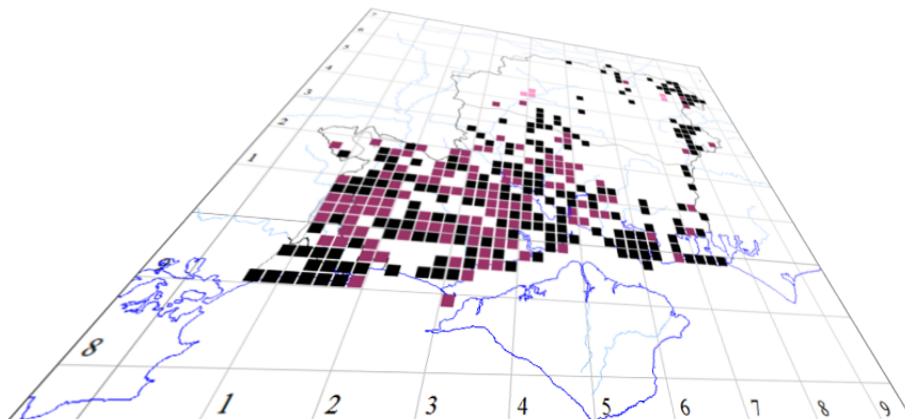


Appendix B

Useful Queries



The author of this handbook is pleased to have errors pointed out and to receive suggestions for improvement and other comments.

Please send all communications to:

vcirecorder@hantsplants.org.uk

Introduction

As I mentioned in Chapter 4, if you feel unable to write your own customised queries for reporting or filters, there are several places you can look for additional SQL code. Even if you don't find exactly what you want, you may be able to see that a very simple modification is all you need.

I would recommend first looking in Martin Harvey's excellent web site at <https://sites.google.com/site/kitenetter/Home/mapmate/sql>. As well as many reporting tools, you will also find some simple procedures for bulk amendment of Records with an incorrect feature such as a site or a taxon. Note the advice attached to these about taking a backup before you apply them. There are also some very useful data validation routines here.

Another source of information is the 'Files' section of the MM-Users Yahoo! group: <http://tech.groups.yahoo.com/group/MM-Users/files/>. This requires a little hunting through to sort out the items of abiding interest from the ephemeral. It's also worth looking in the message board of the group as well, as many SQL queries supplied in response to questions are simply provided in the message and not posted to /files.

The following sections of this Appendix provide some extra queries not, as far as I can see, covered in either of those sites.

Finally, you can try posting a "How do I...?" question on the MM-Users group. It's very likely that you will get expert help.

None of these sources gives any warranties as to fitness for purpose, so please read any accompanying documentation and make sure that the query purports to do what you want. Any "reporting only" query is at least safe; but take a backup of your MapMate database before running any query that modifies data.



A number of the code examples as they appear in this Acrobat PDF file will not run if simply pasted from here to MapMate's SQL editing box. This is because the document conversion process has inserted character codes for new lines into places in the code, so that MapMate cannot interpret it properly. You can overcome this either by editing out the spurious new lines by hand in the places where MapMate tells you there is an error, or by downloading the 'plain text' version of this Appendix from the web site. The latter is probably easier, but you will just need to be careful to grab the right bit of the text to copy and paste, as the visual clues are not as obvious.

APPENDIX B USEFUL QUERIES

Records

Query: Browse all Hybrid Records

Notes: Returns a list of all hybrids, including intergeneric hybrids, in taxon name order.

Code:

```
SELECT Records.[_guk], [Taxa\Default].Taxon,
[Taxa\Default].Vernacular, [Sites\Default].Name AS Site,
[Sites\Default].OSGridRef AS Gridref,
IIf([Sites\Default].ViceCounty>200,'H' &
[Sites\Default].ViceCounty-200,[Sites\Default].ViceCounty) AS
[Vice County], Records.Quantity & IIf([*Sex]<>'u',' ' &
[*Sex],'') AS Quantity, IIf([Records].DateTo-
[Records].Date<27,Format([Records].Date,'dd mmm
yyyy'),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,'yyyy'),Format([Records]
.Date,'mmm yyyy')) AS [Date], Recorders.Name AS Recorder,
Methods.Method, TaxonStage.Stage, RecordStatus.Status,
Records.Comment
FROM (((((Records INNER JOIN [Taxa\Default] ON Records.[*Taxon]
= [Taxa\Default].[_guk]) INNER JOIN [Sites\Default] ON Records.
[*Site] = [Sites\Default].[_guk]) INNER JOIN Methods ON Records.
[*Method] = Methods.[_guk]) INNER JOIN Recorders ON Records.
[*Recorder] = Recorders.[_guk]) INNER JOIN TaxonStage ON
Records.[*Stage] = TaxonStage.[_guk]) INNER JOIN RecordStatus ON
Records.[*Status] = RecordStatus.[_guk]
WHERE ([Taxa\Default].Taxon LIKE 'X *') OR ([Taxa\Default].Taxon
LIKE '* x *'))
ORDER BY [Taxa\Default].Taxon;
```

APPENDIX B USEFUL QUERIES

Query: Browse all Records for a <date>

Notes: Returns a list of all records made on a specific date, in taxon name order.

Code:

```
PARAMETERS [Enter a date strictly in format 'dd-mm-yyyy'] Text;
SELECT Records.[_guk], [Taxa\Default].Taxon,
[Taxa\Default].Vernacular, [Sites\Default].Name AS Site,
[Sites\Default].OSGridRef AS Gridref,
IIf([Sites\Default].ViceCounty>200,'H' &
[Sites\Default].ViceCounty-200,[Sites\Default].ViceCounty) AS
[Vice County], Records.Quantity & IIf([*Sex]<>'u',' ' &
[*Sex],') AS Quantity, IIf([Records].DateTo-
[Records].Date<27,Format([Records].Date,'dd mmm
yyyy'),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,'yyyy'),Format([Records]
.Date,'mmm yyyy')) AS [Date], Recorders.Name AS Recorder,
Methods.Method, TaxonStage.Stage, RecordStatus.Status,
Records.Comment
FROM (((((Records INNER JOIN [Taxa\Default] ON Records.[*Taxon]
= [Taxa\Default].[_guk]) INNER JOIN [Sites\Default] ON Records.
[*Site] = [Sites\Default].[_guk]) INNER JOIN Methods ON Records.
[*Method] = Methods.[_guk]) INNER JOIN Recorders ON Records.
[*Recorder] = Recorders.[_guk]) INNER JOIN TaxonStage ON
Records.[*Stage] = TaxonStage.[_guk]) INNER JOIN RecordStatus ON
Records.[*Status] = RecordStatus.[_guk]
WHERE Format(Records.[Date],'dd-mm-yyyy') LIKE [Enter a date
strictly in format 'dd-mm-yyyy']
ORDER BY [Taxa\Default].Taxon;
```

APPENDIX B USEFUL QUERIES

Query: **Browse all Records for a <place> like**

Notes: Returns a list of all records for all sites which match the entered <place> in any part of the Site name, ordered by taxon name within site name.

Code:

```
SELECT Records.[_guk], [Taxa\Default].Taxon,
[Taxa\Default].Vernacular, [Sites\Default].Name AS Site,
[Sites\Default].OSGridRef AS Gridref,
IIf([Sites\Default].ViceCounty>200,'H' &
[Sites\Default].ViceCounty-200,[Sites\Default].ViceCounty) AS
[Vice County], IIf([Records].DateTo-
[Records].Date<27,Format([Records].Date,"dd mmm
yyyy"),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,"yyyy"),Format([Records]
.Date,"mmm yyyy")) AS [Date], Recorders.Name AS Recorder,
Recorders_1.Name AS Determiner, Records.Comment
FROM (((((Records INNER JOIN [Taxa\Default] ON Records.[*Taxon]
= [Taxa\Default].[_guk]) INNER JOIN [Sites\Default] ON Records.
[*Site] = [Sites\Default].[_guk]) INNER JOIN Methods ON Records.
[*Method] = Methods.[_guk]) INNER JOIN Recorders ON Records.
[*Recorder] = Recorders.[_guk]) INNER JOIN RecordStatus ON
Records.[*Status] = RecordStatus.[_guk]) INNER JOIN [References]
ON Records.[*Reference] = References.[_guk]) INNER JOIN
Recorders AS Recorders_1 ON Records.[*Identifier] = Recorders_1.
[_guk]
WHERE ((([Sites\Default].Name) Like "*"+[Site name]+"*"))
ORDER BY [Sites\Default].Name, [Taxa\Default].Taxon;
```

APPENDIX B USEFUL QUERIES

Query: **Browse all Records new from <centre > at last sync**

Notes: Returns a list of all records owned by the given CUK that were new or amended on the last sync generation, in taxon order.

Code:

```
SELECT Records.[_guk], [Taxa\Default].Taxon,
[Taxa\Default].Vernacular, [Sites\Default].Name AS Site,
[Sites\Default].OSGridRef AS Gridref,
IIf([Sites\Default].ViceCounty>200,'H' &
[Sites\Default].ViceCounty-200,[Sites\Default].ViceCounty) AS
[Vice County], Records.Quantity & IIf([*Sex]<>'u',' ' &
[*Sex],') AS Quantity, IIf([Records].DateTo-
[Records].Date<27,Format([Records].Date,'dd mmm
yyyy'),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,'yyyy'),Format([Records]
.Date,'mmm yyyy')) AS [Date], Recorders.Name AS Recorder,
Methods.Method, TaxonStage.Stage, RecordStatus.Status,
Records.Comment
FROM (((((Records INNER JOIN [Taxa\Default] ON Records.[*Taxon]
= [Taxa\Default].[_guk]) INNER JOIN [Sites\Default] ON Records.
[*Site] = [Sites\Default].[_guk]) INNER JOIN Methods ON Records.
[*Method] = Methods.[_guk]) INNER JOIN Recorders ON Records.
[*Recorder] = Recorders.[_guk]) INNER JOIN TaxonStage ON
Records.[*Stage] = TaxonStage.[_guk]) INNER JOIN RecordStatus ON
Records.[*Status] = RecordStatus.[_guk]
WHERE Records.[_gen]>=[$lastsyncgen] and Right([Records].
[_guk],3)=[Enter a Centre Unique Key - eg 1z1]
ORDER BY [Taxa\Default].Taxon;
```

APPENDIX B USEFUL QUERIES

Query: **Browse all Records since <year>**

Notes: Returns a list of all records in the entered year and all subsequent years, in taxon order.

Code:

```
SELECT Records.[_guk], [Taxa\Default].Taxon,
[Taxa\Default].Vernacular, [Sites\Default].Name AS Site,
[Sites\Default].OSGridRef AS Gridref,
IIf([Sites\Default].ViceCounty>200,'H' &
[Sites\Default].ViceCounty-200,[Sites\Default].ViceCounty) AS
[Vice County], Records.Quantity & IIf([*Sex]<>'u',' ' &
[*Sex],') AS Quantity, IIf([Records].DateTo-
[Records].Date<27,Format([Records].Date,'dd mmm
yyyy'),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,'yyyy'),Format([Records]
.Date,'mmm yyyy')) AS [Date], Recorders.Name AS Recorder,
Methods.Method, TaxonStage.Stage, RecordStatus.Status,
Records.Comment
FROM (((((Records INNER JOIN [Taxa\Default] ON Records.[*Taxon]
= [Taxa\Default].[_guk]) INNER JOIN [Sites\Default] ON Records.
[*Site] = [Sites\Default].[_guk]) INNER JOIN Methods ON Records.
[*Method] = Methods.[_guk]) INNER JOIN Recorders ON Records.
[*Recorder] = Recorders.[_guk]) INNER JOIN TaxonStage ON
Records.[*Stage] = TaxonStage.[_guk]) INNER JOIN RecordStatus ON
Records.[*Status] = RecordStatus.[_guk]
WHERE (Year([Date])>=[Enter a Year - eg 1999])
ORDER BY [Taxa\Default].Taxon;
```


APPENDIX B USEFUL QUERIES

Query: Browse all Archived Records

Notes: returns a list of all records that have been put into 'archive' status (i.e. are not to be synced to partners).

Code:

```
SELECT Records.[_guk], [Taxa\Default].Taxon,
[Taxa\Default].Vernacular, [Sites\Default].Name AS Site,
[Sites\Default].OSGridRef AS Gridref,
IIf([Sites\Default].ViceCounty>200,'H' &
[Sites\Default].ViceCounty-200,[Sites\Default].ViceCounty) AS
[Vice County], Records.Quantity & IIf([*Sex]<>'u',' ' &
[*Sex],') AS Quantity, IIf([Records].DateTo-
[Records].Date<27,Format([Records].Date,'dd mmm
yyyy'),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,'yyyy'),Format([Records]
.Date,'mmm yyyy')) AS [Date], Recorders.Name AS Recorder,
Methods.Method, TaxonStage.Stage, RecordStatus.Status,
Records.Comment
FROM (((((Records INNER JOIN [Taxa\Default] ON Records.[*Taxon]
= [Taxa\Default].[_guk]) INNER JOIN [Sites\Default] ON Records.
[*Site] = [Sites\Default].[_guk]) INNER JOIN Methods ON Records.
[*Method] = Methods.[_guk]) INNER JOIN Recorders ON Records.
[*Recorder] = Recorders.[_guk]) INNER JOIN TaxonStage ON
Records.[*Stage] = TaxonStage.[_guk]) INNER JOIN RecordStatus ON
Records.[*Status] = RecordStatus.[_guk]
WHERE Records.[_gen] = 1
ORDER BY [Taxa\Default].Taxon;
```

APPENDIX B USEFUL QUERIES

Query: **Show duplicated Records entered by <centre>**

Notes: Returns a list of all records owned by the given CUK where there is more than one record having the taxon, site, date, recorder, method and reference the same, in taxon order.

Code:

```
SELECT [Records].[_guk], [Taxa\Default].Taxon,
[Taxa\Default].Vernacular, [Sites\Default].Name & "(" &
[Sites\Default].OSGridRef & ")" AS Site,
Format([Records].Date,"dd mmm yyyy") AS [Date],
[Records].Quantity, Recorders.Name AS Recorder, Methods.Method,
References.Author, References.Year, References.Comment
FROM (((((Records INNER JOIN [Sites\Default] ON [Records].
[*Site] = [Sites\Default].[_guk]) INNER JOIN [Taxa\Default] ON
[Records].[*Taxon] = [Taxa\Default].[_guk]) INNER JOIN Recorders
ON [Records].[*Recorder] = Recorders.[_guk]) INNER JOIN Methods
ON [Records].[*Method] = Methods.[_guk]) INNER JOIN [References]
ON [Records].[*Reference] = References.[_guk]) INNER JOIN
[C\Find Dups at <centre>] ON ([Records].[*Taxon] = [C\Find Dups
at <centre>].[*Taxon]) AND ([Records].[*Site] = [C\Find Dups at
<centre>].[*Site]) AND ([Records].Date = [C\Find Dups at
<centre>].Date) AND ([Records].[*Method] = [C\Find Dups at
<centre>].[*Method])
ORDER BY [Taxa\Default].Taxon, [Sites\Default].Name,
[Records].Date, [Records].Quantity;
```

Requires: **Find Dups at <centre>**

Code:

```
SELECT Records.[*Taxon], Records.[*Site], Records.Date, Records.
[*Method], Records.[*Recorder], Records.[*Reference]
FROM (Records INNER JOIN [Taxa\Default] ON Records.[*Taxon] =
[Taxa\Default].[_guk]) INNER JOIN [Sites\Default] ON Records.
[*Site] = [Sites\Default].[_guk]
WHERE (((Right([Records].[_guk],3))=[Enter a centre unique key -
eg 1z1]))
GROUP BY Records.[*Taxon], Records.[*Site], Records.Date,
Records.[*Method], Records.[*Recorder], Records.[*Reference]
HAVING ((Count(Records.[*Taxon]))>1);
```

APPENDIX B USEFUL QUERIES

Query: Recording events for <year>

Notes: Lists “events” (defined as records made by one Recorder on one date) within the one given year. Prompts for a “threshold” value allowing events with small numbers of records to be left out.

Code:

```
SELECT First([Sites\Default].[2kSquare]) AS Tetrad,
IIf([Records].DateTo-[Records].Date<27,Format([Records].Date,'dd
mmm yyyy'),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,'yyyy'),Format([Records]
.Date,'mmm yyyy')) AS [Date], Recorders.[Name],
Count([Records].[_guk]) AS [Records Made]
FROM (Records INNER JOIN [Sites\Default] ON [Records].[*Site] =
[Sites\Default].[_guk]) INNER JOIN Recorders ON Records.
[*Recorder] = Recorders.[_guk]
WHERE Year([Records].[Date]) = [Enter a year (4 digits)]
GROUP BY Recorders.[Name], IIf([Records].DateTo-
[Records].Date<27,Format([Records].Date,'dd mmm
yyyy'),IIf([Records].DateTo-
[Records].Date>32,Format([Records].Date,'yyyy'),Format([Records]
.Date,'mmm yyyy'))
HAVING Count([Records].[_guk]) >= [Minimum no of records to
report]
ORDER BY First([Sites\Default].[2kSquare]);
```

APPENDIX B USEFUL QUERIES

Sites

Query: **Browse Sites entered by <centre>**

Notes: Returns a list of all sites created by the licensee, in name order.

Code:

```
PARAMETERS [Centre] Text;
SELECT [Sites\Default].[_guk], [Sites\Default].Name AS [Site
Name], [Sites\Default].OSGridRef AS Gridreference,
[Sites\Default].[10kSquare] AS [10km Square],
iif([Sites\Default].ViceCounty>200, 'H' &
[Sites\Default].ViceCounty-200, [Sites\Default].ViceCounty) AS
[Vice County], Habitats.Habitat, [Sites\Default].Description
FROM [Sites\Default] INNER JOIN Habitats ON [Sites\Default].
[*Habitat] = Habitats.[_guk]
WHERE [Sites\Default].[_guk] LIKE '*' & [Centre]
ORDER BY [Sites\Default].Name;
```

Query: **Browse Sites for a <10k square>**

Notes: Returns a list of all sites within the given hectad, in name order.

Code:

```
SELECT [Sites\Default].[_guk], [Sites\Default].Name AS [Site
Name], [Sites\Default].OSGridRef AS Gridreference,
[Sites\Default].[10kSquare] AS [10km Square],
iif([Sites\Default].ViceCounty>200, 'H' &
[Sites\Default].ViceCounty-200, [Sites\Default].ViceCounty) AS
[Vice County], Habitats.Habitat, [Sites\Default].Description
FROM [Sites\Default] INNER JOIN Habitats ON [Sites\Default].
[*Habitat] = Habitats.[_guk]
WHERE [Sites\Default].[10kSquare] = [Enter 10k square]
ORDER BY [Sites\Default].Name;
```

APPENDIX B USEFUL QUERIES

Query: **Browse Sites with name like <name>**

Notes: Returns a list of all sites which match the given name text in any part of their name. in name order.

Code:

```
PARAMETERS [Site Name] Text;
SELECT [Sites\Default].[_guk], [Sites\Default].Name AS [Site
Name], [Sites\Default].OSGridRef AS Gridreference,
[Sites\Default].[10kSquare] AS [10km Square],
iif([Sites\Default].ViceCounty>200, 'H' &
[Sites\Default].ViceCounty-200, [Sites\Default].ViceCounty) AS
[Vice County], Habitats.Habitat, [Sites\Default].Description
FROM [Sites\Default] INNER JOIN Habitats ON [Sites\Default].
[*Habitat] = Habitats.[_guk]
WHERE [Sites\Default].Name LIKE '*' & [Site Name] & '*'
ORDER BY [Sites\Default].Name;
```

APPENDIX B USEFUL QUERIES

Other Data

Query: **Browse my Filters**

Notes: Returns a list of all filters created by the licensee.

Code:

```
SELECT [_guk], [_filterClass], [_filterName], [_filterSQL]
FROM Filters
WHERE Right(Filters.[_guk],3)=[thiscuk] AND Filters.
[_filterName] <> "";
```

Query: **Sync History**

Notes: Returns the history of sync partner exchanges, 'from' and 'to' for each pair of partners listed separately, in order of CUK of sender and receiver.

Code:

```
SELECT Sync.[_guk], Sync.[_host] AS [From (CUK)], Sync.[_prime]
AS [From (Name)], Sync.[_target] AS [To (CUK)], Sync.[_address],
Sync.[_location], Sync.[_lastsync] AS [Last sync], Sync.
[_lastsyncon] AS [Last sync on], Sync.[_syncrecords] AS [No of
records], Sync.[_syncedits] AS [No of edits], Sync.[_syncs] AS
[No of syncs], Sync.[_ulf] AS [Site Filter], Sync.[_dlf] AS
[Taxon Filter], Sync.[_born] AS [Syncs started], Sync.[_gen]
FROM Sync
ORDER BY [_host], [_target];
```