

Recorder 2002

**SQL Reports addin for
Recorder 2002: A guide for
Query authors.**

Stuart Ball

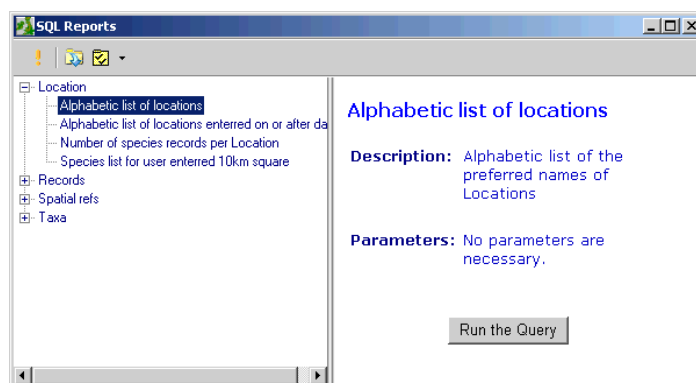
SQL Reports addin for *Recorder 2002*: A guide for Query authors.

Introduction

This addin gives a user of *Recorder 2002* the ability to run queries stored in the main Access 97 database (nbndata.mdb) and print the results table, or save it in a variety of file formats including Excel spreadsheet or a table in a Word document.

Queries may prompt the user to provide necessary information (e.g. a query to list species recorded in a 10km square needs to prompt for the 10km square for which a list is desired) or utilise lists from the current rucksack (e.g. a query to count the number of records associated with particular Locations could get the list of Locations on which to report from the "Locations" list in the currently open rucksack).

Example



The screenshot shows the 'Results of query "Alphabetic list of locations"' window. It displays a table with two columns: 'ITEM_NAME' and 'SPATIAL_REF'. The table contains 217 rows of data, with the first row highlighted. The data is as follows:

ITEM_NAME	SPATIAL_REF
A test site	NZ1063
Aldbury Nowers H&MWTR	SP951133
Aldenham Country Park	TQ166958
Anglesey	SH4481
Angus (Forfarshire)	NO4456
Antrim	D21
Armagh	J03
Ayrshire	NS3718
Baldwins Wood	TQ017991
Banffshire	NJ3837
Batch Wood St Albans	TL138092
Batchworth Heath	TQ0792
Bath Western Riverside	ST7364
Bedfordshire	TL0842
Beech Bottom Dyke, St Albans	TL152088
Berkshire	SUS483
Berrygrove Woods, W of Aldenham	TQ130984
Berwickshire	NT7153
Blackfan Wood	TL308071
Blagrove Common H&MWTR	TL328337
Blaydon	NZ16
Bowsey Hill	SU807796
Box Hill	TQ174520
Brandon Hill, Bristol	ST5772

217 rows.

Writing queries

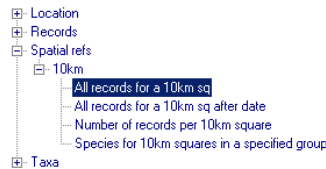
At present, queries must be developed using Access 97. They will be made available through the SQL Report tool only if they are SELECT queries which follow a particular naming convention.

It is very important to document a query so that it is clear to a user what it does and in what circumstances it can be used. This is done by writing a description in the query's properties.

Naming convention

Queries should be named "*|category|name*", e.g. "*|Location|Alphabetic list of locations*". Note the use of the vertical bar character (ASCII character 124 - the character above backslash near the bottom left corner of a standard British keyboard) as the delimiter.

The available queries are shown in the tree-view on the left-hand panel of the screen. The "*category*" part of the name forms the top levels of the hierarchy with the rest of the query name shown as children. You can have more than one level of category in the name. For example "*|Spatial refs|10km|All records for a 10km sq*" will result in a hierarchy with three levels:



Each unique category (case-insensitive) found amongst the queries contained in nbndata.mdb will appear as a parent node in the tree-view.

Notes:

- The addin only considers SELECT type queries (which can include GROUP BY and crosstab constructs for statistics and summaries).
- It only considers queries whose name starts with a vertical bar, "|" (or an underscore "_" - see below).
- Access limits query a name to 64 characters and does not allow it to have leading spaces or to include some special characters (square brackets "[]", full stop ".", exclamation mark "!", accent grave "`", or control characters ASCII 0-31).

Recommendations:

- If a query will prompt the user, this can be indicated to the user by including the parameter in brackets. e.g. "*|Location|Alphabetic list of locations (entry date on or after)*"
- If a query needs to obtain a list of keys, this also should be indicated in the name, e.g. "*|Records|All records from specified Locations(s)*"

Parameters

Simple parameters are included in a query in the usual way by including a term which Access cannot identify as a table or field name. e.g. "... WHERE ENTRY_DATE >= [Entered on or after date]"

In this case, the user will be prompted to enter the date to use when the query is run, and the phrase "Entered on or after date" will be used as the caption for the prompt that will appear.

Alphabetic list of locations entered on or after date

Description: Alphabetic list of the preferred names of Locations

Parameters: You need to enter a parameter.

Entered on or after date: date

Run the Query

Consequently, two factors should be borne in mind when designing queries:

- You cannot use the name of a table or field in *Recorder 2002* as the name of a parameter - it must be something that Access cannot recognise as the name of an object it knows about. In practice this should not cause any difficulty if you type a phrase containing two or more words separated by spaces - such a phrase will never conflict with an existing object name.

- Choose the parameter name so that it will be helpful to the user when it appears as the label for the prompt.

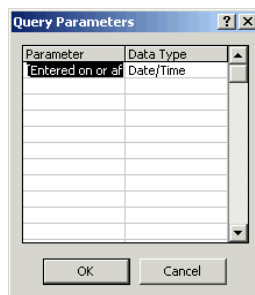
Use of LIKE

It is good practice to use "LIKE" as the comparison operator for text type parameters. For example, when designing the query, if you use the condition "LOCATION_NAME = [Enter location name]" then the user must type the name of the location on which they wish to report exactly as it is recorded (except that SQL conditions are not case-sensitive). For example, if "Blog's Wood" was the name of a Location in the database, then the user must type it formatted exactly in this way. Typing "Blogs Wood", without the apostrophe would not work.

However, if you use the condition "LOCATION_NAME like [Enter location name]" then the user can type a partially matching string which includes Access wildcard characters (e.g. "?" to stand for a single character and "*" to stand for any number of characters). If the condition is framed in this way then the user could respond "blog* wood" to find the desired location whether it had been entered as "Blog's Wood" or "Blogs Wood".

Declaring parameters explicitly

If the parameter needs to be replaced by a date or a number, or is of "Yes/No" type (e.g. "Include subsites?") and in some other circumstances detailed below, you need to explicitly declare the parameter and its type. To do this, in the Access query designer, select Query - Parameters ... from the menu, or right-click in the upper part of the design screen and choose Parameters ... from the drop-down menu.



Enter the name of the parameter on the left (best done by copying and pasting to make sure you get it exactly as you have written in the query) and select the type from the drop-down box on the right. When you press [OK] you will find that Access has added a "PARAMETER" clause at the start of your query:

```
PARAMETERS [Entered on or after date] DateTime;
SELECT ...
```

If you don't set the type of Yes/No, Date/Time or numeric type queries, then the parameter value will not be formatted correctly and the user will see an error when the query is run reporting that the parameter type is incorrect.

Using lists of keys from a rucksack in parameters

Parameters that will be replaced by a set of item keys supplied from a rucksack list in *Recorder* must adhere to the following conventions:

- The parameter name MUST be "[Rucksack/list name]" where *list name* is one of those listed in the table below. The parameter name is not case sensitive (so "[Rucksack/Location]" or "[rucksack/location]" are both acceptable)
- The parameter must be applied to an appropriate key field. The appropriate fields are shown in the table below.
- The form of the WHERE clause MUST be "key IN ([parameter])" e.g. "LOCATION_KEY IN ([Rucksack/Location])"

List name	Appropriate key
Taxon	TAXON_LIST_ITEM_KEY
Biotope	BIOTOPE_LIST_ITEM_KEY
Location	LOCATION_KEY
Name	NAME_KEY
Document	SOURCE_KEY

Dealing with confidential records

Any query which includes the TAXON_OCCURRENCE or BIOTOPE_OCCURRENCE table should take account of the fact that rows in these tables may be flagged as confidential. These rows should only be accessible to the "System Manager" or "Full Edit" users. Other users should not be able to see such rows. This is achieved by applying the following condition to the CONFIDENTIAL field in such rows: "[Conf_True] Or 0". In the Access Query designer, such a field will look as follows in the design grid:

CONFIDENTIAL
TAXON_OCCURRE
<input type="checkbox"/>
[Conf_True] Or 0

If this parameter needs to be declared explicitly (i.e. in a crosstab or base query), then it should be declared to be of type "Integer".

Crosstab queries

SQL Reports can execute crosstab queries. All parameters used in crosstab queries must be declared explicitly (this is generally true in Access, not just for this tool).

Queries on queries

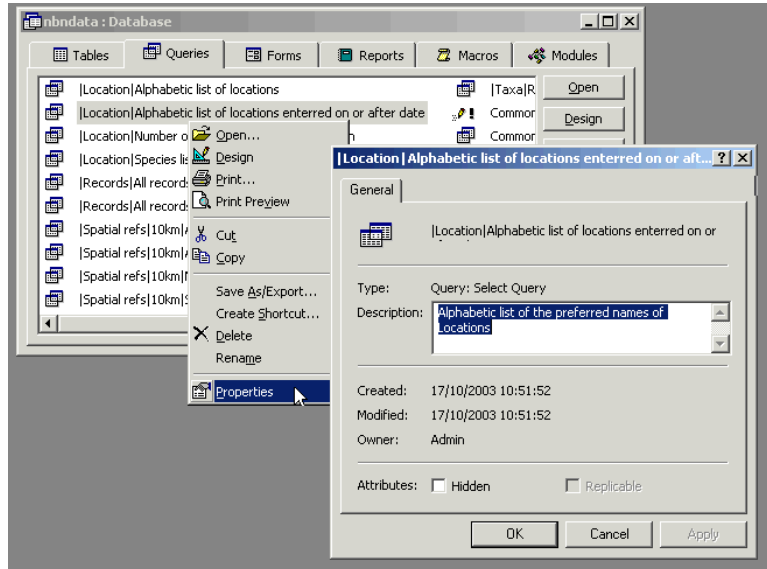
It is not always possible to achieve what you want in a single query. In some cases it is necessary for one query to work on the results of another. For example, suppose you want to count the species recorded from a Location. This requires an initial query which produces a list of the unique species names found in records associated with the Location in question (using a DISTINCT or GROUP BY query). Then a second query would be needed which counts the number of unique names that were found by the first query. It may occasionally be necessary to chain together more than two queries in this way.

Such queries on queries (to any length of chain) can be executed by SQL Reports, but there are some extra points which need to be born in mind:

- **Naming convention.** You probably don't want the "base query" (or several base queries in a longer chain) to be visible to the end user, however you do want SQL Reports to import it. Both these objectives can be achieved by naming the base query(-ies) starting with another special character - the underscore "_". Therefore, in the example given above, the first query could be named "_BaseSpeciesForLocation". Such a query is not displayed to the user, but SQL Report will import SELECT type queries whose names start with "_".
- **Parameters.** All parameters used in a base query MUST be explicitly declared (this is generally true in Access, not just for this tool).
- **Using lists of keys from a rucksack as parameters.** You CANNOT use SQL Report's "Rucksack/list" style parameters in a base query, but they can be used in the final query in the chain (i.e. the one the end user sees). So, for example, if you wanted to count only those species recorded from a location name that are amongst the Taxon list from a rucksack, then the base query could be designed to find the unique TAXON_LIST_ITEM_KEYS from the records associated with the location. The second query would then apply a "Rucksack/list" style parameter by having the condition "TAXON_LIST_ITEM_KEY in ([Rucksack/Taxon])". It would only count the species meeting this condition.

Documenting your queries

Write a description in the query's properties. This can be accessed from the Database Query tab in Access by right-clicking on a query name and selecting Properties... from the drop-down menu, or in the Query designer, by selecting View - Properties ... from the menu, clicking the properties toolbar button or by right-clicking in the upper half of the design screen and selecting Properties... from the drop-down menu.



The Description you type here will appear in the right-hand pane of the SQL Report window when the query is selected (see "Example" above). This is actually an HTML viewer, so you can include HTML tags if in your text if you feel it would help the user. For example, you could use HTML tags to italicise or embolden words, include a bulleted list or display an important point in a different colour. Access limits this text to 250 characters.

The user will need to know what the query is intended to do and any special considerations that should be born in mind. For example, if the query was intended to produce a list of species for a given vice-county, you might want to remind the user that only those species records which are linked to Locations which have the vice-county listed amongst their admin-areas will be included.

Distributing queries

SQL Report has facilities to import new queries (and update existing ones) from an MDB file providing they are SELECT type queries and follow SQL Reports' naming conventions (i.e. their name starts with special characters "|" or "_"). Therefore, all that is necessary to distribute queries to users is to send them an MBD file containing the queries you have designed and instruct them to import the queries into their database using SQL Reports.

The simplest way to make such a file is:

1. Open Access 97,
2. Create a new database by selecting File - New Database from the menu and select "Blank Database" in the "New" dialog which will appear, then click [OK] and give the new file a name,
3. Select File - Get External Data - Import from the menu,
4. Open the database in which you designed the queries and select them (not forgetting any base queries you might have used),
5. Press [OK] to import the selected queries into the new database file.

An MDB file which only contains queries will be small (a few hundred kilobytes at most) and can easily be sent as an email attachment.

Document history

Author	Date	Changes
Stuart Ball	13/10/2003	Original version
	17/10/2003	Change to tree view to list queries Handling confidential records
	4/12/2003	Updated to cover issues relating to queries on queries and crosstabs