

# **The Role of a Database Package for Research Projects**

**November 2000**



**The University of Reading  
Statistical Services Centre**

**Biometrics Advisory and  
Support Service to DFID**



# Contents

|     |                                                   |    |
|-----|---------------------------------------------------|----|
| 1.  | Introduction                                      | 3  |
| 2.  | Data management in Excel                          | 4  |
| 2.1 | Survey data in Excel                              | 4  |
| 2.2 | Data validation and data entry forms              | 5  |
| 2.3 | Linking data from different sheets                | 7  |
| 2.4 | Activity level data                               | 7  |
| 2.5 | Pivot Tables                                      | 8  |
| 2.6 | Review of Excel                                   | 9  |
| 3.  | Components of a database package                  | 11 |
| 3.1 | Designing the database                            | 11 |
| 3.2 | Input into the tables                             | 16 |
| 3.3 | Verification and validation                       | 18 |
| 3.4 | Using the data                                    | 20 |
| 3.5 | Objects in Access                                 | 21 |
| 3.6 | Exporting from Access                             | 22 |
| 3.7 | Review of Access                                  | 23 |
| 4.  | The “Data Flow”                                   | 24 |
| 5.  | Learning about a database package                 | 25 |
| 5.1 | Employ an outside consultant                      | 25 |
| 5.2 | Working in partnership with an outside consultant | 26 |
| 5.3 | Construct the database in-house                   | 26 |
| 5.4 | Recommendations                                   | 26 |

# 1. Introduction

In our guide entitled “Data Management Guidelines for Development Projects” we emphasised the importance of having a good strategy for data management in research projects. We stated there that where spreadsheets are used, they should be used with the same discipline that is imposed automatically if a database package were used.

The guide entitled “Disciplined Use of Spreadsheet Packages for Data Entry” explains what we mean by using a spreadsheet with “discipline” and the guide called “Excel for Statistics” is designed to help researchers decide on the role of a spreadsheet package for their analyses.

This guide is to help researchers and research managers to decide whether they need to make use of a database package to manage their data. We use Microsoft Access as an example, but the concepts are general and apply equally to any relational database package.

There are many textbooks on Access, but they concentrate primarily on HOW to use the software. This guide concentrates on WHETHER the software is needed and if so, what skills do different members of the project team need.

We assume some familiarity with the spreadsheet package and begin, in section 2, with an example of survey data that have been entered into Excel. We review briefly the concepts, from the data entry guide, of using Excel with discipline to improve the data entry process. This section is partly to introduce the concepts of a database system in relation to a spreadsheet. It is also because most projects will make some use of spreadsheets. The main question is usually something like “Given we are reasonably confident with Excel, why do we need to learn about a database package (Access) as well?”

In section 3 we review the components of a database package and look at how the data we used in section 2 could be entered and managed in Access. We show the database design and look at example forms and reports for entering and extracting the data. In section 4 we consider the “flow” of data during a research project moving from data entry through to the data archiving stage at the end of the project. We consider the role of a database package in this entire process. We finish in section 5 with a brief review of the skills necessary for project staff to be able to use a modern database package in an efficient manner.

## 2. Data management in Excel

In this section we review some aspects of data management within Excel. Many of these topics are covered in more detail in our guide on using Excel with discipline.

### 2.1 Survey data in Excel

The data in Figure 1 are taken from an activity diary study carried out in Malawi. Individuals within households kept a record of activities carried out at four different times of the day. Households are grouped into mbumbas or clusters. A cluster is a set of households for a mother, her adult daughters, their husbands and children. There are therefore **three levels** of data, namely mbumba, household and person. In an Excel workbook it is convenient to store each level of data in a separate sheet. Each sheet is given an appropriate name. This is illustrated in Figure 1.

Figure 1 – Extract from Excel showing many worksheets in one file

|    | A    | B       | C      | D        | E                       | F   | G            | H      |  |
|----|------|---------|--------|----------|-------------------------|-----|--------------|--------|--|
| 1  | ID   | Hsehold | Mbumba | PersonNo | IndividualName          | Age | Relationship | Gender |  |
| 31 | 2311 | 3       | 2      | 11       | Roderick                | 5   | 3            | 1      |  |
| 32 | 2312 | 3       | 2      | 12       | Regina                  | 2   | 3            | 2      |  |
| 33 | 2413 | 4       | 2      | 13       | Mr Mukhumba             | 30  | 1            | 1      |  |
| 34 | 2414 | 4       | 2      | 14       | Olaliya                 | 25  | 2            | 2      |  |
| 35 | 2415 | 4       | 2      | 15       | Donata                  | 8   | 3            | 2      |  |
| 36 | 2416 | 4       | 2      | 16       | Gladys                  | 6   | 3            | 2      |  |
| 37 | 2417 | 4       | 2      | 17       | Charles                 | 3   | 3            | 1      |  |
| 38 | 2518 | 5       | 2      | 18       | Hilda                   | 20  |              | 2      |  |
| 39 | 2619 | 6       | 2      | 19       | Uncle                   | 65  | 7            | 1      |  |
| 40 | 3101 | 1       | 3      | 1        | Mai                     | 70  | 4            | 2      |  |
| 41 | 3102 | 1       | 3      | 2        | Elizabeth               | 45  | 1            | 2      |  |
| 42 | 3103 | 1       | 3      | 3        | Enoch January Manyela   | 23  | 3            | 1      |  |
| 43 | 3104 | 1       | 3      | 4        | Binette January Manyela | 21  | 3            | 2      |  |

In this survey the **mbumba level** includes the name of the mbumba, its location, etc. At the **household level** the family name is stored. The **person level** includes the name, age and gender of the individual. The unique identifier for the person is a combination of mbumba number, household within mbumba and person within mbumba. Thus person **2518** is the 18<sup>th</sup> person in mbumba number 2 and is in the fifth household in mbumba 2. In Figure 1 we can see details of the **person level** sheet. We see that the mbumba and household numbers are also stored at this level and these act as a reference to the **household** and **mbumba level** sheets.

Much of the data that was recorded was of activities. These were recorded at 4 times of the day. They were stored on a fourth sheet as shown in Figure 2, though a better way is shown later. This has introduced a 4<sup>th</sup> level to the data, namely a time-of-day level.

**Figure 2 – Extract from the Activities worksheet in the Excel file**

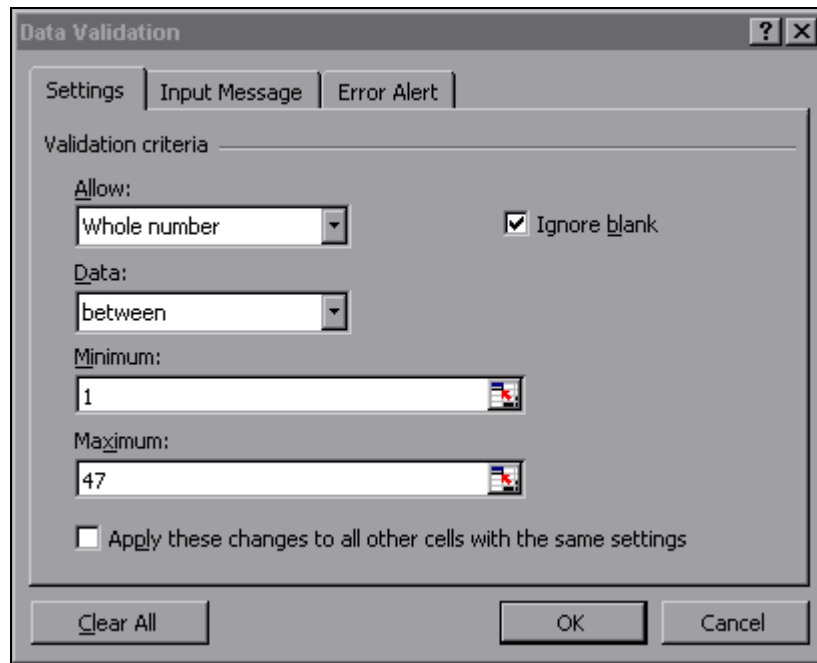
|    | A      | B        | C           | D          | E          | F          | G          | H          | I          | J          | K          | L          | M           |  |
|----|--------|----------|-------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|-------------|--|
|    | PERSON | DATE     | TIME OF DAY | ACTIVITY 1 | ACTIVITY 2 | ACTIVITY 3 | ACTIVITY 4 | ACTIVITY 5 | ACTIVITY 6 | ACTIVITY 7 | ACTIVITY 8 | ACTIVITY 9 | ACTIVITY 10 |  |
| 1  |        |          |             |            |            |            |            |            |            |            |            |            |             |  |
| 2  | 2101   | 01/03/98 | 1           | 14         |            |            |            |            |            |            |            |            |             |  |
| 3  | 2101   | 01/03/98 | 2           | 30         | 29         |            |            |            |            |            |            |            |             |  |
| 4  | 2101   | 01/03/98 | 3           | 17         |            |            |            |            |            |            |            |            |             |  |
| 5  | 2101   | 01/03/98 | 4           | 16         | 29         | 30         |            |            |            |            |            |            |             |  |
| 6  | 2205   | 01/03/98 | 1           | 29         | 28         |            |            |            |            |            |            |            |             |  |
| 7  | 2205   | 01/03/98 | 2           | 29         |            |            |            |            |            |            |            |            |             |  |
| 8  | 2205   | 01/03/98 | 3           | 17         | 11         |            |            |            |            |            |            |            |             |  |
| 9  | 2205   | 01/03/98 | 4           | 30         | 29         |            |            |            |            |            |            |            |             |  |
| 10 | 2206   | 01/03/98 | 1           | 16         | 14         |            |            |            |            |            |            |            |             |  |
| 11 | 2206   | 01/03/98 | 2           | 27         |            |            |            |            |            |            |            |            |             |  |
| 12 | 2206   | 01/03/98 | 3           | 16         | 13         |            |            |            |            |            |            |            |             |  |
| 13 | 2206   | 01/03/98 | 4           | 16         | 17         | 28         | 29         |            |            |            |            |            |             |  |
| 14 | 2518   | 01/03/98 | 1           | 45         |            |            |            |            |            |            |            |            |             |  |

Codes have been assigned to the activities. A coding table is stored in a fifth sheet in the same file. A maximum of 10 activities at any one time of the day is assumed.

## 2.2 Data validation and data entry forms

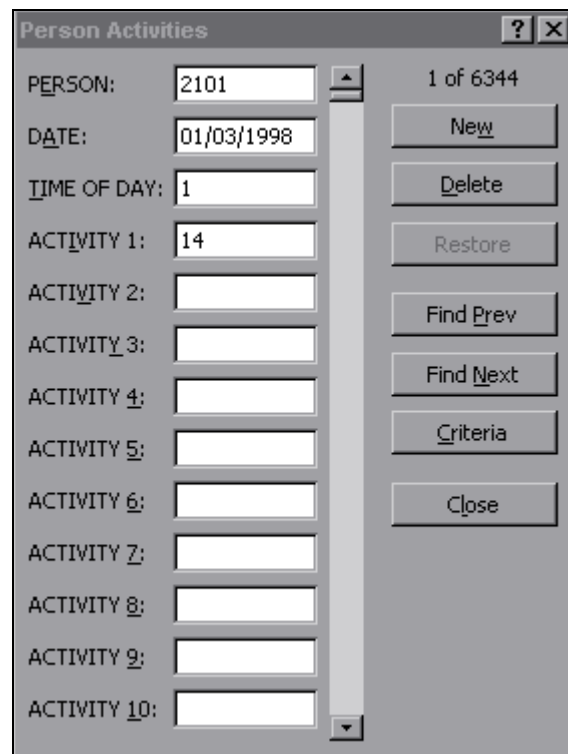
As mentioned in our guide on Excel, it is possible to set validation checks on cells in an Excel worksheet. As there are 47 activities numbered 1 through 47 we can set validation checks on columns D to M as shown in Figure 3.

**Figure 3 – Validation rules in Excel**

The image shows the 'Data Validation' dialog box in Microsoft Excel. It has three tabs: 'Settings', 'Input Message', and 'Error Alert'. The 'Settings' tab is active. Under 'Validation criteria', the 'Allow:' dropdown is set to 'Whole number'. The 'Ignore blank' checkbox is checked. The 'Data:' dropdown is set to 'between'. The 'Minimum:' field contains the value '1' and the 'Maximum:' field contains the value '47'. At the bottom, there is an unchecked checkbox labeled 'Apply these changes to all other cells with the same settings'. There are three buttons at the bottom: 'Clear All', 'OK', and 'Cancel'.

Another useful feature is the ability to use a form for data entry. Choosing **Form** from the **Data** menu produces the form shown in Figure 4

**Figure 4 – Data entry forms in Excel.**

The image shows the 'Person Activities' data form in Microsoft Excel. It is a vertical form with a title bar that says 'Person Activities'. On the left, there are labels for 'PERSON:', 'DATE:', 'TIME OF DAY:', and ten 'ACTIVITY' fields (ACTIVITY 1 through ACTIVITY 10). Each label is followed by a text input box. The 'PERSON' box contains '2101', the 'DATE' box contains '01/03/1998', and the 'TIME OF DAY' box contains '1'. The 'ACTIVITY' boxes are empty. On the right side of the form, there is a vertical scrollbar and a set of buttons: 'New', 'Delete', 'Restore', 'Find Prev', 'Find Next', 'Criteria', and 'Close'. At the top right, it says '1 of 6344'.

When data are entered via a form they are checked against validation rules only at the end of each record and not as each value is entered.

## 2.3 Linking data from different sheets

We have mentioned that each person is assigned a unique identifier. This identifier is used in the **Activities** sheet and acts as a link to the **Person** level data. Using this link we are able to view data from the **Person level** alongside data in the **Activities** sheet. For example, Figure 5 shows the **Activities** sheet with additional columns for **Age** and **Gender**. We have used the **VLOOKUP** function to display data stored in the **Person level** sheet. The key point here is that these data are only stored once – in the **Person level** sheet – but using **VLOOKUP** we are able to view them in other sheets. Storing a data value just once helps to minimise errors. This has been achieved by dividing data into levels and storing each data item at the appropriate level.

Figure 5 - Use of VLOOKUP to combine data from separate worksheets

|    | A      | B        | C   | D      | E           | F          | G          | H          | I          | J          | K          | L          | M          | N          | O           |
|----|--------|----------|-----|--------|-------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|-------------|
|    | PERSON | DATE     | Age | Gender | TIME OF DAY | ACTIVITY 1 | ACTIVITY 2 | ACTIVITY 3 | ACTIVITY 4 | ACTIVITY 5 | ACTIVITY 6 | ACTIVITY 7 | ACTIVITY 8 | ACTIVITY 9 | ACTIVITY 10 |
| 1  | 2101   | 01/03/98 | 55  | 2      | 1           | 14         |            |            |            |            |            |            |            |            |             |
| 2  | 2101   | 01/03/98 | 55  | 2      | 2           | 30         | 29         |            |            |            |            |            |            |            |             |
| 3  | 2101   | 01/03/98 | 55  | 2      | 3           | 17         |            |            |            |            |            |            |            |            |             |
| 4  | 2101   | 01/03/98 | 55  | 2      | 4           | 16         | 29         | 30         |            |            |            |            |            |            |             |
| 5  | 2205   | 01/03/98 | 40  | 1      | 1           | 29         | 28         |            |            |            |            |            |            |            |             |
| 6  | 2205   | 01/03/98 | 40  | 1      | 2           | 29         |            |            |            |            |            |            |            |            |             |
| 7  | 2205   | 01/03/98 | 40  | 1      | 3           | 17         | 11         |            |            |            |            |            |            |            |             |
| 8  | 2205   | 01/03/98 | 40  | 1      | 4           | 30         | 29         |            |            |            |            |            |            |            |             |
| 9  | 2206   | 01/03/98 | 31  | 2      | 1           | 16         | 14         |            |            |            |            |            |            |            |             |
| 10 | 2206   | 01/03/98 | 31  | 2      | 2           | 27         |            |            |            |            |            |            |            |            |             |
| 11 | 2206   | 01/03/98 | 31  | 2      | 3           | 16         | 13         |            |            |            |            |            |            |            |             |
| 12 | 2206   | 01/03/98 | 31  | 2      | 4           | 16         | 17         | 28         | 29         |            |            |            |            |            |             |
| 13 | 2206   | 01/03/98 | 31  | 2      | 4           | 16         | 17         | 28         | 29         |            |            |            |            |            |             |

## 2.4 Activity level data

In this survey respondents were asked to list the activities they carried out at particular times of the day as shown in Figure 5. This is an example of a multiple response question that is common in surveys. A respondent could list one or more activities and the number of activities is different for each person. One way of entering and storing the activity data is as shown in Figures 2 and 5 but it is not entirely satisfactory as it results in a non-rectangular data block. This is seen in Figure 5— few individuals have as many as 10 activities and consequently there are many missing values.

An alternative way of entering these data is to consider an **activity** level rather than a **time-of-day** level. The equivalent to Figure 5 is shown in Figure 6 where each row of

data now refers to an activity rather than a time of day. This layout uses more rows of data but has the advantage of a simple rectangular structure with no arbitrary limit on the number of activities. We will see, in section 3, that this structure is the natural one to use if the data are to be stored in a database package.

**Figure 6 - A single activity per row**

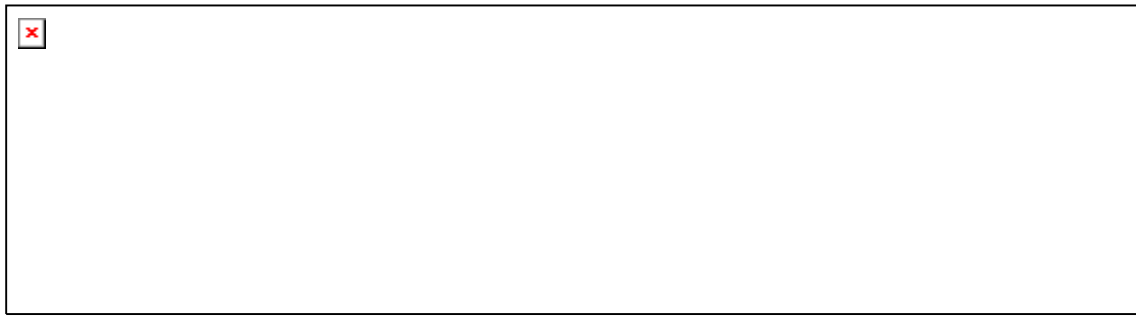
|    | A        | B        | C   | D        |  |
|----|----------|----------|-----|----------|--|
| 1  | PersonID | Date     | TOD | Activity |  |
| 2  | 2101     | 01/03/98 | 1   | 14       |  |
| 3  | 2101     | 01/03/98 | 2   | 30       |  |
| 4  | 2101     | 01/03/98 | 2   | 29       |  |
| 5  | 2101     | 01/03/98 | 3   | 17       |  |
| 6  | 2101     | 01/03/98 | 4   | 16       |  |
| 7  | 2101     | 01/03/98 | 4   | 29       |  |
| 8  | 2101     | 01/03/98 | 4   | 30       |  |
| 9  | 2105     | 01/03/98 | 1   | 21       |  |
| 10 | 2105     | 01/03/98 | 1   | 27       |  |
| 11 | 2105     | 01/03/98 | 2   | 17       |  |
| 12 | 2105     | 01/03/98 | 2   | 29       |  |
| 13 | 2105     | 01/03/98 | 3   | 26       |  |
| 14 | 2105     | 01/03/98 | 3   | 17       |  |
| 15 | 2105     | 01/03/98 | 3   | 28       |  |
| 16 | 2105     | 01/03/98 | 4   | 28       |  |
| 17 | 2105     | 01/03/98 | 4   | 29       |  |
| 18 | 2205     | 01/03/98 | 1   | 29       |  |
| 19 | 2205     | 01/03/98 | 1   | 28       |  |
| 20 | 2205     | 01/03/98 | 2   | 29       |  |
| 21 | 2205     | 01/03/98 | 3   | 17       |  |

## 2.5 Pivot Tables

Once the data are entered they need to be analysed. Simple analyses usually consist of summary tables and graphs: both are standard features of spreadsheet packages. In Figure 7 we illustrate with a summary table that uses Excel's powerful Pivot table feature. These are in effect cross-tabulations with the advantage of being interactive – you can easily swap rows and columns for instance. Figure 7 shows a pivot table created using the activity data, where a subset of the activities have been chosen and are shown as row headings. Individuals have been grouped into boys, girls, men and women depending on their age and gender and these groupings appear as column headings in the table. The cells of the table show the number of records falling into each category. Such tables can give percentages and other summary values. If the original data are changed the pivot table can be refreshed to reflect these changes.



**Figure 7 – Pivot Table in Excel**



## **2.6 Review of Excel**

We can now review some of the strengths and weaknesses of Excel for Scientific data entry and management.

When used with discipline it is adequate for data that has a simple structure. We define “simple structure” as not having too many levels. In the Excel for data entry guide we looked at data with one or two levels and Excel seemed adequate. Here we had 4 levels and this level of complexity has already made Excel more difficult to use.

Notice also that the multiple response question that we discussed earlier on the activities carried out at a particular time of day, was easily handled by entering the activity data into a separate sheet. When surveys have more than one multiple response question the data entry requires yet more tables.

A second similar problem with Excel was shown in Figure 4 where we used a simple data entry form. When we have a lot of data it is sensible to make the data entry process as simple as possible, i.e. to make the form on the screen resemble the questionnaire form, and this can not be done effectively in Excel alone. If you have Access available on your PC it is possible to make use of Access forms from within Excel. This is done via the Microsoft AccessLinks Add-In for Excel. When you use this feature Excel will create an Access database file with your current worksheet as a linked table – changes made to the data within Access will be reflected in the Excel file. With this feature you have more flexibility on the design of the form and can exploit all the form design features of Access. It should be noted, however, that validation rules set up in Excel are not carried through to Access – you would need to set checks on the Access form itself.

A third possible limitation, when we have complicated data structures is that we often have many different ways of wanting to summarise the data. In Excel, it is usually appropriate to consider each such way as equivalent to writing a simple “report” and each will go on a new sheet. Once we have many (report) sheets we have to be sure

that we document our workbook well, so that we can review what we have done on a future occasion.

Excel and other spreadsheets have major strengths. These include the fact that what you are doing is always visible. They are also powerful and very flexible. Set against this is the fact that it is difficult to work with “discipline” if datasets are large and/or complex in their structure. Then a structured approach is needed for entry and management to fully exploit the data. A database package provides this structure.

### 3. Components of a database package

In this section we review briefly the components of a database package. We build on the ideas from section 2, but use standard database jargon. This is so readers will be able to understand consultants and read the literature that extols the virtues of databases. We look at designing the database, inputting the data, and using the data. As an example we use the data from the activity study that we introduced in section 2.

#### 3.1 Designing the database

In a database package data are stored in “tables”. The example in section 2 had four tables, for the mbumba, household, person and activity levels. In a database package the tables must be created before the data can be entered. As a minimum you must specify the number of fields or columns of data required, give a name to each field and define the data type, e.g. text or numeric. This goes some way to enforcing much of the “discipline” that we have encouraged for the use of Excel in both our Excel guide and section 2 of this guide. Figure 8 shows the table design screen. This is where the field names and data types are set up.

Figure 8 - Table design in Access

| Field Name     | Data Type | Description                                                     |
|----------------|-----------|-----------------------------------------------------------------|
| ID             | Text      | The Unique identifier for the person - this is made up of the r |
| Hsehold        | Number    | Household number links to the household table                   |
| Mbumba         | Number    | Mbumba and household link to the household table where the      |
| PersonNo       | Number    | Person number within the household                              |
| IndividualName | Text      | The name of the individual                                      |
| Author         | Yes/No    | Whether or not this person was an "author" ie. did they keep    |
| Age            | Number    | Allow for decimal places here if necessary - young children m   |
| Relationship   | Number    | Links to the relationship table - the number code will be store |
| Gender         | Number    |                                                                 |

| Field Properties  |                     |
|-------------------|---------------------|
| General           | Lookup              |
| Field Size        | 4                   |
| Format            |                     |
| Input Mask        | 0000                |
| Caption           |                     |
| Default Value     |                     |
| Validation Rule   |                     |
| Validation Text   |                     |
| Required          | No                  |
| Allow Zero Length | No                  |
| Indexed           | Yes (No Duplicates) |

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

The table design screen in Figure 8 shows the design of the Person level table. We must specify names for the fields and define their data types. Once the table is created we can enter data via the datasheet or spreadsheet view. This is shown in Figure 9.

The datasheet resembles the Excel worksheet. The datasheet is tailored to the data you want to enter; each column refers to a field in the table and will only accept data of the type specified in the table design. There is no limit to the number of rows you can enter other than the physical limit on the size of your disk. One difference you may notice between the datasheet in Access and the worksheet in Excel is that there is no automatic numbering of the rows in Access. However, information at the bottom of the window tells you which record or row you are on and how many records there are in total.

Figure 9 – “Datasheet” view of Person level data

| ID   | Hsehold | Mbumba | PersonNo | IndividualName | Author                              | Age | Relationship | Gender |
|------|---------|--------|----------|----------------|-------------------------------------|-----|--------------|--------|
| 2101 | 1       | 2      | 1        | Mai Mazinga    | <input type="checkbox"/>            | 55  | 1            | Female |
| 2102 | 1       | 2      | 2        | Mercy          | <input checked="" type="checkbox"/> | 18  | 3            | Female |
| 2103 | 1       | 2      | 3        | Tokozani       | <input type="checkbox"/>            | 15  | 3            | Male   |
| 2104 | 1       | 2      | 4        | Charity        | <input type="checkbox"/>            | 1   | 6            | Female |
| 2105 | 1       | 2      | 5        | Unknown        | <input type="checkbox"/>            |     |              |        |
| 2205 | 2       | 2      | 5        | Mr Nangwale    | <input checked="" type="checkbox"/> | 40  | 1            | Male   |
| 2206 | 2       | 2      | 6        | Martha         | <input type="checkbox"/>            | 31  | 2            | Female |
| 2207 | 2       | 2      | 7        | Enifa          | <input type="checkbox"/>            | 11  | 3            | Female |
| 2308 | 3       | 2      | 8        | Frank Filipo   | <input type="checkbox"/>            | 30  | 10           | Male   |
| 2309 | 3       | 2      | 9        | Femia          | <input type="checkbox"/>            | 27  | 2            | Female |
| 2310 | 3       | 2      | 10       | Mundolani      | <input type="checkbox"/>            | 8   | 2            | Female |

As with the use of a spreadsheet, it is important that you use a database package “with discipline”. Minimal discipline – defining the number of fields and their data type – is enforced but you should normally do more than the minimum. As an example we explain why it is important that all tables have what is called a **primary key**.

All data, whether stored in a database, spreadsheet, or elsewhere, should have a unique identifier for each record. This may be a single field or a combination of fields. In Excel and other spreadsheets there is no way to enforce uniqueness for this identifier and thus duplicates can occur. In Access and other database packages however you can and should set a **primary key** for each table. This is either a single field or combination of fields, which acts as a unique identifier. The primary key is always unique – Access does not allow for duplicates in the primary key. At the Person level the unique identifier is the ID. Referring again to Figure 8 we see that this field has a key symbol by the side of it indicating that this is the primary key field for this table. In many cases the choice of primary key field is obvious.

Now consider a situation where the primary key field is not so clear-cut. Data at the Activity level include *PersonID*, *Date*, *TOD*, *Activity*. An extract of these data is shown in Figure 10.

Figure 10 – “Datasheet” view of Activity level data

| PersonID | Date     | TOD | Activity |
|----------|----------|-----|----------|
| 2101     | 01/03/98 | 1   | 14       |
| 2101     | 01/03/98 | 2   | 30       |
| 2101     | 01/03/98 | 2   | 29       |
| 2101     | 01/03/98 | 3   | 17       |
| 2101     | 01/03/98 | 4   | 16       |
| 2101     | 01/03/98 | 4   | 29       |
| 2101     | 01/03/98 | 4   | 30       |
| 2102     | 01/03/98 | 1   | 14       |
| 2102     | 01/03/98 | 1   | 27       |
| 2102     | 01/03/98 | 1   | 21       |
| 2102     | 01/03/98 | 1   | 13       |
| 2102     | 01/03/98 | 2   | 17       |
| 2102     | 01/03/98 | 2   | 28       |

Clearly none of these fields is unique by itself. Thus we must look at combinations of fields and when we do that we find that the only combination that must be unique is the combination of all four fields. It is possible to define this combination as our primary key, however, multi-field primary keys comprising more than 2 fields, become difficult to handle and can easily lead to mistakes when setting relationships.

An alternative is to use an *autonumber* field as the primary key. This will assign a unique number to each record. However, we still want to ensure that the combination of the four original fields is unique. This we can do by creating what Access refers to as an “index”.

An index can be created for any field and any combination of fields and speeds up the processes of sorting and selecting. Once an index has been created it can be made unique, in other words you would not be able to enter duplicates into that field or combination of fields.

Figure 11 shows the table design screen for the Activity level data and includes the autonumber field that we have added as the primary key. We can also see the **Index** window which shows that there is an index called “identifier” which is a combination of the original four fields. The **Unique** property has been set to “Yes” for this index.

Figure 11 – Table design with Index window

| Field Name       | Data Type  | Description                                                                                        |
|------------------|------------|----------------------------------------------------------------------------------------------------|
| PersonActivityID | AutoNumber | Automatic number field to assign a unique ID to each record                                        |
| PersonID         | Text       | Link to the person table                                                                           |
| Date             | Date/Time  | The date in question                                                                               |
| TOD              | Number     | Lookup field looking up the Time of day from the TOD table                                         |
| Activity         | Number     | Lookup field showing the activity - there can be many records for each person/date/tod combination |

| Index Name | Field Name       | Sort Order |
|------------|------------------|------------|
| Identifier | PersonID         | Ascending  |
|            | Date             | Ascending  |
|            | TOD              | Ascending  |
|            | Activity         | Ascending  |
| PrimaryKey | PersonActivityID | Ascending  |

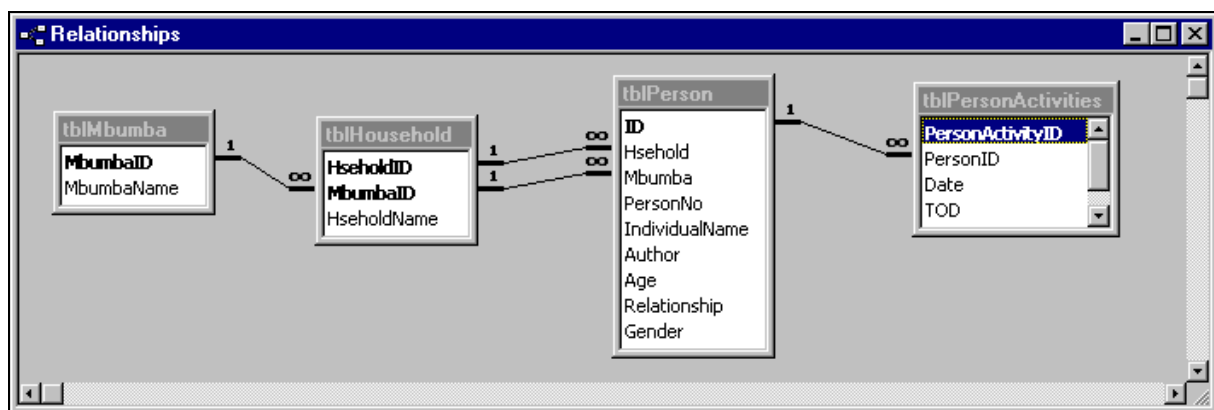
  

| Index Properties |     |
|------------------|-----|
| Primary          | No  |
| Unique           | Yes |
| Ignore Nulls     | No  |

The name for this index. Each index can use up to 10 fields.

An important extra that comes with relational database packages such as Access, is the ability to create relationships or links between tables of data. This is implied in our discussion earlier on Excel when we talked about linking data from different sheets using *VLOOKUP*. These links can be built into the design in Access. Figure 12 shows the same structure of data that we developed in Excel but in Access. The 4 levels are translated into 4 tables with relationships between them. The relationships are all “one-to-many” in that a single record in one table is related to potentially many records in another table. For example each household has many individuals.

Figure 12 : Database structure in Access



Access includes a set of rules known as *Referential Integrity*. When enforced this helps to validate relationships by not allowing you to enter a record in a table on the “many” side of a relationship where there is no corresponding record in the table on the “one” side. For example with referential integrity you would not be able to enter details of an individual until there was a household for that individual.

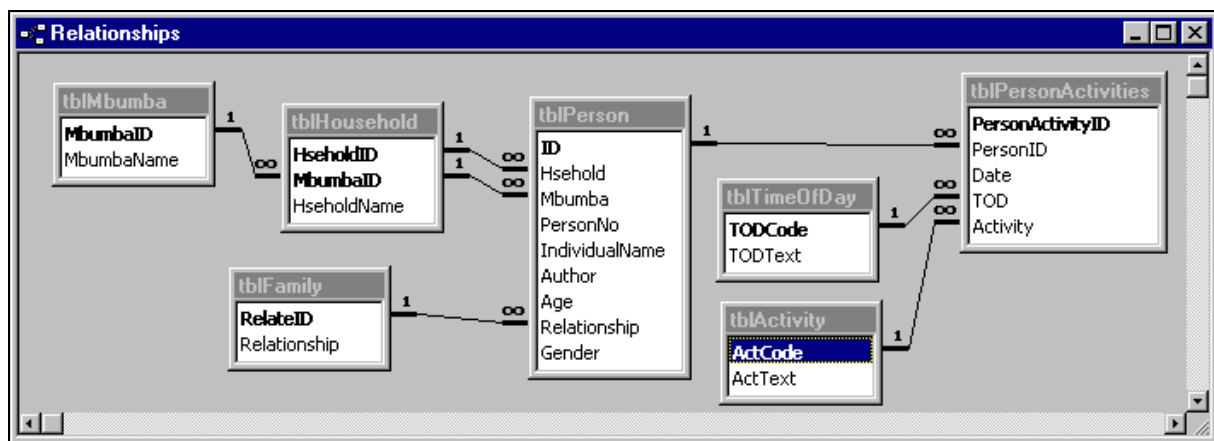
Once you realise the value of multiple tables you will find that you can use more of them. Consider for example the “activities” in our example set of data. The activities are coded from 1 to 47 and the code is stored in the database. It would be relatively easy to add a 2-column table containing these codes and their associated descriptions. Figure 13 shows some of the data from the ‘Activities’ table and Figure 14 shows how this table and similar tables for the ‘time of day’ and ‘family relationship’ can be added to the database structure.

**Figure 13 - Extract from the table of Activities**

| tblActivity : Table |                                                                                                      |
|---------------------|------------------------------------------------------------------------------------------------------|
| ActCode             | ActText                                                                                              |
| 1                   | Cultivating/checking the field                                                                       |
| 2                   | Planting/transplanting crops                                                                         |
| 3                   | Tilling/ridging                                                                                      |
| 4                   | Sowing seeds/preparing seeds/materials for planting/maintaining nursery                              |
| 5                   | Weeding                                                                                              |
| 6                   | Banking                                                                                              |
| 7                   | Fertilising a field/dimba or applying manure/watering crops                                          |
| 8                   | Applying pesticides/removing infected plants                                                         |
| 9                   | Harvesting or hauling crops from the fields, e.g. maize, cassava, potatoes, etc.                     |
| 10                  | Buying agricultural inputs/collecting starter packs and other benefits                               |
| 11                  | Feeding animals/Dipping animals/Gathering animal feed/ Cleaning a khola/Leading livestock to a khola |
| 12                  | Gathering firewood and chopping wood                                                                 |
| 13                  | Drawing water                                                                                        |
| 14                  | Sweeping, washing, weeding, cleaning, chasing mosquitoes, killing insects, etc.                      |
| 15                  | Harvesting green crops for relish                                                                    |
| 16                  | Preparing food, making porridge                                                                      |
| 17                  | Eating                                                                                               |
| 18                  | Going to a milling machine                                                                           |
| 19                  | Pounding, winnowing, unsheathing, shelling, drying maize                                             |
| 20                  | Making fire                                                                                          |
| 21                  | Bathing self or child, Shaving                                                                       |

Unlike a spreadsheet, where seven tables with data would be confusing, this is quite a simple structure for a database. A database would typically have between 5 and 50 tables.

Figure 14 - Complete Database structure



### 3.2 Input into the tables

The next aspect we need to consider is how to get data into the tables. We mentioned earlier that data can be entered directly into the table via the datasheet. We saw an example datasheet in Figure 9. When there is just a small amount of data this is easy and is all that is needed. Figure 15 shows all five records from the mbumba table in “spreadsheet” view.

Figure 15 – Mbumba level data

|   | MbumbaID | MbumbaName |
|---|----------|------------|
| ▶ | 1        | MUTHOWA    |
|   | 2        | MAZINGA    |
|   | 3        | MARICHI    |
|   | 4        | CHIMVULA   |
|   | 5        | SIMEON     |

Record: 1 of 5

For larger volumes of data it is more common to set up special data entry forms. These need more practice than in Excel but simple forms are very easy to design. The form in Figure 16 is for entering data on individuals. This form was generated automatically from the corresponding table using one of the autoform wizards in Access, and is similar in structure to Figure 4 that was given in Excel.



Figure 16 – Simple Person level data entry form

tblPerson

ID: 1101

Hsehold: 1

Mbumba: 1

PersonNo: 1

IndividualName: Mr Muthowa

Author: ☐

Age: 70

Relationship: 1

Gender: M

Record: 1 of 94

Figure 17 shows the same form after a few simple design changes. Thus it is easy to start with an automatically generated form and change the layout to match your questionnaire. The ease of producing forms of this type in Access is one of the reasons for its popularity.

Figure 17 – Variation on Person level data entry form

tblPerson

**Activity Study Database**  
**Data on Individuals**

ID: 1101

Name: Mr Muthowa

Household No.: 1

Author: ☐

Mbumba No.: 1

Age: 70

Person Number within household: 1

Relationship to head of household: 1

Gender: ☒ Male ☐ Female

Record: 1 of 94

However, a survey form often includes data from more than one table. In our case the person form included space to record the activity level data. Ideally we would therefore want to enter data from a single questionnaire into 2 or even 3 tables at the same time. This further step does require some degree of expertise but is also relatively easy in a database package such as Access. This is important as it makes the data entry much easier and hence more reliable.

Figure 18 shows a form that was used in this study. The top part of the form is for entering data on the individuals. This is similar to the forms in Figures 16 and 17. The bottom half of the form is for entering activity data. This is actually a *sub-form* and data entered here are stored in the activity table.

Figure 18 – Person level form with Activity level sub-form

**Activity Study Database**  
**Data on Individuals**

ID: 1206      Name: Elaton Naluso      Gender: ☐ Male ☒ Female  
Mbumba: MUTHOWA      Relationship to Head of Household:        
Household: Naluso      Age: 21  
Person Number within household: 6      ☒ Author

**Daily Activities**

| Date      | Time of Day    | Activity                                                                           |
|-----------|----------------|------------------------------------------------------------------------------------|
| 31-May-98 | Evening        | 17 Eating                                                                          |
| 01-Jun-98 | Morning        | 21 Bathing self or child, Shaving                                                  |
| 01-Jun-98 | Morning        | 14 Sweeping, washing, weeding, cleaning, chasing mosquitoes, killing insects, etc. |
| 01-Jun-98 | Morning        | 13 Drawing water                                                                   |
| 01-Jun-98 | Morning        | 40 Attending school/related activities                                             |
| 01-Jun-98 | Afternoon      | 23 Building/looking for building materials, fencing                                |
| 01-Jun-98 | Afternoon      | 17 Eating                                                                          |
| 01-Jun-98 | Late Afternoon | 21 Bathing self or child, Shaving                                                  |
| 01-Jun-98 | Late Afternoon | 13 Drawing water                                                                   |

Record: 1 of 713  
Record: 7 of 94

Because of links between the main form and the sub-form you only see the activity data for the individual displayed in the main form. Generally there is a one-to-many relationship between data in the main form and data in the sub-form. In Figure 18 we can see that this particular individual has several activities for the morning of 1<sup>st</sup> June 1998. Thus the multiple response question on the different activities in each time period translates into a separate record for each response.

In our Excel guide we emphasised the importance of distinguishing between the person who designs the “system” for data entry and the staff who do the actual entry. This is now a much clearer distinction with a database package. If there is a complex survey or database it becomes a skilled task to design an effective data entry system.

### 3.3 Verification and validation

In the following discussion we consider **validation** to mean checking the data as it is entered, and interpret **verification** as checking the data once it has been entered. The

**auditing** process we describe in our Excel guide can be thought of as verification in this definition.

In the entry of survey data it is important that the data are verified. This may be by providing checks as the data are entered or by a system such as double entry. A double entry system is where two data entry operators enter the same data into separate files and the files are then compared. Differences are checked against the original paper version of the data.

Double data entry is an automatic feature in some software that is designed for survey data entry. In some cases this software combines many of the database concepts described so far with easy facilities for double data entry. One example of such software is **Epi-Info**. This is free and can be downloaded from the web.

Epi-Info offers a rigorous method of data verification. After records have been entered and saved in a file there is an option to re-enter and verify records in the existing data file. Data are entered exactly as for new records. When a field entry matches the data in the file the cursor moves on exactly as for new entries. When an entry does not match a message appears and the operator is given a chance to re-enter the value or to compare the original entry with the new one and make a choice.

Data from Epi-Info can be imported into Access. It is therefore possible to use something like Epi-Info for the main data entry phase and then transfer the data to Access for storage and management.

In Access you can set validation rules on individual fields. Figure 11 shows a validation rule of ***Between 1 and 47*** for the Activity field. It is also possible to set validation rules on the table. This might be used for example where the value in one field cannot exceed the value in another field. As an example assume we were storing the number of people in a household and the number of children. Obviously there cannot be more children than there are people so we can set a validation rule of ***[People]>[Children]*** for the table.

Database packages such as Access were primarily designed for business users where the process of entering data and using the data is an on-going cycle. The case for double entry is less clear in these circumstances and is not provided by Access or other similar database packages. In surveys and scientific work on the other hand there is a recognised data entry phase and in these cases verification is necessary.

For any given application it is relatively easy to construct a simple double-entry system in Access.

### 3.4 Using the data

In Excel, we showed in Figure 7 how a Pivot table was used to summarise and present the data. In Access we use queries and reports to do the same thing.

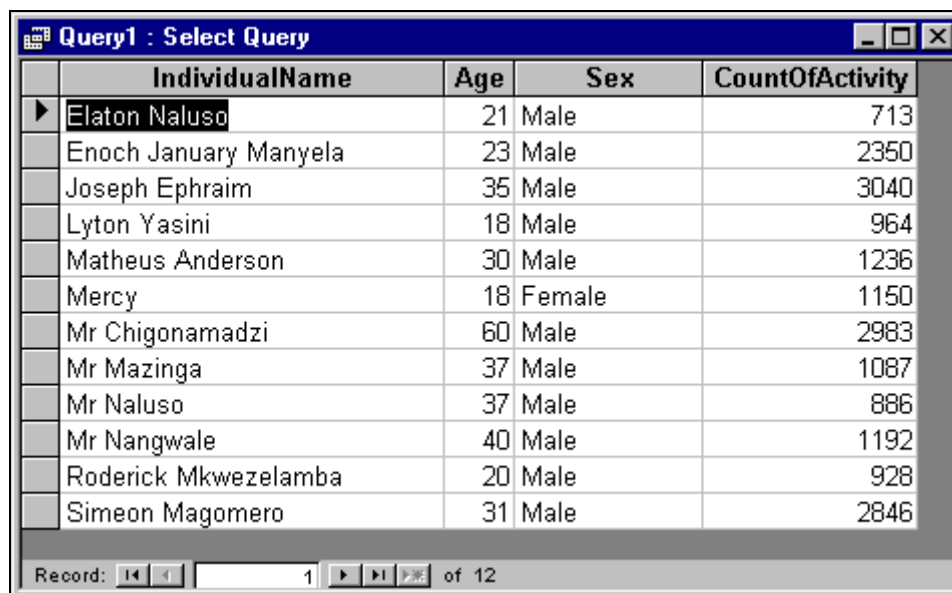
A simple query provides a way to view or summarise subsets of data from a given table in the database. An example is shown in Figure 19, which is similar to the pivot table produced by Excel.

Figure 19 – Crosstab query; equivalent to Pivot table



However, the idea of a database is that the tables are linked. Hence it will be no surprise to find that queries can involve data from multiple tables. Figure 20 shows the results of a query that includes data at both the person level and the activity level. The query counts the number of activities for each individual.

Figure 20 – Query counting activities for selected individuals



| IndividualName        | Age | Sex    | CountOfActivity |
|-----------------------|-----|--------|-----------------|
| Elaton Naluso         | 21  | Male   | 713             |
| Enoch January Manyela | 23  | Male   | 2350            |
| Joseph Ephraim        | 35  | Male   | 3040            |
| Lyton Yasini          | 18  | Male   | 964             |
| Matheus Anderson      | 30  | Male   | 1236            |
| Mercy                 | 18  | Female | 1150            |
| Mr Chigonamadzi       | 60  | Male   | 2983            |
| Mr Mazinga            | 37  | Male   | 1087            |
| Mr Naluso             | 37  | Male   | 886             |
| Mr Nangwale           | 40  | Male   | 1192            |
| Roderick Mkwezelamba  | 20  | Male   | 928             |
| Simeon Magomero       | 31  | Male   | 2846            |

The results from a query can be used in a report, used as the basis for further queries, viewed with a form, exported to another package or stored in a new table.

Another way of using the data in Access is to create reports. A report provides a “snapshot” of the data at a given time. They can be designed to show the same sort of data that you would see in a query but they extend the idea of a query by allowing a

display of the data or summary to suit your needs. The extract below in Figure 21 is taken from a report that lists the activities for each individual and for each time period.

**Figure 21 – Report listing activities for each time period**

|                                                                 |                                                                                 |
|-----------------------------------------------------------------|---------------------------------------------------------------------------------|
| <b><u>Activity Study</u></b>                                    |                                                                                 |
| <b><u>Name:</u></b> Mai Mazinga                                 | <b><u>14 January 1998</u></b>                                                   |
| <b><u>Age:</u></b> 55                                           | <b><u>Morning</u></b>                                                           |
| <b><u>12 January 1998</u></b>                                   | 1 Eating                                                                        |
| <b><u>Morning</u></b>                                           | 2 Cultivating/checking the field                                                |
| 1 Attending a funeral rite                                      | 3 Hearing cases                                                                 |
| 2 Harvesting green crops for relish                             | 4 Preparing food, making porridge                                               |
| 3 Preparing food, making porridge                               | <b><u>Afternoon</u></b>                                                         |
| <b><u>Afternoon</u></b>                                         | 1 Preparing food, making porridge                                               |
| 1 Preparing food, making porridge                               | 2 Gathering firewood and chopping wood                                          |
| 2 Eating                                                        | 3 Playing, resting, sleeping, basking in the sun, playing games                 |
| <b><u>Late Afternoon</u></b>                                    | <b><u>Late Afternoon</u></b>                                                    |
| 1 Gathering firewood and chopping wood                          | 1 Preparing food, making porridge                                               |
| 2 Preparing food, making porridge                               | 2 Eating                                                                        |
| 3 Eating                                                        | 3 Chatting, visiting friends, families, places, going out for social activities |
| <b><u>Evening</u></b>                                           | <b><u>Evening</u></b>                                                           |
| 1 Playing, resting, sleeping, basking in the sun, playing games | 1 Playing, resting, sleeping, basking in the sun, playing games                 |
| <b><u>13 January 1998</u></b>                                   |                                                                                 |

Unlike Excel, when you save queries and reports you do not generally save the results. Instead you save the instructions that produce the results. Whenever a query or report is run the data are taken from the underlying table(s). Thus the results always reflect recent changes in the data. This is a little like “refreshing” a Pivot table in Excel so it reflects any changes in the data. The results of a report can be viewed on screen, sent to a printer or saved in a “snapshot” file. Access 2000 includes a **Report Snapshot Viewer** which is used to view these snapshot files. The viewer can be acquired separately from Access and an add-in is available for Access 97 so that snapshot files can be saved from there.

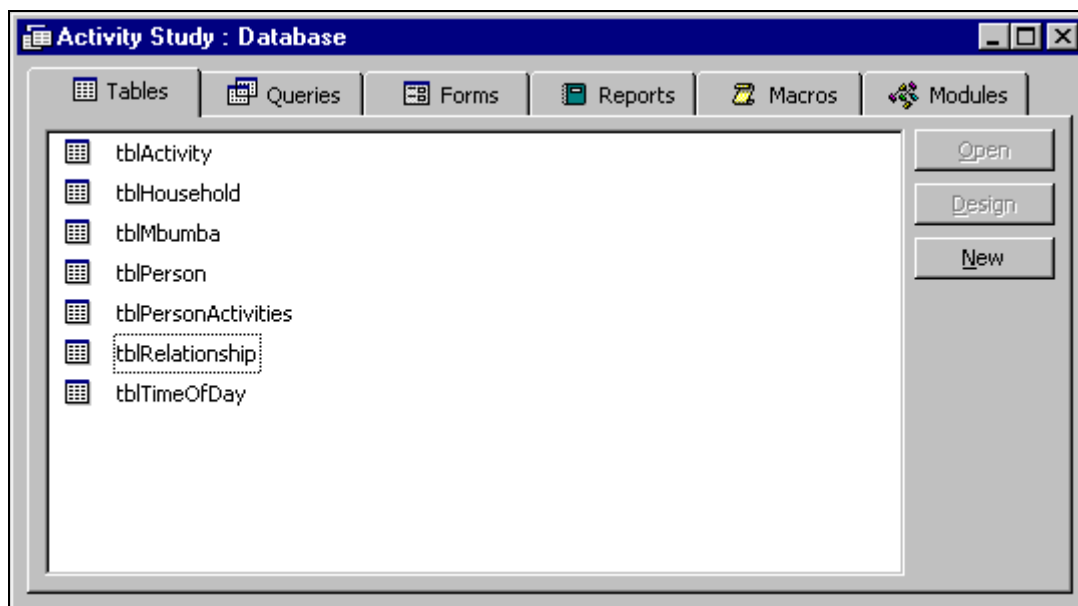
Because Access stores the instructions to run the queries and reports it is possible to do a pilot survey, or just collect a few records initially and develop all the queries and reports you want, based on just these few records. The data are just used to check that you are producing the right looking table or summary. Then, when you have entered all the real data, you just run the queries and/or reports to produce the results.

### 3.5 Objects in Access

Access refers to tables and forms as **objects**. An Access database can include up to six different types of object. We have so far mentioned four of these namely tables, forms, queries and reports. The remaining two, macros and modules, can be used to

automate tasks and pull the other objects together into a user-friendly database application. Use of these objects is not essential for good data management practice. All the objects within a database are accessible from the main database window, an example of which is shown in Figure 22 below.

Figure 22 - Database Window from Access



The objects are grouped by type and by clicking on the appropriate tab it is easy to move from the list of tables to the list of forms for example. This is an example of a data management “system”.

### 3.6 Exporting from Access

One aspect that often discourages users from adopting a database package such as Access is the difficulty they perceive in extracting data in a format ready for analysis. However, by its very nature Access is more flexible in this respect than Excel. Using **queries** it is easy to extract subsets of the data based on given criteria, view data from linked tables, summarise data, and perform simple calculations and summaries. Data produced from queries can, at the click of a button, be exported into Excel.

Many statistics packages such as SPSS, GENSTAT and MINITAB now use ODBC (*Open DataBase Connectivity*) to import data directly from database packages. Transferring data between packages is no longer the problem it once was.

You may wonder why, if you already have your data in Excel, should you move to Access only to be told that you can easily export it back to Excel. What we are suggesting is to store and manage your data in Access and then extract subsets of it to Excel or another package for analysis as and when needed. Ease of data transfer enables you to use the best features from each package.

### 3.7 Review of Access

We concluded section 2 by examining the positive and negative points about spreadsheets. Here we do the same with database packages.

On the positive side, database packages are designed to handle large and complex volumes of data. We believe that their avoidance, in favour of spreadsheets is risky in the task of exploiting research data fully. Database packages enforce much of the “use with discipline” that we have encouraged in the Excel guide and in the first part of this guide. Using a database does not guarantee that you will have complete, error-free data, but used efficiently they can move you nearer to that goal. Merely by having to design tables for your data you are forced to think about the data and its structure. This in itself is a good thing.

We saw in section 3.4 how the different objects in Access are kept separate and easily accessible within the database. This differs substantially from Excel where data and results, pivot tables, calculations, charts, and so on, are all stored in the same way as sheets within the workbook. Unless you are well organised and document all your work it is not always easy to find the sheet you are looking for. In Access the data and results are separated. In general the results are not stored in the database but are generated each time the query or report is run.

A database may have to be used as a final step, to leave a usable archive after the project has finished. In such cases it is more efficient to use a database from the start, so the project team can benefit from the system for data management.

On the negative side, some expertise is needed to construct an effective database. Sometimes we find that inexperienced users do not add the relationships of the type shown in Figures 12 and 14. A database without relationships is just like a spreadsheet, except it is harder to see all the data.

## 4. The “Data Flow”

In this section we consider the “flow” of data during the lifetime of a research project and think about the role of the database package in this process.

We can consider four aspects here namely data entry, data management, data analysis and data archiving. For large volumes of data or data collected at more than two levels we would recommend using a database package for the data entry and data management. One of the roles of data management is to provide good quality data for analysis. We have already said that the use of a database package does not in itself guarantee this but when used effectively with validation checks, primary key fields, referential integrity on relationships and so on, we can at least move in the right direction.

Access is not generally sufficient for data analysis. Cross-tabulations are possible using queries but Excel’s Pivot Table feature is far more flexible. Charts in Access are extremely limited. This is the point where subsets of data should be exported to other packages. It is important to realise that once data are exported you have duplication – if you notice an error in the exported data the correction must be made in the database and the data exported again. If this is not done then data integrity can be compromised. The database should contain the definitive copy of the data.

Data archiving can be thought of as just a copy of the database containing all the project data, but it can be so much more. Ideally it should also include copies of graphs, results of analyses and copies of programs run on the data. An archive CD should include all the output files and data files, whether they be in Excel, SPSS, Access or whatever. This all needs to be documented and one way to document it is to use a database. We saw earlier how easy it is to add additional tables to the database; why not add a table to store information about the analyses that have been carried out. A record could include the name of the data file, the name of the results file, the software used, the type of analysis, the date the analysis was carried out, the person running the analysis, and so on. In addition to text and numeric data Access can store images. It is therefore also feasible to scan photos and maps and store them in the database as images.



## **5. Learning about a database package**

In this section we consider team members who have some skills in Excel and are considering incorporating a database package into their work. With a spreadsheet, such as Excel, it is often adequate for individual staff members to start without a formal training course and simply add to their knowledge of the package as the need arises.

Spreadsheets are normally used on an individual basis, with data to be shared often being copied to each person. Databases can be used in the same way, but it is usually more effective to share the data from within a single database. This is the natural way to operate when computers are networked, but applies even if the database is on a single stand-alone machine.

Thus the establishment of one or more databases will normally involve decisions of responsibility for the entry, validation and use of the data. This extra formality is usually also important to ensure good quality data.

When a database package is used, alternative ways range from employing an outside consultant to proceeding in a similar step-by-step approach to that often used for Excel. We consider these alternatives in turn.

### **5.1 Employ an outside consultant**

One option is to employ an outside consultant or database professional to construct each database for the project. You inform the consultant on the data elements that need to be stored and specify how you would like to enter, view and extract the data. He/she then creates the database structure together with a set of queries, forms and reports. The consultant could also produce a “front-end” to your database so that reports can be run and data extracted at the click of a button. This effectively turns your database into an application. At this level all the project team need to know is how to run this application. Instruction on how to use the database might take perhaps half a day at the end of the consultancy.

This option requires very little time and effort from the project team members.

However, this is a dangerous and expensive option and we would not normally recommend this route. If no team member understands sufficient database principles, it is often difficult to specify exactly what is needed. Defects in the specification then normally become clear when the initial system is delivered and more time and expense is needed to improve the system.

Changes and additions are often required during the life of the project and it is both time-consuming and expensive to return to an outside consultant each time.

Finally, although it is easy to find database consultants, most are experienced in business applications and you may be posing new challenges for them, both in the data entry requirements and in the necessary queries and reports.

## **5.2 Working in partnership with an outside consultant**

We suggest that some database knowledge is needed by project team members, for them to be able to work constructively with a consultant. For staff who are already familiar with Windows and Excel this might typically be a course of between 2 days and a week, with up to half the time being spent on the construction of queries and reports.

The difficult part of the work is the setting up of the initial database, with the relationships and the data entry forms. We suggest that a consultant might be used for this work. The system as delivered, would also contain some queries and reports.

In section 3.3 we discussed verification. This should be considered at the design stage of the database. Remember an outside consultant may not have considered this aspect and it is therefore important that you clearly describe your requirements in this aspect.

It is then relatively easy for the project staff to add extra queries or reports as needed. They could also make minor modifications to the structure. There is however a difference in these two types of task. An error in a query will only affect the person who wishes to run the query, but an error when changing the worksheet structure could render the database unusable.

## **5.3 Construct the database in-house**

The final level is to construct the entire database in-house. This is the obvious approach, if one of the project team is a database expert, but otherwise we counsel caution. It is just as easy to construct a poor database as it was to write a poor programme in earlier days. The diagram of relationships resembles a plate of spaghetti and it becomes difficult to write reports or to modify the structure in any way.

## **5.4 Recommendations**

In project teams that do not include a database expert, we suggest that the partnership approach is normally appropriate. The major change in database software in recent years has been the ease with which users who have relatively little experience can modify a system once it is in place.

Whereas with Excel, there might be equal training, if any, for all team members, we suggest it is normally appropriate to select a subset of the team for training in the basics of database management. They, perhaps in conjunction with a consultant, would then deliver a one-day course on the principles of the current system for the project data, once a test version is available.

Data entry staff would have special training. Their task should be simpler because of the facilities available in a database system to facilitate efficient data entry. If the data entry is not simple, then the project team should demand that improvements be made.

## **Acknowledgements**

The datasets used in this guide were a component of socio-anthropological studies conducted within the DFID-funded Farming Systems Integrated Pest Management (FSIPM) project in Malawi. We are grateful to Julie Lawson-McDowall (Social Anthropologist) and Mark Ritchie (Team Leader, FSIPM) for permission to use the data.

The Statistical Services Centre is attached to the Department of Applied Statistics at The University of Reading, UK, and undertakes training and consultancy work on a non-profit-making basis for clients outside the University.

These statistical guides were originally written as part of a contract with DFID to give guidance to research and support staff working on DFID Natural Resources projects.

The available titles are listed below.

- *Statistical Guidelines for Natural Resources Projects*
- *On-Farm Trials – Some Biometric Guidelines*
- *Data Management Guidelines for Experimental Projects*
- *Guidelines for Planning Effective Surveys*
- *Project Data Archiving – Lessons from a Case Study*
- *Informative Presentation of Tables, Graphs and Statistics*
- *Concepts Underlying the Design of Experiments*
- *One Animal per Farm?*
- *Disciplined Use of Spreadsheets for Data Entry*
- *The Role of a Database Package for Research Projects*
- *Excel for Statistics: Tips and Warnings*
- *The Statistical Background to ANOVA*
- *Moving on from MSTAT (to Genstat)*
- *Some Basic Ideas of Sampling*
- *Modern Methods of Analysis*
- *Confidence & Significance: Key Concepts of Inferential Statistics*
- *Modern Approaches to the Analysis of Experimental Data*
- *Approaches to the Analysis of Survey Data*
- *Mixed Models and Multilevel Data Structures in Agriculture*

The guides are available in both printed and computer-readable form. For copies or for further information about the SSC, please use the contact details given below.



**Statistical Services Centre, University of Reading**  
**P.O. Box 240, Reading, RG6 6FN United Kingdom**  
**tel: SSC Administration +44 118 378 8025**  
**fax: +44 118 378 8458**  
**e-mail: [statistics@lists.reading.ac.uk](mailto:statistics@lists.reading.ac.uk)**  
**web: <http://www.reading.ac.uk/ssc/>**