Recorder 6

Adding simple filters to the main windows in *Recorder 6*

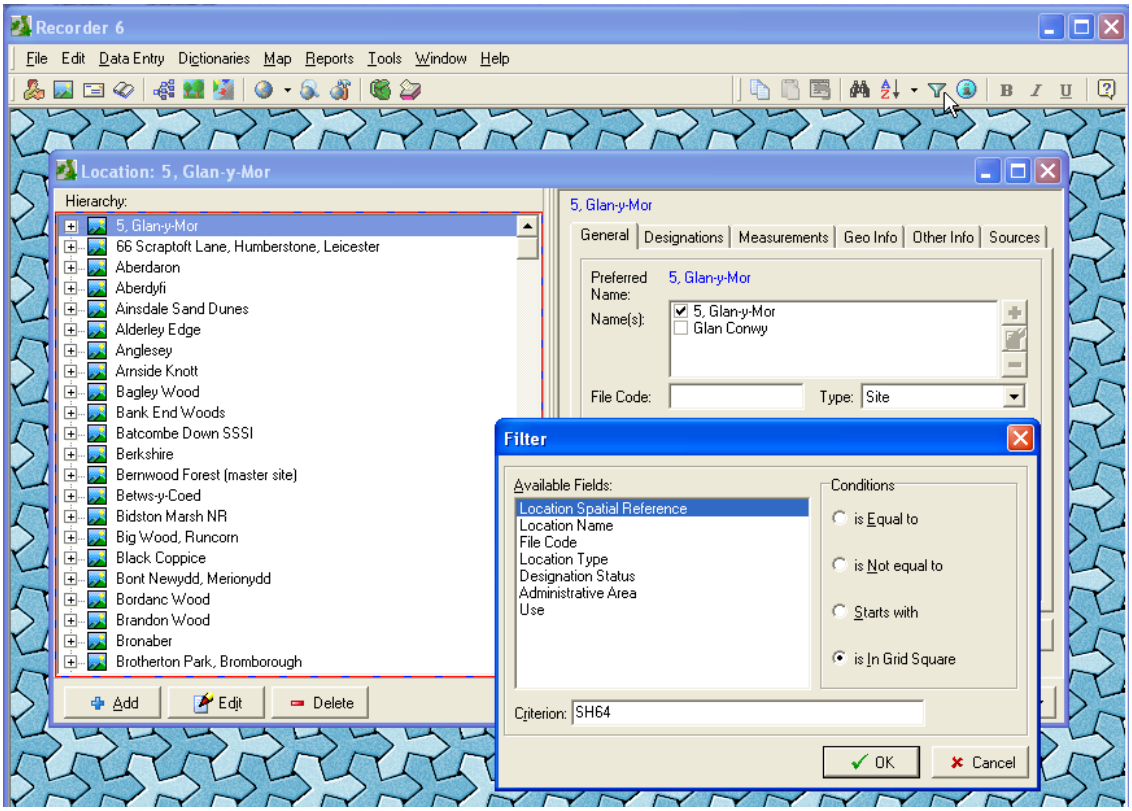
JNCC, August 2008

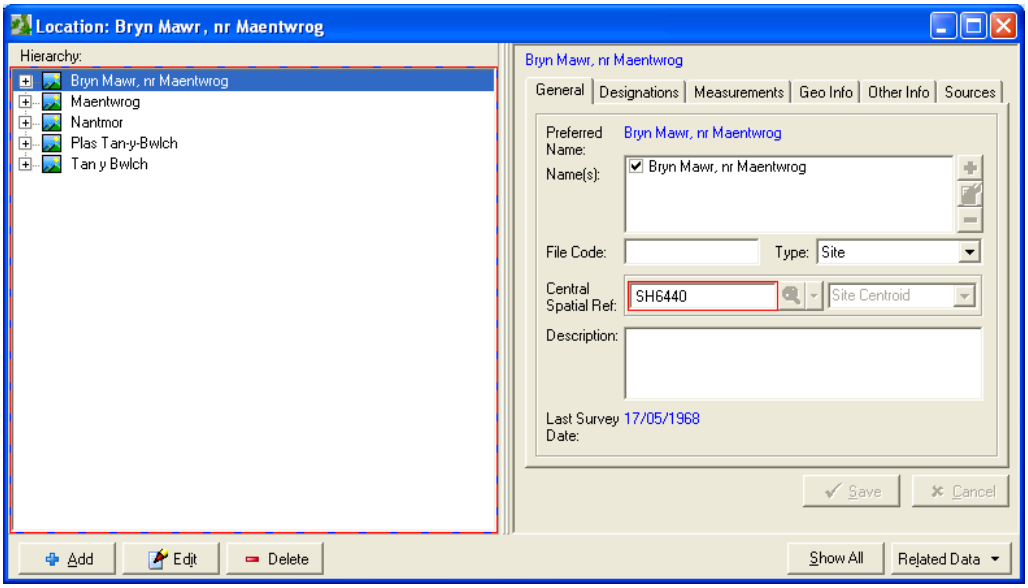
Contents

1 Introduction

This guide is aimed at all skill levels.

The simple filter facility on the main windows in *Recorder 6* allows the user to display a subset of rows according to some criterion they supply. In this example, Locations are filtered so that only those in grid square SH64 are displayed:





When a filter is applied, the Location window has a [Show All] button which will remove the filter and display all the Locations again.

Filters are available for each of the main windows:

- Location
- Names and addresses
- References
- Observations
- Taxon dictionary
- Biotope dictionary
- Admin areas dictionary

When one of these windows is open and active, the filter facility is invoked by:

- Selecting Edit – Simple Filter… from the menu
- Clicking the button in the toolbar

The filter is removed again by the [Show All] button (or by closing the window).

Only one filter can be applied at a time – i.e. if you already have applied a filter and then invoke the filter facility again, it will work on the whole table NOT on the subset selected by the first filter.

2 How is the definition of a simple filter stored?

The definitions for simple filters are stored in the USABLE_FIELD table in the database. This table also stores definitions for the Report Wizard. Simple filters are identified by an APPLY_TO code of 'F' (for Filter).

Field	Example	Explanation
USABLE_FIELD_KEY	ABABABAB00000053	16 character running key
TABLE_NAME	LOCATION	The name of the table to which this filter applies
FIELD_NAME	SPATIAL_REF	The name of the field to which this filter applies
FIELD_DESCRIPTION	Location Spatial Reference	The name that appears in the Filter dialog and by which the user identifies this filter
FIELD_TYPE	SPATIAL_REF	Tells the Filter dialog what type of information is stored in this field and therefore what options to offer to the user and how to treat the criterion they enter

APPLY_TO	F	Tells Recorder that this is the definition of a simple filter
SELECTABLE	0	These are used for Report Wizard fields and are not relevant to simple filters.
SORTABLE	0	
FILTERABLE	0	
CALCULATION_SQL	SELECT Location_Key AS ItemKey FROM Location WHERE Spatial_Ref	The incomplete SQL command that will be run to actually perform the filter. It is completed by appending a condition according to the information given by the user in the Filter dialog.

2.1 How it works

- The user opens one of the main windows and selects an item (e.g. opens the Observation window and selects a survey event). They then invoke the Simple filter facility to clicking the toolbar button or selecting the menu item.
- Recorder runs a query on the USABLE_FIELD table for entries where APLLY_TO= "F" and TABLE_NAME = "SURVEY_EVENT" in this example. The appropriate table to look for depends on the active window and the item that is currently selected.
- If rows are found, the Filter dialog is displayed with the entries from FIELD_DESCRIPTION shown in the Available Fields list.
- The user selects one of these Available Fields. The Filter dialog displays controls in its Conditions panel suitable for the FIELD_TYPE of the selected item.
- The user types an entry in the Criterion box and presses [OK].
- The Filter dialog generates an SQL statement by appending conditions to the CALCULATION_SQL string generated from the Criterion typed by the user and the option selected in the Conditions panel.
- This SQL statement is executed and the rows identified by the ItemKey column of the result set are displayed in the window.

2.2 Details

2.2.1 Usuable_Field_Key

This is the normal Recorder database key, i.e. A 16 –character running code where the first 8 characters are the SiteID and the other 8 characters are a running code which together provide a unique identifier within the table. If my SiteID is "JNCCDEV1" and I want to make new entries in this table, I could use keys "JNCCDEV100000001", "JNCCDEV100000002", etc.

2.2.2 Table_Name

The following table names are valid for simple filters and depend on the main window for which you are defining a filter:

Table name	Window to which the filter applies
ADMIN_AREA	Admin area dictionary browser
BIOTOPE	Biotope dictionary browser
BIOTOPE_OCCURRENCE	Observations window
INDIVIDUAL	Names and addresses window
LOCATION	Location window
ORGANISATION	Names and addresses window
REFERENCE	Documents window
SAMPLE	Observations window
SURVEY	Observations window
SURVEY_EVENT	Observations window
TAXON	Species dictionary browser
TAXON_OCCURRENCE	Observations window

2.2.3 Field_Name

The name of a field from the table named above.

Spatial references and vague dates are stored by Recorder in a group of fields.

- For a spatial reference, give the name of the field that stores the original string entered by the user (usually SPATIAL_REF).
- For a vague date, give the name of the field that stores the start data (usually VAGUE_DATE_START).

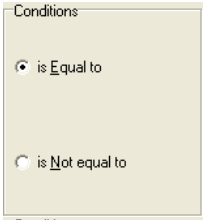
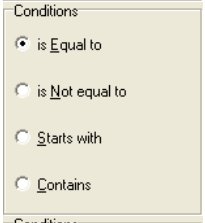
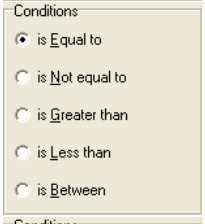
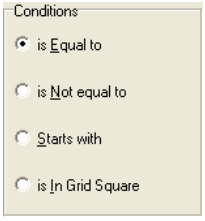
2.2.4 Field_Description

This is the name that will appear in the Filter dialog box when the user wants to apply a filter. It should be something descriptive which will allows the user to identify the filter and what it is intended to do. The maximum string length allowed is 70 characters (but this is too wide to fit in the dialog box. About 50 characters is the most that will fit.)

2.2.5 Field_Type

The possibilities are: BOOLEAN, MEMO, NUMBER, RICH_TEXT, SPATIAL_REF, TEXT or VAGUE_DATE

These determine the options displayed in the Conditions panel in the Filter dialog and, hence, how the Criterion typed by the user is turned into a condition to append to the CALCULATION_SQL string. The following table shows the options available for each of these types:

BOOLEAN	
MEMO, RICH_TEXT, TEXT	
NUMBER, VAGUE_DATE	
SPATIAL_REF	

2.2.6 Calculation_SQL

This string contains the incomplete SQL statement that will be executed to actually apply the filter. It is completed by the Filter dialog which appends a condition based on the Conditions option selected and the Criterion typed by the user. Therefore it MUST end with name of the field to which

the condition is to be applied as part of the WHRE clause. The keys of the items which will be displayed in the window as the result of the filter are obtained from the column called “ItemKey”, therefore it MUST have a result field which consists of the primary key of the relevant table and which has the alias “ItemKey”.

The format of the CALCULATION_SQL is therefore:

```
SELECT <key> as ItemKey FROM <target table> [join clause(s)]
WHERE [other conditions,] <target field>
```

Where:

- <target table> the name of the table we are filterring i.e. one of the tables listed under TABLE_NAME above
- <key> the name of the primary key field of the <target table>
- <target field> the name of the field to which the filtering condition is applied (as named in FIELD_NAME above)
- [] other elements that may or may not be needed depending on what you are rying to achieve

Examples:

```
SELECT Taxon_Occurrence_Key AS ItemKey FROM Taxon_Occurrence
WHERE ZERO_ABUNDANCE
```

This is a very simple example to apply a BOOLEAN type filter to taxon occurrences in the Observation window.

```
SELECT Survey_Event_Key AS ItemKey FROM Survey_Event SE INNER
JOIN Location_Name LN ON SE.Location_Key = LN.Location_Key
WHERE LN.Preferred = True AND Item_Name
```

This is a more complicated case to filter SURVEY_EVENT by Location name. This involves joining to the LOCATION_NAME table and an additional condition to restrict selections to the PREFERRED Location name.