to be used by everyone on the network can also be stored in a root directory of any name eg \$.DATABASES.

Finally, any Econet user may search the *databases* belonging to any other user provided the database 'owner' has established access rights (see below).

Owners and Non-owners

Network users are said to 'own' the *databases* stored in their SUPASTORE sub-directory. They also have complete control over these and can use them in exactly the same way as stand-alone users.

Individuals accessing a network database available to everybody, or a database belonging to another user, do so as 'non-owners'. Non-owners are not allowed to change the information stored in the *databases* they are accessing, but can use all the other SupaStore facilities. Moreover, non-owners can still use the «Main menu» option Edit (Section 11) to view the record contents of somebody else's database.

Installation of SupaStore EPROM's

SupaStore must be installed in all the British Broadcasting Corporation microcomputer stations in the network that are to be used for database operations. EPROM's should be fitted according to the instructions on the sheet accompanying the pack either by the network manager, or by individual users if they have responsibility for their own station. In a multi-user environment, it will also be possible to ★ LOAD SupaStore if you have a RAM card or second processor fitted to your British Broadcasting Corporation microcomputer.

Installation of Utility and Example Database files (Network managers)

The network manager should be responsible for transferring software and databases supplied with the SupaStore pack into appropriate directories. The tasks, some of which require system privilege, fall into the following areas;

1) To install the SupaStore Utilities firstly create a \$.SUPASTORE directory. Run the program called TRANSFER on the Utilities Disk to copy all the utilities files to the \$.SUPASTORE directory by entering **CHAIN "TRANSFER"**. At the same time the programs SSUTILS and SEND should be transferred to \$.LIBRARY. (If necessary an 80 track Utilities Disk can be created using the CONVERT program explained in the separate sheet supplied with the pack).

Create sub-directories named SUPASTORE for each person who will be using the system by entering

★ CDIR dname.SUPASTORE

substituting 'dname' with the user's directory name in each case.

3) The British Trees example database files must be copied from the floppy disk supplied to each user's SUPASTORE directory. The TRANSFER program used to copy the Utilities can be used for this. Note, because of Econet directory rules the example database files have to be renamed so that they end with _Trees; TRANSFER deals with this automatically.

4) It will probably be the manager's responsibility to identify databases that should be accessed by all network users, and ensure that they are installed in another root directory (eg \$.DATABASES). Databases should be installed with WR/R access rights so that everyone uses them as 'non-owners'.

Because the network database directory should be kept as secure as possible, the network manager should also be the only person responsible for keeping network databases up to date. If it is necessary to allow other users direct 'owner' access to the database directory to edit files it should be ensured that only one person is responsible for updating any one database. Moreover, all users must be warned that it is inadvisable for more than one user to update a database at the same time.



Using the example British Trees Database on the network

Because the network manager will have installed a copy of the example database on each user's SUPASTORE directory, this will be treated as an 'owner' database and they can follow the Getting Started section (5) to practice fundamental operations in the same way as standalone users.

Creating your own SupaStore databases

Individual network users can create databases by following the instructions in section 15. All they have to do is ensure they are in their SUPASTORE sub-directory. When running their own databases, network users can do all of the things described in the earlier sections of this Guide.

When creating databases network users have to provide a 7 character database name as well as the standard 15 character name. A number of database files are stored in their SUPASTORE directory under the shorter name. These file names are automatically prefixed with different characters (eg, L, J-) so that the system can distinguish between them and their functions.

Modifying your SupaStore configuration

If you have been using the British Trees database, the *configuration* information (specifying the printer type you will be using, and the screen colours) will be set initially to the example database's *default* values. If you want to change the printer type, access the network printer, or change the screen colours, you can use the Modify configuration option in the «SupaStore utilities» menu to reset these (see Sections 13 and 15).

Permitting users to access your databases

Assume you have a database called 'Books' in your SUPASTORE directory that you want to allow other users to have access to. To give permission for this, ensure you are in your SUPASTORE directory and enter;

★ ACCESS ## Books WR/R

To permit access to any other database use its 7 character name instead of Books.

Accessing other users databases

You can only access databases owned by other users if access has been permitted by them.

To check which databases have been permitted for access by 'non-owners' in another user's SUPASTORE directory simply enter;

★ CAT \$.dname.SUPASTORE

substituting the user's directory name for dname. All databases that you can search will have WR/R access rights next to them.

To establish working access to another user's database you must use the utility Access another user's database, so enter **★ SSUTILS** to obtain «SupaStore utilities».

You will be prompted to enter the database owners directory, then the database name. Assume for example you want to access a database named 'Census' belonging to a user called MARTIN. You would enter

\$.MARTIN.SUPASTORE

for the owners directory and,

Census

as the database name.

If you have asked to use a database that is not permitted, you will get an 'Insufficient access' message.

If you don't have any databases on your own SUPASTORE directory you will be asked to provide printer and screen colour configuration information (Section 15) for SupaStore to use.

The Access another user's database utility copies the files containing query, format and sorting information from the owner's directory to your own SUPASTORE directory. The 'owner's' queries and formats are thus available to search and report the database, or you may define you own. You can now do virtually everything SupaStore allows with databases that you have access to, including the creation of personal *libraries* (Section 18) and label files (Section 19). The only thing you can't do is change the information stored by the owner.

'Non-owners' should always bear in mind that the queries they are using to search other people's databases are not automatically updated if the owner changes any records. However, in this situation the system is able to detect that changes have been made since you last used own query and gives you a warning or automatically re-searches as appropriate.



Accessing network databases

In certain situations, some databases need to be accessed by everyone on the network. If this is the case it may be most convenient to store them in their own root directory (eg \$.DATABASES).

It would then be a simple matter for any user to access these as a 'non-owner' from their own SUPASTORE sub-directory in the manner described above. For example, if network databases were stored in a root directory \$.DATABASES, the user would enter

\$.DATABASES

when prompted for the owners directory name in the Access another user's database utility.

Using a network printer

All printer output is automatically sent to the network printer if you select the Network option when creating a new database (Section 15). To change between local and network printers re-select the appropriate type by using the Modify configuration option in «SupaStore utilities» (see above and Section 15)

Transferring to other network facilities

On installation, the program SEND in \$.LIBRARY will effect a return to BASIC when a user leaves SupaStore or the utilities. The network manager is free to replace the SEND supplied with his own version to enact a transfer to any other network facilities as necessary.

Appendices

Appendix A: Botanical information	77
Appendix B: Error messages	83
Appendix C: Query reference information	85
Appendix D: Command reference information	87
Appendix E: Format reference information	88
Appendix F: Sorting reference information	90
Appendix G: Database reference information	92
Appendix H: Technical details	95
Appendix I: Screen design sheet	96
Glossary	97
Index	101

Appendix A: Botanical information

1) Database Description Utility output

SupaStore Database Description

Database name: British Trees Creation date: 04/09/85

Number of records: 44

Maximum record number: 339

Percentage usage: 13%

Menu foreground colour: cyan

Menu background colour: black

Editor foreground colour: blue

Editor background colour: white

Printer type: Parallel

Field Descriptions

Number	Field name	Length	Input type	Page	Screen position
1	Pop. Name	19	Any	1	20,4
2	Other Name	22	Any	1	20,5
3	Bot. Name	24	Any	1	6,7
4	Family	16	Letters	1	48,7
5	Decid/Ever	9	Letters	1	6,10
6	Native	1	Y/N	1	35,10
7	Mature Ht	2	Integers	1	56,10
8	Leaf Descr	25	Any	1	17,13
9	Fruit	7	Letters	1	28,15
10	Location	35	Any	1	16,18
11	Use	35	Any	1	16,20
12	Keywords	35	Any	1	6,22

2) Explanation of Fields

Pop Name : Popular name. Unique to each record, although some strings are common to a number of records eg Oak, Elm.

Bot Name : Botanical name. Some genera occur in a number of records (see glossary below).

Other Name : Alternative name in common use. Unique to record, but only applies to limited number of records.

Family : Botanical Family.

Decid/Ever : Keyword field, containing either Evergreen or Deciduous.

Native: Contains Y or N to indicate whether the tree type is native to Britain or not.

Mature Ht : The mature height in metres.

Leaf Descr : Contains keywords describing the trees leaves. In conifer records the field contains the keyword 'Needles' plus how many rows or bunches these are arranged in. In broadleaf records the field starts with either 'Simple' or 'Compound', If simple, a second keyword defines the leaf shape. If compound, two further keywords describe the arrangement of leaflets and their shape respectively (see glossary below).

Fruit : Keywords cover broad definition of the fruit used for propogation.

Location : Where the trees are most commonly seen.

Uses : What the tree is used for. Covers use of bark and fruit as well as wood, and also historical uses.

Keywords : An empty field for storing the users own keywords.



3) Glossary

Botanical name In Latin. The first is the GENUS. The second describes a characteristic feature, (examples in the glossary are PUBESCENS, NIGRA, CAMPESTRE, SYLVESTRIS).

Campestre Second part of BOTANICAL NAME meaning field.

Catkin Defined here as a former female flower spike retaining collections of seeds for later dispersal, usually by wind or water.

Compound Said of leaves composed of several LEAFLETS.

Cones FRUIT of CONIFEROUS trees.

Coniferous Cone-bearing.

Coppice A managed wood where trees are cut at ground level to induce denser slender growth.

Cordate Referring to leaf shape, meaning heart-shaped.

Deciduous Trees which lose their leaves annually.

Evergreen Trees which keep their leaves throughout the year.

Family Grouping of related genera with a common characteristic.

Fleshy Referring to a type of FRUIT. Includes berries and fruits like apple, pear and plum.

Fruit The ripe seeds and structure surrounding them. See also CATKIN, CONES, FLESHY, NUTS and WINGS.

Genus Smallest natural group of related but distinct species. Plural Genera.

Lanceolate Referring to leaf shape, meaning spear-like.

Leaflet Part of a COMPOUND leaf.

Lobed Said of leaves which are divided, but not into separate LEAFLETS.

Native Not introduced into Britain in the historical past.

Needles Slender needle-like leaves on CONIFEROUS trees. Arranged in rows or bunches.

Nigra Second part of BOTANICAL NAME meaning black.

Nuts Hard dry FRUIT.

Obovate Referring to leaf shape. Egg-shaped as in OVATE, but with the widest part most distant from the stem.

Ovate Referring to leaf shape, meaning egg-shaped.

Palmate Referring to the arrangement of LEAFLETS in a COMPOUND leaf which arise from one point in a hand shape.

Palmately-lobed Referring to the shape of SIMPLE leaves which have lobes arranged like a hand.

Pinnate Referring to a COMPOUND leaf with LEAFLETS arranged in two rows either side of a single stem.

Pinnately-lobed Referring to the shape of SIMPLE leaves which have pairs of lobes down the leaf.

Pubescens Second part of BOTANICAL NAME meaning downy.

Simple Where there is a single leaf on a stem.

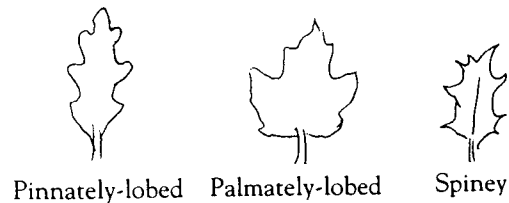
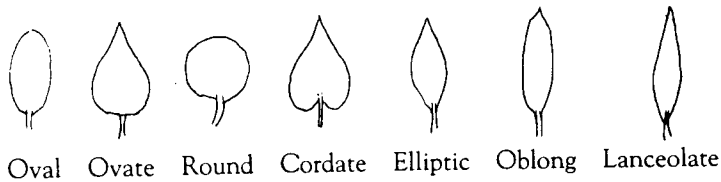
Scales Degenerate leaf.

Sylvestris Second part of BOTANICAL NAME meaning wild.

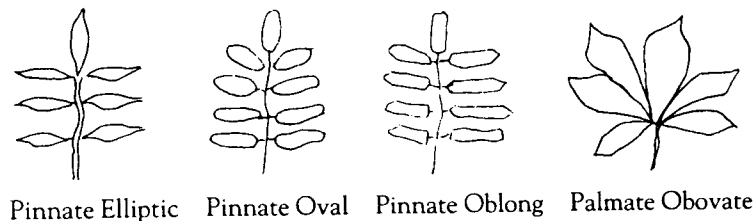
Wings FRUIT with wing-like feature attached to facilitate dispersal by wind.

4) Leaf Silhouettes

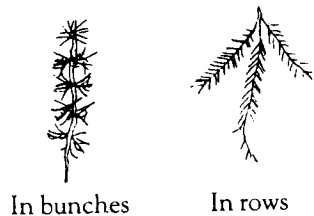
Simple Leaves



Compound Leaves



Needles



Scales



5) Botanical References & Field Guides

Mitchell, A. *A Field Guide to the Trees of Britain and Northern Europe*, Collins London, 415pp ISBN 0 00 212035 6

Phillips, R. *Trees in Britain, Europe and North America* Pan, London, 224pp ISBN 0 330 25480 4

Wilkinson, G. *A History of British Trees* Hutchinson, London, 176pp ISBN 0 09 146000 x

6) Other resources

Child Education Special (1985) *Trees* No 42, Scholastic Publications, Leamington Spa.

Lucht, I. *Tree Calendar Blacks*, ISBN 0-7136-1908-2

Selberg, I. *The Nature Trail Book of Trees*, Usborne, ISBN 0-86020-099-X

7) Suggested enquires

- What do trees growing in hedgerows have in common?
- What trees can be seen in both woods and hedgerows?
- Do all evergreen trees display needles?
- How many trees have compound compared to simple leaves?
- Have all evergreen species been introduced into Britain?
- Do any deciduous trees bear cones?
- Which deciduous trees are not native to Britain?
- What are the leaf characteristics of native compared to alien species?
- How many deciduous trees between 30 and 40 metres are there in the database?
- What are the tallest and shortest species in the database? Is there any relationship between mature height and family, location or use?
- What makes ornamental species attractive? Are any particular families common in parks and gardens?
- Which trees don't have an alternative name?
- What are the characteristics of trees put to unusual uses?
- Which trees yield useful products apart from timber?

8) Data for Aspen and White Poplar

Record no. 44 The Aspen.

Field	Data
Pop. Name	Aspen
Other Name	NA
Bot. Name	Populus Tremula
Family	Salicaceae
Decid/Ever	Deciduous
Native	Y
Mature Ht	15
Leaf Descr	Simple Round
Fruit	Catkins
Location	Thickets
Use	NA
Keywords	NA

Record no. 45 The White Poplar.

Field	Data
Pop. Name	White Poplar
Other Name	Abele
Bot. Name	Populus alba
Family	Salicaceae
Decid/Ever	Deciduous
Native	Y
Mature Ht	25
Leaf Descr	Simple Palmately-lobed
Fruit	Catkins
Location	Parks Roadsides
Use	Ornament
Keywords	NA

Appendix B: Error messages

This appendix anticipates possible mistakes arising from use of the system. Problems and their solutions are listed by the section where they are likely to occur. Any error messages appear in *italics*.

Section 5 : Getting started

i) *Not Configured* — refer to manual.

If you are using floppy disks, the disk in drive 0 is not a SupaStore database disk. Insert the correct disk and press **RETURN**.

If you are using a hard disk or Econet, there is no database matching the name supplied. Ensure you are in the correct directory and check the database names available.

ii) *The date must be in dd/mm/yy format.* The date entered must be in a precise day/month/year format (eg 01/02/85). Single digit days and months must be prefixed by a zero; only / can be used as a separator. Use the **DELETE** key to remove your entry and re-enter one in the correct format.

iii) The computer's bell sounds when entering a date. You've tried to enter an illegal character. Only numbers and / are allowed at date entry.

Section 6 : Defining Queries

i) The query you thought you'd defined is not on the «Query selection» menu. Most likely you used **ESCAPE** during query definition. You must define it again. Always use the Return to main menu option.

ii) *All queries are used. Delete one and try again.* The maximum of ten queries are already defined. You must delete (Section 10) or archive (Section 18) one or more before you can define any new ones.

Section 7 : Defining Formats

i) The format you've just defined is not on the «Format selection» menu. Most likely you used **ESCAPE** during format definition. You must define it again. Always use the Return to main menu option.

ii) *All formats are used. Delete one and try again.* The maximum of ten formats are already defined. You must delete (Section 10) or archive (Section 18) one or more before you can define any new ones.

Sections 11 and 12 — The Editor

i) The computer's bell sounds when you attempt to enter something into a seemingly empty field for two reasons.

Firstly, the input code for that field is preventing entry of certain characters. The input codes for all fields can be checked by using the database description utility (Section 16). Section 14 and Appendix G list what characters are allowed under the different input codes.

Alternatively, the field may contain spaces. Test this by moving the flashing editor-cursor to the end of the field and use the **DELETE** key several times. After this you should be able to enter some characters.

ii) The changes you thought had been made to a record haven't been saved. Most likely, you left that record by pressing **ESCAPE**. Always leave a record you've made changes in by pressing **CTRL** and **F**, **CTRL** and **Y**, or move to another using function key **F10**.

Sections 15-20 : Utilities

If you are using floppy disks and the computer's bell sounds when you press the **SPACE BAR** as requested in all these sections. You haven't exchanged the disk as required, or you have inserted the wrong disk. Simply ensure the correct disk is in drive 0 and proceed.

Section 21 : Printer output

Nothing happens after requesting printer output. If your printer is not connected, or you've selected printer output and don't have a printer, press **BREAK** and start again from the «Main menu». If the printer's power is off, or its switched to off-line, switch on or change to on-line.

Appendix C: Query reference information

Note, all examples can be defined and used to search the British Trees database.

Simple Queries

Simple queries are composed of a FIELD COMMAND STRING 'sequence'.

Example : Pop Name Contains Ash

This retrieves the Ash and the Mountain Ash. If only the common Ash was required, Starts with should have been used instead of Contains. Moreover, a search for *Fraxinus excelsior* in the botanical name field would be even more precise.

Operators and complex Queries

Complex queries are composed of FIELD COMMAND STRING sequences joined by AND or OR operators.

Examples:

- 1) Use Contains Ornament or Decid/Ever Contains Evergreen
- 2) Use Contains Ornament and Decid/Ever Contains Evergreen
- 3) Mature Ht Less than 41 and Mature Ht Greater than 29
- 4) Mature Ht Greater than 40 or Mature Ht Less than 30

Example 1 retrieves all ornamental trees and all evergreen trees; some deciduous ornamentals may be included. Use of AND instead of OR in example 2 means only evergreen ornamentals would be recovered.

Examples 3 and 4 illustrate how a combination of Less than and Greater than extract records matching a range of values. Example 3 finds trees with mature heights between 30 and 40 metres, while 4 uses or to retrieve all trees outside this range.

Use of Brackets

FIELD COMMAND STRING sequences in long queries are evaluated first if they are enclosed in brackets.

Examples;

- 1) Location Contains Hedgerows and Ever/Decid contains Deciduous or Mature Ht greater than 30.
- 2) Location Contains Hedgerows and (Ever/Decid Contains Deciduous or Mature Ht greater than 30)

Example 1 without brackets finds all deciduous trees found in hedgerows and, all trees over 30 metres irrespective of whether they are deciduous or in hedgerows.

Example 2 with brackets makes the search more restrictive. It first confirms the tree is deciduous or over 30 metres, but only adds it to the retrieved list if it is also found in hedgerows.

Related Sections; 5, 6 and Appendix D.

Appendix D: Command reference information

Explanation of Query Commands

In FIELD COMMAND STRING sequences, any of the following can be selected for COMMAND.

Contains : The string may occur anywhere within the field as whole or part words. To find whole words in the middle of fields include a space before and after the string. Note, short strings used with 'Contains' may occur elsewhere as part of words and retrieve unexpected records.

Starts with : The field must begin with the string specified. The string can be whole or part words.

Doesn't contain : The string must not occur anywhere within the field.

Equal to : The entire field contents must match the string.

Not equal to : The entire field contents must be different from the string.

Less than : The contents of numerical fields must be less than the string value. When used with alphanumeric fields comparisons are based on ASCII values (see Appendix F).

Greater than : The contents of numerical fields must be greater than the string value. Works with alphanumeric fields like 'Less than' but with reversed priorities.

Examples

These examples can be tried with the British Trees database. The commands achieve varied success in retrieving trees with mature heights of 30 metres.

- | | |
|-----------------------------|---------------------------------|
| 1) Mature Ht Contains 30 | 2) Mature Ht Starts with 3 |
| 3) Mature Ht Starts with 30 | 4) Mature Ht Doesn't contain 40 |
| 5) Mature Ht Equal to 30 | 6) Mature Ht Not equal to 40 |
| 7) Mature Ht Less than 60 | 8) Mature Ht Greater than 29 |

1, 3 and 5 are the only examples to retrieve just trees of 30m. Example 2 will find all trees between 30m and 39m. 4 and 6 will recover all 30m trees, but every tree which is not 40m as well. 7 retrieves every tree below 60m, and 8 recovers all trees 30m and above.

Related Sections; 5, 6, Appendix C and Appendix F.

Appendix E: Format reference information

Report organisation

A short example report follows.

Database: British Trees

Query date: 04/09/85

Query: Oak Trees

Format name: Popular and Botanical Names

Page: 1

Total: 3

REC	Pop. Name	Bot. Name	Family
2	Pedunculate Oak	Quercus robur	Fagaceae
3	Holm Oak	Quercus ilex	Fagaceae
4	Sessile Oak	Quercus petraea	Fagaceae

The pages in a SupaStore report are organised into headers and data. The header provides summary information covering, the database name, the query being reported, the search date, the number of records answering the query and the report format name.

The format defines the list of items to be displayed in columns across a report. The items may be field contents, record numbers or the date when the record was last edited. In the example above, record number and the contents of three fields are reported.

Adjusting the width of formats

Initially, formats only provide ten spaces for the contents of each field. Because the three fields in the example were wider than ten characters the format width was adjusted to allow room for long names. If this adjustment hadn't been made the report would have looked like this;

Database: British Trees

Query date: 04/09/85

Query: Oak Trees

Format name: Popular and Botanical Names

Page: 1

Total: 3

RECORD NO	Pop. Name	Bot. Name	Family
2	Pedunculat	Quercus ro	Fagaceae
3	Holm Oak	Quercus il	Fagaceae
4	Sessile Oa	Quercus pe	Fagaceae

Note, only the first ten characters of the 'Pop. Name' and 'Bot. Name' fields have been printed.

Format field widths are adjusted by selecting SET TABS after the field list and pressing;

SHIFT to increase the space beyond ten characters, or,
SHIFT to reduce the space.

Justification and Sorting symbols

Before saving a report format, you can use simple function key presses to specify that field contents are to be printed right-justified or used for sorting. On doing so different symbols appear under the fields in the format list. These symbols are reproduced below along with their meaning and the function keys used for 'encoding'.

Symbols Meaning

- Field is encoded for right-justification
- ^1 First field used for an ascending sort
- ^2 Second field used for an ascending sort
- ^3 Third field used for an ascending sort
- ^4 Fourth field used for an ascending sort
- ^5 Fifth field used for an ascending sort
- v1 First field used for an descending sort
- v2 Second field used for an descending sort
- v3 Third field used for an descending sort
- v4 Fourth field used for an descending sort
- v5 Fifth field used for an descending sort

Function Key

	then
	then
	then
	then
	then

Compare the two example reports above. In the first, RECORD NO had a justification code causing the numbers stored in the field to be printed right-justified.

Appendix F explains how field contents are used for sorting.

Related Sections; 7, 8 and Appendix F.

Appendix F: Sorting reference information

This appendix explains the rules by which field contents are used for sorting. However, the same principals apply when the contents of alphanumeric fields are compared with query strings during searching (see Appendix D).

Sorting criteria

When field contents are used for sorting records each character is converted into its decimal ASCII value and records are ordered on this basis. Each character in the sort field is used in turn so that records with the same first character in that field are grouped together but ordered according to the second character and so on.

ASCII values are numerical. Thus, in ascending sorts records with low values in their sort field come first in the sorted list. The reverse applies to descending sorts.

Actual ASCII values for the alphanumeric character set are listed in the table below. Note these automatically establish priorities between numbers and characters of the alphabet, and between lower and upper case letters. Furthermore, because 'space' has the lowest ASCII value, leading spaces in fields can affect ordering significantly (see example below)

ASCII Values

Space	32	A	65	N	78	a	97	n	110
+	43	B	66	O	79	b	98	o	111
-	45	C	67	P	80	c	99	p	112
0	48	D	68	Q	81	d	100	q	113
1	49	E	69	R	82	e	101	r	114
2	50	F	70	S	83	f	102	s	115
3	51	G	71	T	84	g	103	t	116
4	52	H	72	U	85	h	104	u	117
5	53	I	73	V	86	i	105	v	118
6	54	J	74	W	87	j	106	w	119
7	55	K	75	X	88	k	107	x	120
8	56	L	76	Y	89	l	108	y	121
9	57	M	77	Z	90	m	109	z	122

Example

The seven records with their sort field contents on the left would be sorted by SupaStore into the order on the right.

Rec 1 [willow]	Rec 5 [Elm]
Rec 2 [Apple]	Rec 6 [Beech]
Rec 3 [Walnut]	Rec 4 [Oak]
Rec 4 [Oak]	Rec 2 [Apple]
Rec 5 [Elm]	Rec 3 [Walnut]
Rec 6 [Beech]	Rec 7 [Willow]
Rec 7 [Willow]	Rec 1 [willow]

Note;

- 1) The leading spaces in the fields of records 5 and 6 produce an unexpected ordering of the tree names. Moreover, the two spaces before 'Elm' rank it ahead of the 'Beech'.
- 2) A typing mistake using zero instead of 'O' in record 4 means the first character is numerical and puts this before the remaining records.
- 3) The last four records are strictly in alphabetical order with 'Apple' first and the names beginning with 'W' grouped together and ordered by the second field character. But note, record 7 has been put before record 1 because it begins with an upper case 'W'.

Related Sections; 6, 8, and Appendix D.

Appendix G: Database reference information

Database specifications

Maximum Record Size : 32k British Broadcasting Corporation Model B
= 3000 chs
64k with second Processor = 10000 chs

Maximum Number of Records per database: Specified by hard disk and Econet users. Varies according to configuration for floppy disk users, see tables A and B below.

Maximum Number of Fields per Record : 250

Maximum Field Size : 255 chs

Field Input Codes

- A — Any keyboard characters.
- O — All characters except digits.
- L — Letters and space.
- P — Telephone Numbers.
- Y — Y or N for Yes or No responses.
- I — Integer numbers.
- F — Floating point numbers.
- D — Dates in dd/mm/yy format.

Calculating Record Size

Add up the individual lengths of all the fields in your record. Record size in characters can then be calculated from;

Rec Size = (Total Length of all Fields) + (No Fields) + 9

TABLE A : Database Configurations and Disk Formatting

Databases can be created for storage on any of the following floppy disk drive configurations. The codes in parenthesis (eg SS40) are used to represent the configurations in table B and the graph below. The numbers under 'Sides' indicate the sides of the disks to be formatted prior to database creation.

Configuration	Code	Sides
Single drive, single-sided 40 track	(SS40)	0
Single drive, single-sided 80 track	(SS80)	0
Single drive, double-sided 80 track	(SD80)	0 2
Dual drive, single-sided 40 track	(DS40)	0 1
Dual drive, single-sided 80 track	(DS80)	0 1
Dual drive, double-sided 80 track	(DD80)	0 2 1 3

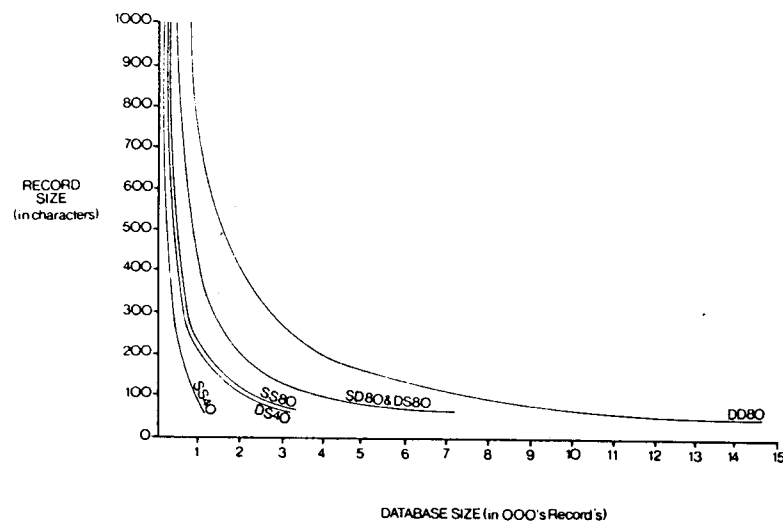
TABLE B Database Size by Record size and Configuration

Rec Size	Configuration					
	SS40	SS80	SD80	DS40	DS80	DD80
50	1164	3507	7192	3496	7192	14579
100	875	1843	3782	1839	3782	7659
150	593	1250	2564	1247	2564	5192
200	448	947	1940	944	1940	3929
250	361	762	1560	760	1560	3160
300	301	637	1303	635	1303	2641
350	258	547	1121	545	1121	2269
400	227	480	983	478	983	1989
450	202	426	875	425	875	1770
500	182	384	788	383	788	1595
600	151	321	657	319	657	1331
750	122	257	527	256	527	1067
1000	91	193	396	192	396	802

The table above lists the database sizes possible for a number of record size and disk drive combinations. Note, single drive, double-sided systems have identical capacities as dual drive, single-sided systems. If either is possible on your drives, a single drive, double-sided database is preferable because the second drive is left free for spooling and other options.

The same table is represented in graph form below. You can use this to;

- i) Read off the database size possible for selected disk drive characteristics and any given calculated record size.
- ii) Read off the maximum record size possible under different configurations with fixed database size in mind.
- iii) With record and database size fixed you can work out the optimum disk drive configuration.



Related Sections; 14, 15 and Appendix H.

Appendix H: Technical Details

SupaStore Specifications

Maximum Record Size : 32k British Broadcasting Corporation Model B
= 3000 chs
64k with second Processor = 10000 chs

Max no records per database: Varies according to configuration, (see Table B, Appendix G).

Maximum no of screen pages per record in editor = 7

Max no of fields per record = 250

Max field length = 255 chs

Field name length = 10 chs

Max no queries stored = 10

Max query name length = 31 chs

Max query string length = 20 chs

Max no formats stored = 10

Max format name length = 31 chs

Max no reports sorted at one time = 1

Max report width = 255 chs

Max no printed copies = 99

Multi user Specifications

Max no simultaneous users : Reading = Number of stations on Econet
Writing = 1

System Specifications

Required Computer Configuration: British Broadcasting Corporation microcomputer model B, B+ and Master 128, with OS 1.20 and above.
Filing Systems; DFS 0.9 and above, NFS 3.34, ADFS or DNFS.

Storage Systems :

- i) Standalone Versions — All combinations of single/dual, single-sided/double-sided 40/80 track floppy disk drives
- ii) Standalone Versions — Hard disk system up to 30 megabytes.
- iii) Multi-user Versions — User directories on Econet file-servers.

EPROM : 16k 27128

Appendix I: Screen design sheet

Database Name: Length:	Input Type:	Record Type:	Usable Records: Page 1
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			

Note: The space required on the screen for a field and its name involves three extra columns to accommodate a separating space and the symbols [].

Glossary

ASCII American Standard Code for Information Interchange. Each item of a computer's character set is given a numerical code (see Appendix F).

Alphanumeric Refers to FIELDS, STRINGS and keyboard entries which are a mixture of letters and numbers.

Autoboot To automatically run a computer program stored on FLOPPY DISK by pressing the **SHIFT** and **BREAK** keys together.

Baud Rate A measure of the speed at which data is transmitted through a serial INTERFACE. Has to be specified for serial printers when CONFIGURING DATABASES.

Command Language Used to define QUERIES. Based on FIELD COMMAND STRING sequences and OPERATORS.

Command The part of a QUERY which specifies how a STRING is to be compared with FIELD contents during SEARCHING. There are seven Supastore commands (see Appendix D).

Condensed Print Small type of print produced by sending special codes to dot-matrix printers (see PITCH).

Configuration Describes the DISK DRIVE characteristics and printer INTERFACE a DATABASE has been designed to work with.

Cursor Takes a variety of forms in Supastore. Mostly as a block in INVERSE COLOURS over OPTIONS in CURSOR SELECTION menus or as a small flashing block to mark the point where keyboard entries will be placed. See also FIELD CURSOR and EDITOR CURSOR.

Cursor Selection The method of choosing OPTIONS in Supastore. The user moves a CURSOR over an OPTION and selects it by pressing **RETURN**.

Database Generally an organised collection of information which can be SEARCHED by computer to answer the user's questions. Supastore databases consist of a finite number of RECORDS.

Disk See FLOPPY DISK

Disk Drive A mechanism for reading and writing information from and to FLOPPY DISKS.

Default The value or setting assumed in the absence of instructions.

Econet A NETWORK for the BBC microcomputer.

Editor A program used to change or edit RECORDS, and to add new records to DATABASES. Accessed by the «Main menu» OPTION Edit.

Editor Cursor Small flashing CURSOR in the EDITOR which marks the point where keyboard entries will be placed in a FIELD.

Encode To assign a SORT or JUSTIFICATION CODE to a FIELD when defining a REPORT FORMAT. Codes are acted upon when the format is used to report a QUERY.

Encoded Field A FIELD with a SORT or JUSTIFICATION CODE assigned to it.

EPROM The microprocessor device that stores the main processing options in SupaStore. A type of Programmed Read Only Memory installed in the microcomputer making its programs immediately available.

Field Part of the internal organisation of a RECORD. A space of fixed length to store information about a specific aspect of the record's subject. Also part of a QUERY to indicate which field is to be checked for a STRING during SEARCHING.

Field Cursor A block in INVERSE COLOURS that fills the FIELD currently being changed in the EDITOR.

Floppy Disk A storage medium for information and computer programs.

Format see REPORT FORMAT

Formatted Disk A FLOPPY DISK prepared to receive information using a formatter program provided with the DISK DRIVE or microcomputer.

Free Record Refers to the empty RECORD maintained after the last complete record in a DATABASE. Automatically used when you add a new record.

Hard Disk A fixed disk storage system with very high capacity for storing information. Also referred to as a Winchester disk.

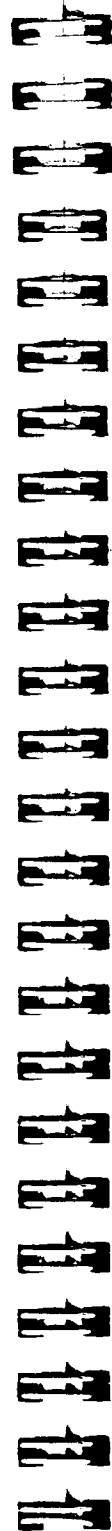
Header Part of a REPORT containing QUERY, FORMAT and DATABASE names, the date it was produced, and the number of records retrieved.

Inverse Colours A way of highlighting an area of the screen. Text in that area is displayed in the background colour, and is made visible by a surrounding block in the normal text colour. Used to form the CURSOR in CURSOR SELECTION menus, and also to indicate FIELD length when defining REPORT FORMATS.

Input Type Defines the characters allowed to be entered in a FIELD from the EDITOR. Has to be specified using an INPUT CODE for each field when CONFIGURING a new DATABASE.

Input Code One of eight single character codes defining a FIELD'S INPUT TYPE (See Section 14)

Interface The circuitry connecting equipment together. For example between a printer and microcomputer. May be parallel or serial. Has to be specified when CONFIGURING a new DATABASE.



Justification Code A code assigned to a FIELD during REPORT FORMAT definition. Acknowledged in the «Define a New Format» screen by a ★ symbol. When the format is used to report a QUERY the contents of that field are printed right-justified.

Keywords Type of FIELD contents. A simple word stored in that field over a number of RECORDS. Provides a consistent STRING to retrieve those records by.

Library A large collection of QUERIES and FORMATS stored on FLOPPY DISK separately from the current lists stored with the DATABASE. Queries and formats can be transferred between the library and the current lists using a UTILITY program.

Multi-user A version of SupaStore that operates in a multi-user or NETWORK environment.

Network System of linked computers sharing storage facilities.

New Query Flag An ! symbol placed to the right of QUERIES in the «Query selection» menu to signify they are newly-defined and the DATABASE hasn't been SEARCHED with respect to them yet.

Operators Part of the SupaStore COMMAND LANGUAGE. The words AND and OR are used to join simple FIELD COMMAND STRING sequences into complex QUERIES (See Query Appendix).

Options The choices available to the user in CURSOR SELECTION menus.

Pitch Meaning characters per inch. Wider reports can be achieved with daisy-wheel printers by fitting a 15-pitch wheel.

Query A question that is put to each RECORD in the DATABASE. Simple queries are represented by FIELD COMMAND STRING sequences. Complex queries are simple queries joined by OPERATORS. Operations involving queries are accessed from the «Main menu» OPTION Queries.

Record The basic unit of a SupaStore DATABASE. Information within a record is organised into a FIELDS. Each record has a number which signifies its position in the database. Record numbers start at zero.

Report SupaStore reports contain information from the records which answer a QUERY after the database has been SEARCHED. They are divided into pages each with a HEADER followed by the data itself. Reports are generated by selecting the «Main Menu» OPTION Reports.

Report Format Defines the FIELDS and other items which appear in a REPORT. Operations involving report formats are accessed by the «Main menu» OPTION Formats.

Search To put a QUERY to each of the RECORDS in a DATABASE in turn.

Search Result The list of RECORDS answering a QUERY.

Sort Code A code assigned to a FIELD during REPORT FORMAT definition. When the format is combined with a QUERY the contents of that field are used to sort the records in the REPORT.

Sort Flag An S symbol placed to the right of a QUERY and FORMAT in the «Query selection» and «Format selection» menus. Signifies these were the last used for a sorted REPORT.

Spool File A disk file used to save REPORTS in the form that they appear on the screen.

Standalone Used alone, not linked to other computers in a network. Used here to describe such a version of SupaStore.

String Generally a sequence of characters. In SupaStore also part of a QUERY, representing the characters checked for in a FIELD when the DATABASE is SEARCHED.

Sub-database A DATABASE created from an existing SupaStore database using a UTILITY program. The sub-database can consist of just the records matching a QUERY asked of the existing database, or it can be a copy of the original. The sub-database can have fewer FIELDS, the same fields, or more fields than the original.

Utilities A collection of programs that provide a range of facilities additional to the basic operation of the system. So called because they are generally useful.

Window A feature in the Selected Records UTILITY. A RECORD list for output is entered and changed through this.

Wrap Used in the context of screen and printer output. Lines of output longer than the screen or printer width wrap onto a second line.

Index

G denotes a Glossary Item

Section	Section
Applications 24	Date, Set Today's 5
ASCII G	Entry Format 5,15
Alphanumeric characters G	When to 5,11,12,15
Autoboot G,13	Econet G,1,23
Baud Rate, of serial printer ... G,15	Editor G,11,12
British Trees Database 4,5	Cursor G,11,12
Fields 4	Cursor Movement 11
Formats 4,5	Screen 11
Queries 4,5	Screen Design 14,15
Reports 5	Changing Screen Design 15
Selected Records 17	Editor Cursor G,11,12
Command Language G,6	Edits 11
Command G,6	Network User 11
Condensed Print G,21	Saving 11
Cursor G,3,5	Tracing 11
Cursor Selection G	Encoded Field G,8
Definition 3	EPROM G
Menus 3	Installation 5,23
Database G,14	Errors 5
Aims 14	ESCAPE key 3
British Trees 4	Escape Message 3
Capacity 14,15	Field G
Checklist 15	Alphabetical 8,14
Configuration G,14,15	Common Functions 14
Creation 14,15	Consistency 14
Creation Date 5,15	Contents 3,14
Default Configuration 15	Cursor 11,12
Design 14,15	Decisions 14
Modifying Configuration . 15,23	Definition 2,15
Names 15	Design 14
Preliminary Considerations .. 13	Encoded 8
Saving Configuration 15	Full Text 14
Saving Definition 15	Index 14
Size 14	Input Code G,14,15
Size Restrictions 14,15	Input Type G,14,15
Floppy Disk Users 14,15	Keyword 14

Section	Section
Layout 15	Printer 21
Length 3,14,15	Interface G,15,21
Maximum Number in Record 14	Network 15,21,23
Name, truncation of 7,14	Output 21
Names 14,15	Parallel 15
Number 14,15	Serial 15
Numerical 8	Query G
Sorting 8	Archiving 10
FIELD COMMAND STRING sequences 6	Command Language G,6
Field Cursor G,11,12	Complex 6
Free Record G,12	Definition 2,6
Header G,5,7,21,22	Deletion 10
Information Management 2	Libraries 10
Inverse Colours G,7	Maximum Number 6,10
Justification Code G,8	Name 6
Keywords G,14	Recalling Definition 9
Database Organisation, and .. 14	Retrieving 10
Searching, and 14	Revision 9
Keyboard Entries 1,5	Simple 6
Library G,10,18	Standard 5,10
Mistakes, dealing with 3	Updating during edits 11,12
Multi-user Version G,1,23	Record G
New Query Flag G,5,6,9	Definition 2,14
Operators G,6	Field Organisation 14
Options G,3	New 12
Detached 3	New, saving 12
Option Maps G,1,13	Next Free 12
Paper Database 2	Size 3
Pilot Data 14	Unwanted 12
Pitch G,21	Reference Sources 1
Preparing Disks 14	Reports G,2
	Header 5,7,22
	Organisation 5,7
	Report Format G,2
	Advantages 7
	Definition 7
	Field Order 7
	Maximum Number 7,10
	Name 7
	Organisation 5,7
	Printing 21
	Profile 5,10
	Sorting 8
	Width 7,21

Section	Section
Screen Colours 15,23	Sub-database G
Searching G	Option 14,20
Paper Databases 2	Symbols, used in the Guide 1
SupaStore Databases 2,5	Utilities G,13
Search Result G,2,5	Accessing from BASIC 13
Sort Code G	Accessing from SupaStore 13
Sort Flag G,8,9	Floppy Disk User
Clearing of 8	Considerations 13
Sorting 8	Installing 13,23
Ascending 8	Leaving 13
Descending 8	Menu 13
Hierarchical 8	Window G
Message 8	★BASIC 13
Spool File G	★SSEND..... 13
Option 14,21	★SSUTILS 13
Standalone Version G,1,23	★SUPASTORE 5
String G,6,14	