

PlanePlotter Plane Symbols

Prepared by Lionel K Anderson MSc

When PlanePlotter is first installed it provides the user with a standard symbol to indicate the positions of aircraft when be displayed on a chart. The default symbol used to be a crude arrow shape but in the latest versions of PlanePlotter the shape of the standard symbol is defined in a special text file called *planesymbol.txt*. Users may notice that the PlanePlotter application directory contains a number of aircraft symbol bitmap images, one for each of the 36 ten degree points but these are associated solely with the Memory Map functions that few users use nowadays, and not with PlanePlotters own chart displays.



0 8 n Users can create their own library of symbols by studying the existing *planesymbol.txt* file that contains a
1 7 n list of vertices to display a different polygonal symbol for the aircraft. Each vertex is a pair of x y integer
1 3 n coordinates as shown to the left. When PlanePlotter starts it imports the symbol from the *planesymbol.txt* file
8 -4 and use it instead of the default aircraft symbol. Note that the aircraft symbol size parameter already specified
8 -6 in *Options, Chart, Options* will still apply to the customised shape.
1 -2

1 -6 PlanePlotter colour codes the nose of the aircraft to indicate ascent or descent. In order to identify the vertices
3 -8 that constitute the nose, an “n” must be added after a space following the Y coordinate of each vertex that is
3 -9 part of the nose.
0 -8

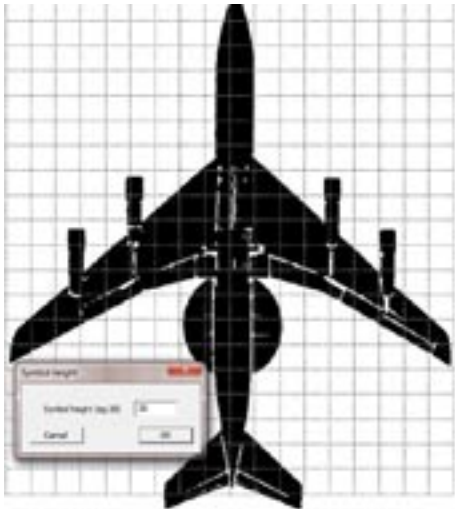
-3 -9 One can also make PlanePlotter display different symbols for different aircraft or different aircraft types. This is
-3 -8 achieved by creating one or more text files with names *planesymbol1.txt, planesymbol2.txt, planesymbol3.txt*
-1 -6 etc. Each file must follow the same convention as described above. The new symbols created in these files will
-1 -2 be used when a User tag for an aircraft includes the sequence \$1, \$2, \$3 etc. For example when an aircraft is
-8 -6 tagged with \$6 PlanePlotter will use the symbol contained in the file *planesymbol6.txt*. Tag numbers 1-9 & A-F
-8 -4 are supported however if no tag is used then the default symbol *planesymbol.txt* is used.
-1 3 n

-1 7 n
0 8 n An easy way of creating these is described on the next page.

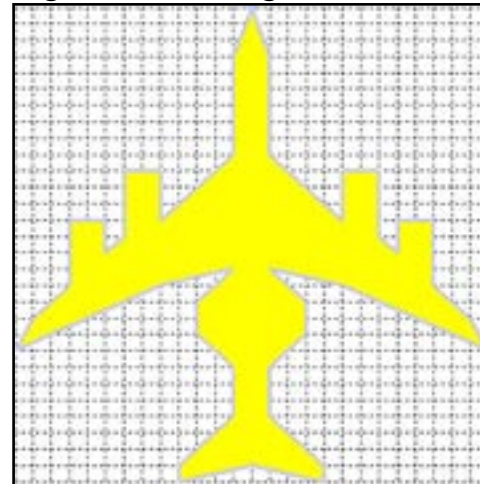
SymbolMaker2

There is now available a small utility program that enables users to create their own aircraft symbol from a silhouette image of an aircraft. All that is required to create a custom symbol is to run the SymbolMaker program from the Internet at <http://www.coaa.co.uk/SymbolMaker2.exe>. Silhouettes of aircraft are readily available from web sites such as http://www.au.af.mil/au/awc/awcgate/clip_af.htm. They may be downloaded and saved to a local folder but should be inverted if necessary to create a black image on a white background. Note Images should be a minimum of 750px wide and proportionally tall.

With SymbolMaker2 running select *File, Silhouette* to open the downloaded image, in this case it's an AWAC silhouette. The next step is to specify the size of the symbol in pixels, 20 or 25 is a good starting point. Then starting at the nose of the aircraft and working clockwise click points on the grid to create the symbol outline finishing back at the nose. At this stage the symbol will be filled with a yellow colour. Finally click once more at the nose and two or three points more to define the nose area, this will change the nose area colour to blue. Finally save the plane symbol to a file called planesymbolX.txt where X is a numeral 1-9 or a letter A-H. The plane symbol text file contains a list of the vertices defining the symbol. A letter "n" after a pair of number indicates that point will change colour if the aircraft is ascending or descending.



AWACS Silhouette



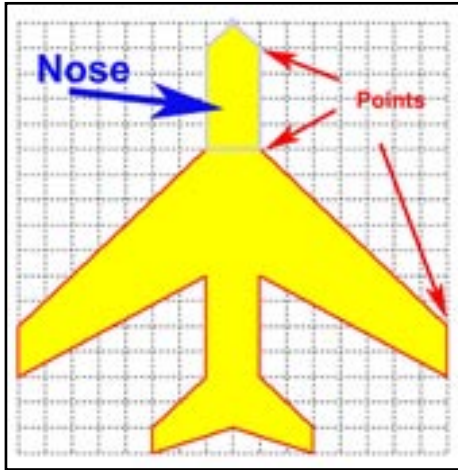
planesymbol9.txt

```
0 7 n      0 -9
0 10 n     3 -10
1 8 n      -3 -9
1 3        -1 -8
4 1        -1 -5
4 3        -2 -4
5 3        -2 -3
5 0        -1 -2
6 -1       -1 -1
6 1        -2 -1
7 1        -4 -2
7 -2       -9 -5
9 -4       -9 -4
9 -5       -7 -2
4 -2       -7 1
2 -1       -6 1
1 -1       -6 -1
1 -2       -5 0
2 -3       -5 3
2 -4       -4 3
1 -5       -4 1
1 -8       -1 3
3 -9       -1 8 n
3 -10     0 10 n
```

Examples

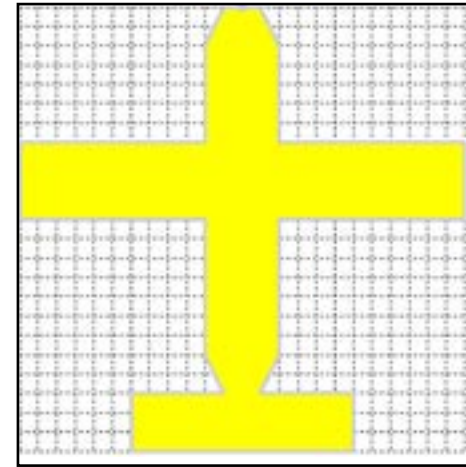
Some further examples are shown below. A Zip file, Planesymbol_files.zip, containing these can be downloaded from <http://groups.yahoo.com/group/planeplotter/files/> courtesy of 'John Satcom911'.

```
0 8 n
1 7 n
1 3 n
8 -4
8 -6
1 -2
1 -6
3 -8
3 -9
0 -8
-3 -9
-3 -8
-1 -6
-1 -2
-8 -6
-8 -4
-1 3 n
-1 7 n
0 8 n
```



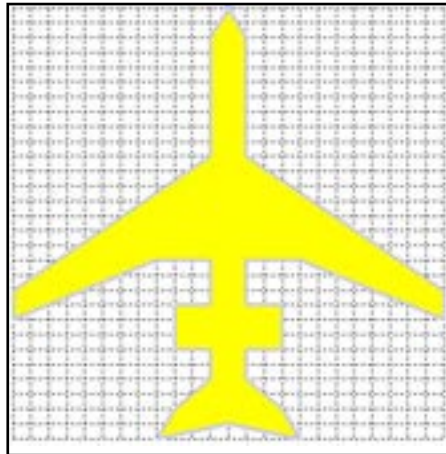
planesymbol0.txt
Default

```
0 13 n
1 13 n
2 11 n
2 6 n
12 6 n
12 2 n
2 2 n -12 6 n
2 -5 n -2 6 n
1 -7 n -2 11 n
6 -7 n -1 13 n
6 -10 n 0 13 n
-6 -10 n
-6 -7 n
-1 -7 n
-2 -5 n
-2 2 n
-12 2 n
```



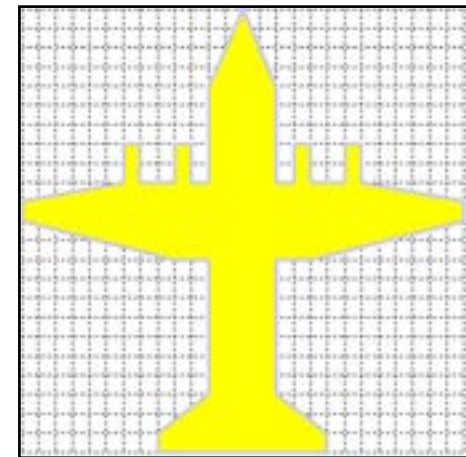
planesymbol1.txt
Light

```
0 14 n -1 -11
1 12 n n
1 4 n -1 -9 n
12 -5 n -3 -9 n
12 -7 n -3 -6 n
4 -3 n -1 -6 n
1 -3 n -1 -5 n
1 -5 n -1 -3 n
1 -6 n -4 -3 n
3 -6 n -12
3 -9 n -7 n
1 -9 n -12
1 -11 n -5 n
3 -13 n -1 4 n
4 -15 n -1 12
0 -14 n n
-4 -15 n 0 14 n
-3 -13 n
```



planesymbol2.txt
VC10

```
0 11 n
2 7 n -5 -12 n
2 2 n -5 -11 n
3 2 n -2 -9 n
3 4 n -2 -2 n
4 4 n -4 -2 n
4 2 n -13 0 n
6 2 n -13 1 n
6 4 n -7 2 n
7 4 n -7 4 n
7 2 n -6 4 n
13 1 n -6 2 n
13 0 n -4 2 n
4 -2 n -3 4 n
2 -2 n -3 2 n
2 -9 n -2 2 n
5 -11 n -2 2 n
5 -12 n -2 7 n
0 -12 n 0 11 n
```

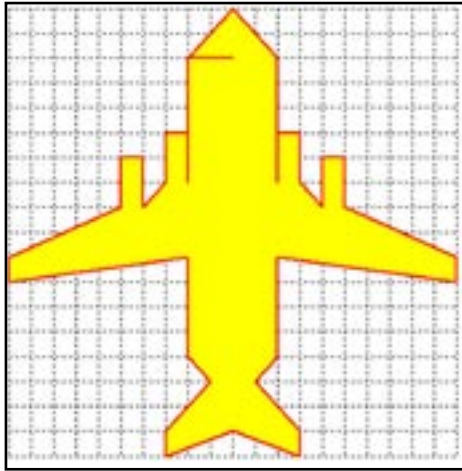


planesymbol3.txt
C130

If all the vertices have a letter “n” after them then the whole symbol will change colour as the aircraft ascends or descends and not just the nose.

```

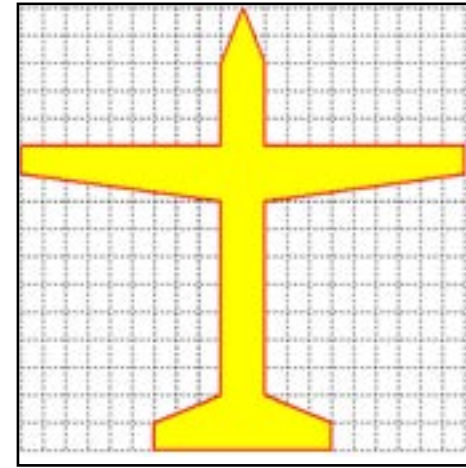
0 10
2 8
2 4
2 3
2 5
3 5
3 3
4 2
4 4
5 4
5 2
10 0
10 -1
2 0
2 -4
1 -5
3 -7
3 -8
0 -7
-3 -8
0 8
    
```



planesymbol4.txt
C5

```

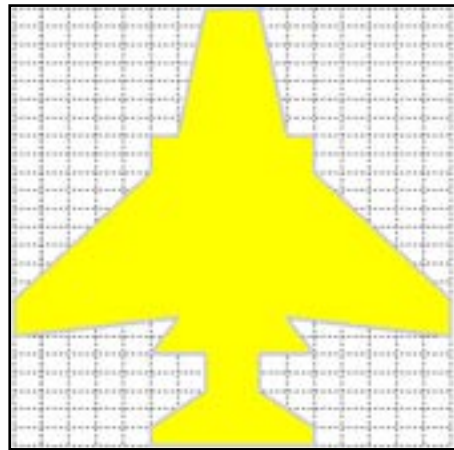
0 8
1 6
1 3
10 3
10 2
1 1
1 -6
4 -7
4 -8
-4 -8
-4 -7
-1 -6
-1 1
-10 2
-10 3
-1 3
-1 6
0 8
    
```



planesymbol5.txt
U2/Glider

```

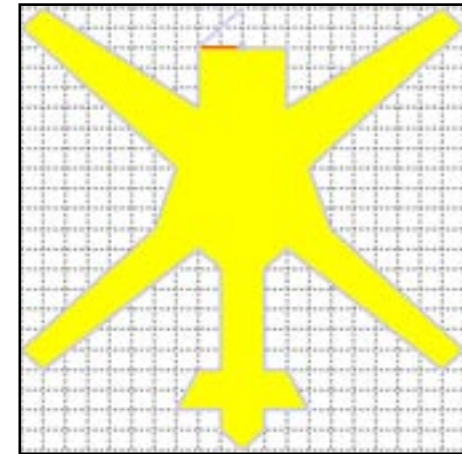
1 10 n
2 3 n
3 3 n
3 1 n
8 -6 n
8 -8 n
2 -7 n
3 -9 n
1 -9 n
1 -11 n
3 -13 n
3 -14 n
-3 -14 n
-3 -13 n
-1 -11 n
-1 -9 n
-3 -9 n
1 10 n
    
```



planesymbol6.txt
Fast Jet

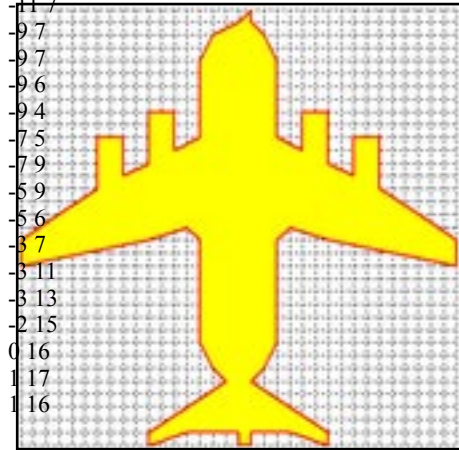
```

0 9 n
1 9
2 9 n
2 6 n
9 11 n
10 10 n
3 3 n
4 0 n
10 -6 n
9 -7 n
2 -1 n
1 -2 n
1 -7 n
2 -7 n
3 -9 n
1 -9 n
1 -10 n
0 -11 n
-1 -10 n
-1 -9 n
-3 -9 n
-1 -7 n
-1 -2 n
-2 -1 n
-9 -7 n
-10 -6 n
-4 0 n
-3 3 n
-10 10 n
-9 11 n
-2 6 n
-2 9 n
0 11 n
    
```

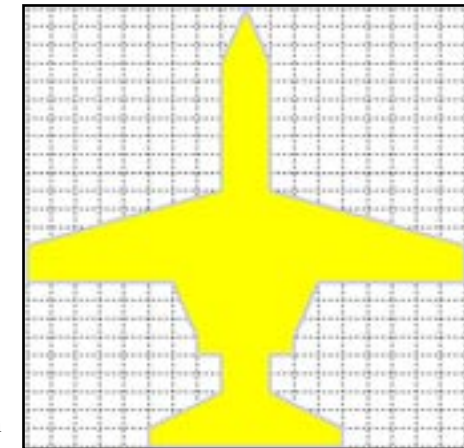


planesymbol7.txt
Helo

1 16
 2 15
 3 13
 3 11
 3 7
 5 6
 5 9
 7 9
 7 5
 9 4
 9 7
 11 7
 11 4
 11 3
 17 -1
 17 -3
 7 -1
 4 0
 3 -1
 3 -4
 3 -9



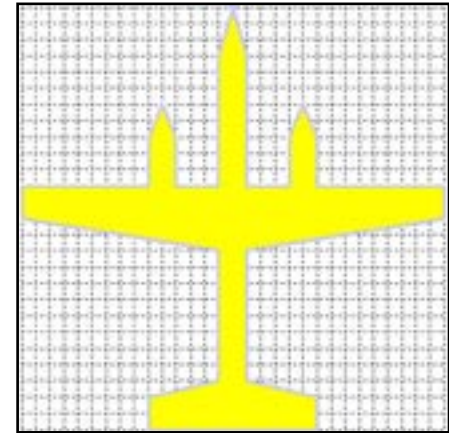
planesymbol8.txt
 C17



planesymbolB.txt
 Gulfstream

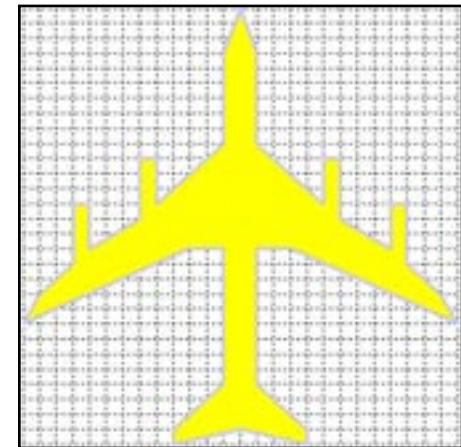
0 13 n
 1 10 n
 1 3 n
 9 0 n
 9 -2 n
 3 -2 n
 2 -5 n
 2 -6 n
 1 -6 n
 1 -8 n -2 -6 n
 4 -10 n -2 -5 n
 4 -11 n -3 -2 n
 0 -11 -9 -2 n
 -4 -11 n -9 0 n
 -4 -10 n -1 3 n
 -1 -8 n -1 10 n
 -1 -6 n 0 13 n

0 13 n
 1 10 n
 1 2 n
 4 2 n
 4 5 n
 5 7 n
 6 5 n -1 -2 n
 6 2 n -15 0 n
 15 2 n -15 2 n
 15 0 n -6 2 n
 1 -2 n -6 5 n
 1 -10 n -5 7 n
 6 -11 n -4 5 n
 6 -13 n -4 2 n
 -6 -13 n -1 2 n
 -6 -11 n -1 10 n
 -1 -10 n 0 13 n



planesymbolA.txt
 Airbus Baluga

0 14 n
 1 11 n -4 -15 n
 1 5 n -4 -14 n
 5 1 n -1 -11 n
 5 4 n -1 -2 n
 6 4 n -3 -2 n
 6 0 n -13 -7 n
 9 -2 n -12 -5 n
 9 1 n -10 -3 n
 10 1 n -10 1 n
 10 -3 n -9 1 n
 12 -5 n -9 -2 n
 13 -7 n -6 0 n
 3 -2 n -6 4 n
 1 -2 n -5 4 n
 1 -11 n -5 1 n
 4 -14 n -1 5 n
 4 -15 n -1 11 n
 0 -14 n 0 14 n



planesymbolC.txt
 K35R

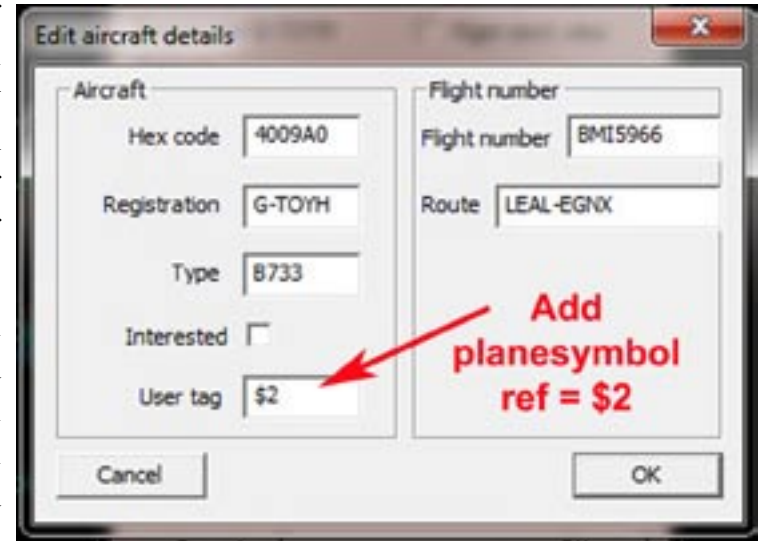
Associating a symbol with an aircraft

Once the planesymbolx.txt files have been created anyone of them can be associated with an individual aircraft an aircraft type, an airline or a country. By *right clicking* an aircraft symbol displayed on a PlanePlotter chart



Aircraft position report dialog

Aircraft position report dialog will be opened as shown here. By clicking the Edit button a second dialog box will be displayed and the user may then enter a value in the *User tag*. By using the dollar sign and a number or letter such as \$2 or \$A PlanePlotter will associate the newly created symbol from in these cases either *planesymbol2.txt* or *planesymbolA.txt*.



Aircraft position report dialog

Clicking OK will close the dialog and the aircraft symbol displayed will immediately be updated to the one specified in the text file.

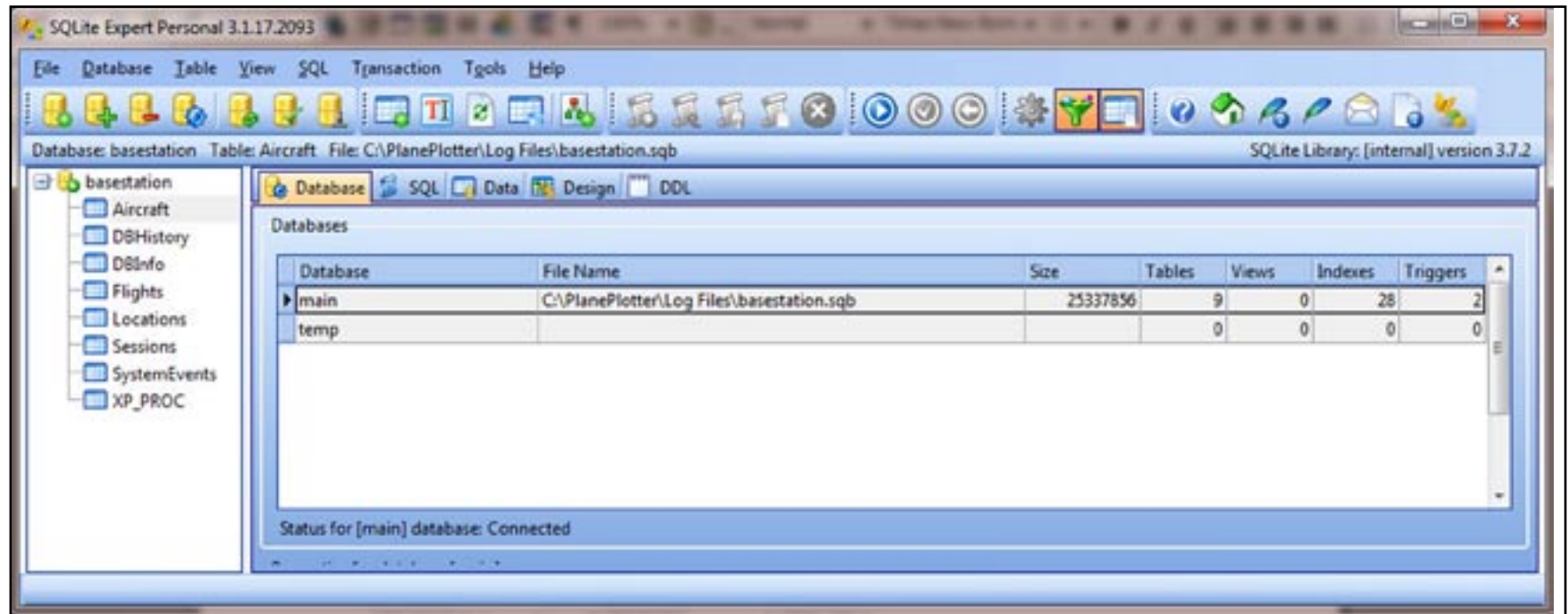
To ensure that association between the particular aircraft and chosen symbol is retained then user should confirm that the PlanePlotter Menu *Options, Mode-S, Kinetic, Update registrations* box is checked. Where the symbol needs to be associated with a number of aircraft this can be achieved by a bulk updating of the *User tags* in the basestation.sqb database. An explanation of how to do this is given in the following pages.

Editing the basestation.sqb database

Both the PlanePlotter databases basestation.sqb and flightroute.sqb use the SQLite relational database management system. Fortunately for PlanePlotter users there are a number of readily available database management tools that allow them to edit these databases without the need for a too-in-depth knowledge of SQLite.

The easiest tool to use is called *SQLite Expert Personal*. It can be downloaded from <http://www.sqliteexpert.com/SQLiteExpertPersSetup.exe> and as it is Freeware there is no cost nor expiration data.

After downloading, installing and then running the program the user will see a screen similar to that below. By then selecting *File, Open Database* the user can view the basestation.sqb database. First click on the *Aircraft* table indicator in the left panel and then click the *Data* button to display a table of all aircraft in the database. As the table contains 49 fields of data displayed as 49 columns it is necessary to scroll right and left to see all the data.

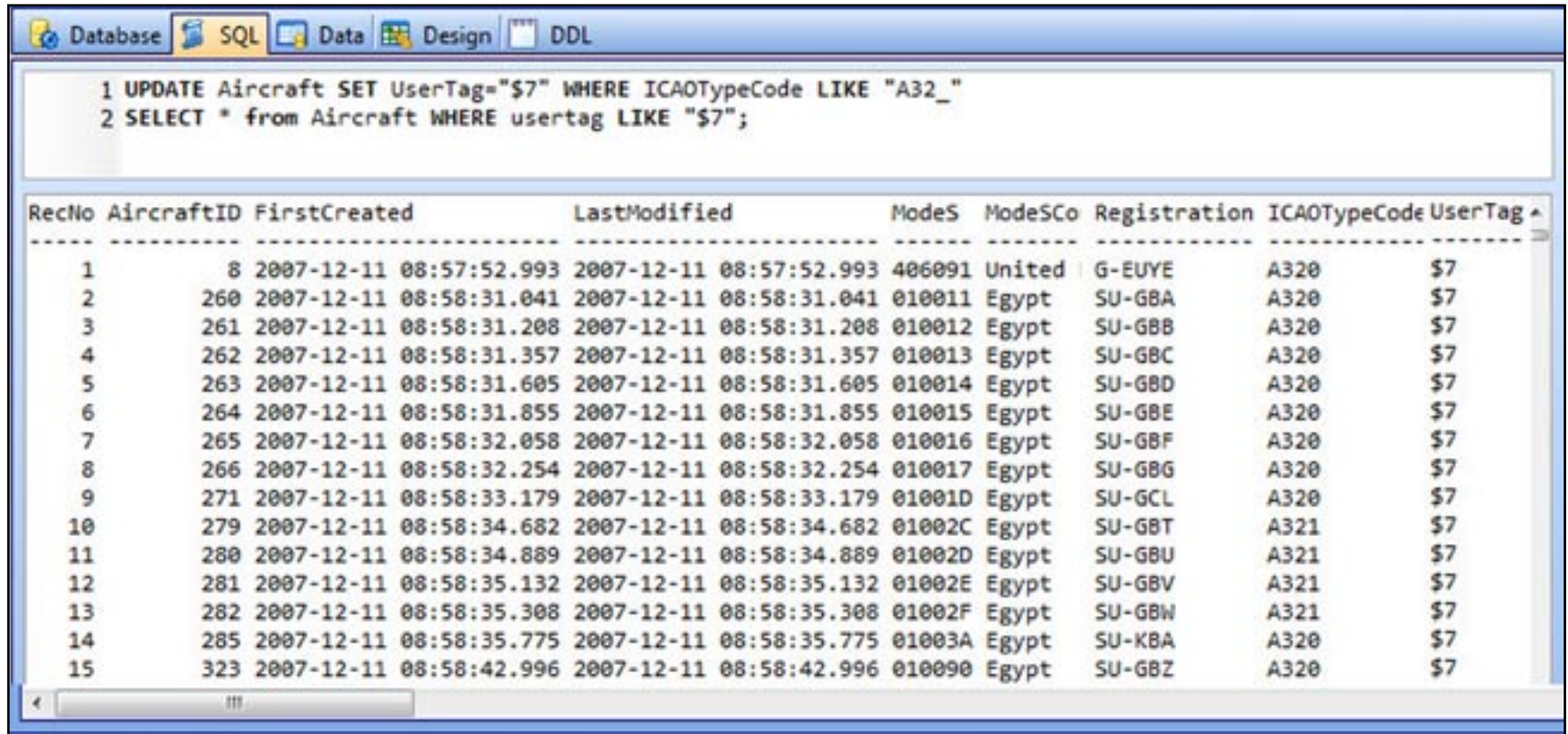


Associating a User Tag with type of Aircraft

In this first example we want to use the aircraft symbol in the planesymbol7.txt file with any Airbus in the A32x series. The first step is click the *SQL* button that opens a panel where some SQL queries may be entered, these are effectively commands to manipulate the contents of a table in the database. In this case we enter two queries:

```
UPDATE Aircraft SET UserTag="$7" WHERE ICAOTypeCode LIKE "A3%"  
SELECT * from Aircraft WHERE UserTag LIKE "$%";
```

The first is a command to update every entry in the 'Aircraft' table 'UserTag' field where the 'ICAOTypeCode' is A320 or A321. (The use of a '%' after the 'A3' permits any following characters).



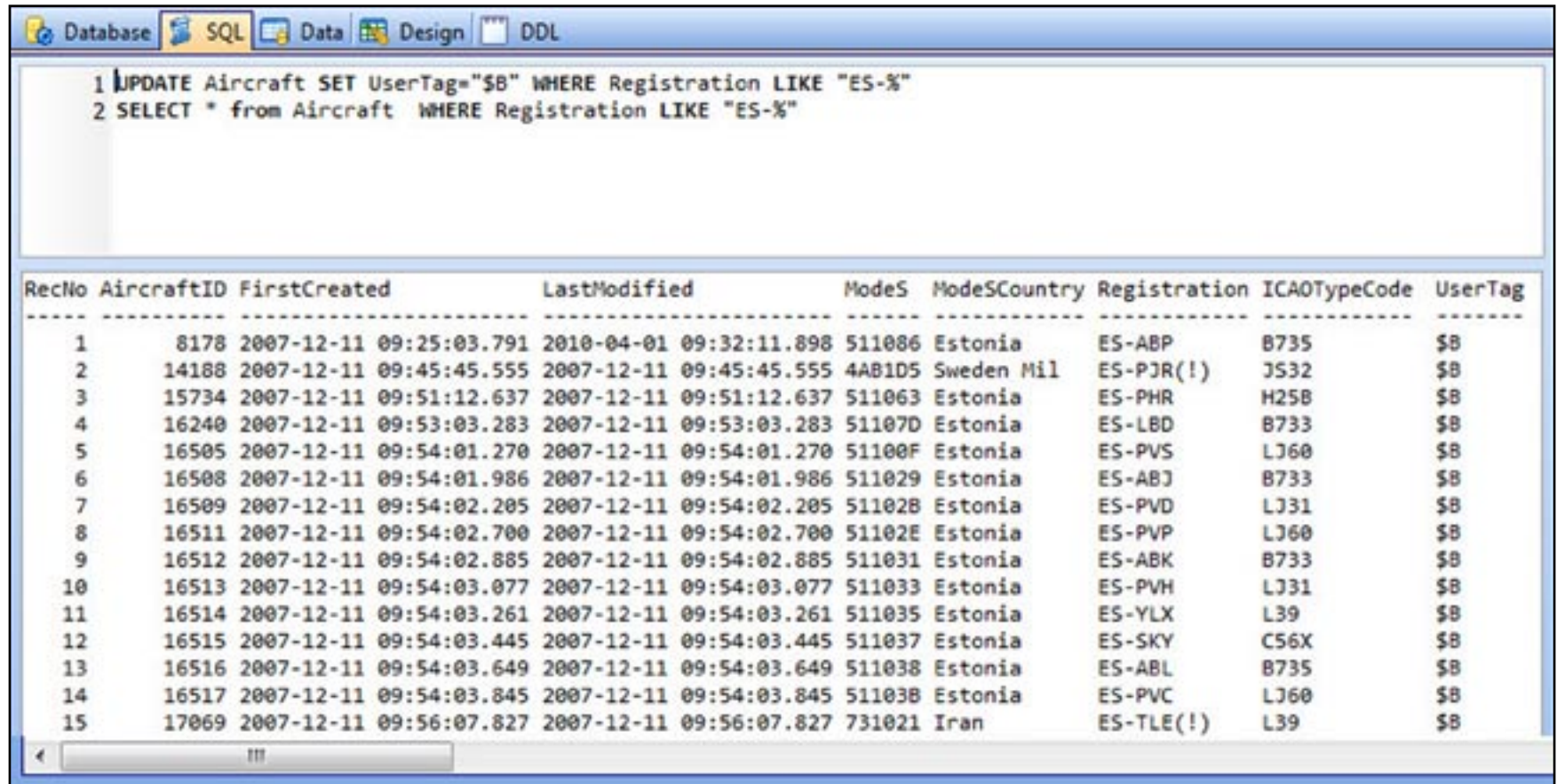
The screenshot shows a database management interface with a menu bar (Database, SQL, Data, Design, DDL) and a query editor containing two SQL statements. Below the editor is a table with 15 rows of aircraft data. The columns are: RecNo, AircraftID, FirstCreated, LastModified, ModeS, ModeSCo, Registration, ICAOTypeCode, and UserTag. The UserTag column contains '\$7' for all rows, indicating the result of the update query.

RecNo	AircraftID	FirstCreated	LastModified	ModeS	ModeSCo	Registration	ICAOTypeCode	UserTag
1	8	2007-12-11 08:57:52.993	2007-12-11 08:57:52.993	406091	United	G-EUYE	A320	\$7
2	260	2007-12-11 08:58:31.041	2007-12-11 08:58:31.041	010011	Egypt	SU-GBA	A320	\$7
3	261	2007-12-11 08:58:31.208	2007-12-11 08:58:31.208	010012	Egypt	SU-GBB	A320	\$7
4	262	2007-12-11 08:58:31.357	2007-12-11 08:58:31.357	010013	Egypt	SU-GBC	A320	\$7
5	263	2007-12-11 08:58:31.605	2007-12-11 08:58:31.605	010014	Egypt	SU-GBD	A320	\$7
6	264	2007-12-11 08:58:31.855	2007-12-11 08:58:31.855	010015	Egypt	SU-GBE	A320	\$7
7	265	2007-12-11 08:58:32.058	2007-12-11 08:58:32.058	010016	Egypt	SU-GBF	A320	\$7
8	266	2007-12-11 08:58:32.254	2007-12-11 08:58:32.254	010017	Egypt	SU-GBG	A320	\$7
9	271	2007-12-11 08:58:33.179	2007-12-11 08:58:33.179	01001D	Egypt	SU-GCL	A320	\$7
10	279	2007-12-11 08:58:34.682	2007-12-11 08:58:34.682	01002C	Egypt	SU-GBT	A321	\$7
11	280	2007-12-11 08:58:34.889	2007-12-11 08:58:34.889	01002D	Egypt	SU-GBU	A321	\$7
12	281	2007-12-11 08:58:35.132	2007-12-11 08:58:35.132	01002E	Egypt	SU-GBV	A321	\$7
13	282	2007-12-11 08:58:35.308	2007-12-11 08:58:35.308	01002F	Egypt	SU-GBW	A321	\$7
14	285	2007-12-11 08:58:35.775	2007-12-11 08:58:35.775	01003A	Egypt	SU-KBA	A320	\$7
15	323	2007-12-11 08:58:42.996	2007-12-11 08:58:42.996	010090	Egypt	SU-GBZ	A320	\$7

The second query then displays a list of all aircraft with the \$7 UserTag. In this case we want to associate planesymbolB.txt file with any Estonian registered aircraft.

```
UPDATE Aircraft SET UserTag="$B" WHERE Registration LIKE "ES-%"  
SELECT * from Aircraft WHERE Registration LIKE "ES-%";
```

Here every entry in the 'Aircraft' table where the 'Registration' starts with ES- is given a UserTag of %B. Again the second query displays a list of the resulting update and changes.



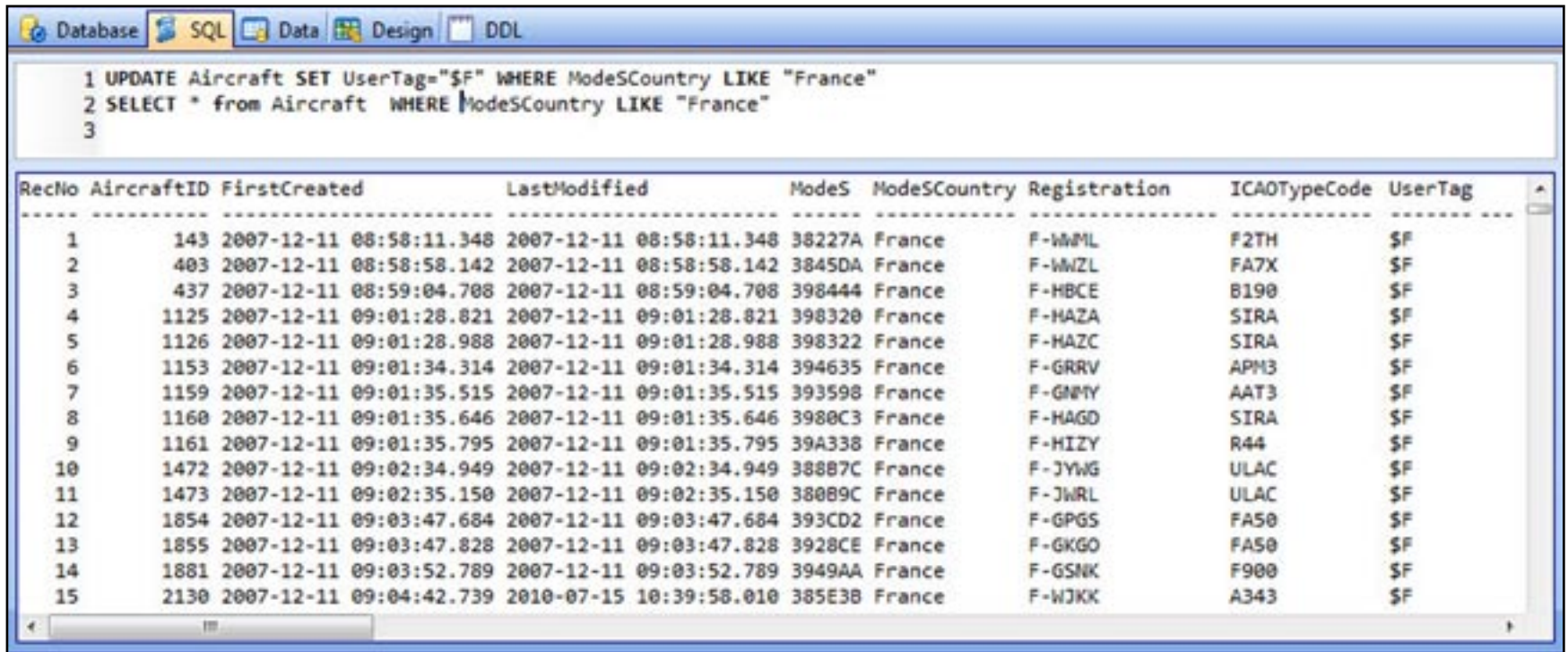
The screenshot shows a database management interface with a menu bar (Database, SQL, Data, Design, DDL) and a query editor containing two SQL statements. Below the editor is a table with 11 columns: RecNo, AircraftID, FirstCreated, LastModified, ModeS, ModeSCountry, Registration, ICAOTypeCode, and UserTag. The table contains 15 rows of data, all with a UserTag of '\$B'.

RecNo	AircraftID	FirstCreated	LastModified	ModeS	ModeSCountry	Registration	ICAOTypeCode	UserTag
1	8178	2007-12-11 09:25:03.791	2010-04-01 09:32:11.898	511086	Estonia	ES-ABP	B735	\$B
2	14188	2007-12-11 09:45:45.555	2007-12-11 09:45:45.555	4AB1D5	Sweden Mil	ES-PJR(!)	J532	\$B
3	15734	2007-12-11 09:51:12.637	2007-12-11 09:51:12.637	511063	Estonia	ES-PHR	H25B	\$B
4	16240	2007-12-11 09:53:03.283	2007-12-11 09:53:03.283	51107D	Estonia	ES-LBD	B733	\$B
5	16505	2007-12-11 09:54:01.270	2007-12-11 09:54:01.270	51100F	Estonia	ES-PVS	LJ60	\$B
6	16508	2007-12-11 09:54:01.986	2007-12-11 09:54:01.986	511029	Estonia	ES-ABJ	B733	\$B
7	16509	2007-12-11 09:54:02.205	2007-12-11 09:54:02.205	51102B	Estonia	ES-PVD	LJ31	\$B
8	16511	2007-12-11 09:54:02.700	2007-12-11 09:54:02.700	51102E	Estonia	ES-PVP	LJ60	\$B
9	16512	2007-12-11 09:54:02.885	2007-12-11 09:54:02.885	511031	Estonia	ES-ABK	B733	\$B
10	16513	2007-12-11 09:54:03.077	2007-12-11 09:54:03.077	511033	Estonia	ES-PVH	LJ31	\$B
11	16514	2007-12-11 09:54:03.261	2007-12-11 09:54:03.261	511035	Estonia	ES-YLX	L39	\$B
12	16515	2007-12-11 09:54:03.445	2007-12-11 09:54:03.445	511037	Estonia	ES-SKY	C56X	\$B
13	16516	2007-12-11 09:54:03.649	2007-12-11 09:54:03.649	511038	Estonia	ES-ABL	B735	\$B
14	16517	2007-12-11 09:54:03.845	2007-12-11 09:54:03.845	51103B	Estonia	ES-PVC	LJ60	\$B
15	17069	2007-12-11 09:56:07.827	2007-12-11 09:56:07.827	731021	Iran	ES-TLE(!)	L39	\$B

The third example associates a plane symbol with an aircraft ModeSCountry indicator. Again there are two SQL queries.

```
UPDATE Aircraft SET UserTag="%F" WHERE ModeSCountry LIKE "France"  
SELECT * from Aircraft WHERE ModeSCountry LIKE "France";
```

This time every entry in the 'Aircraft' table is given a UserTag of %F where the aircraft has a ModeSCountry of France. Once the update has been executed the second query displays a list of the resulting update and changes.



The screenshot shows a database application window with a menu bar (Database, SQL, Data, Design, DDL) and a query panel containing two SQL queries. Below the queries is a table displaying the results of the second query. The table has columns for RecNo, AircraftID, FirstCreated, LastModified, ModeS, ModeSCountry, Registration, ICAOTypeCode, and UserTag. The results show 15 aircraft entries, all with ModeSCountry 'France' and UserTag '%F'.

RecNo	AircraftID	FirstCreated	LastModified	ModeS	ModeSCountry	Registration	ICAOTypeCode	UserTag
1	143	2007-12-11 08:58:11.348	2007-12-11 08:58:11.348	38227A	France	F-WMML	F2TH	\$F
2	403	2007-12-11 08:58:58.142	2007-12-11 08:58:58.142	3845DA	France	F-WMZL	FA7X	\$F
3	437	2007-12-11 08:59:04.708	2007-12-11 08:59:04.708	398444	France	F-HBCE	B190	\$F
4	1125	2007-12-11 09:01:28.821	2007-12-11 09:01:28.821	398320	France	F-HAZA	SIRA	\$F
5	1126	2007-12-11 09:01:28.988	2007-12-11 09:01:28.988	398322	France	F-HAZC	SIRA	\$F
6	1153	2007-12-11 09:01:34.314	2007-12-11 09:01:34.314	394635	France	F-GRRV	APM3	\$F
7	1159	2007-12-11 09:01:35.515	2007-12-11 09:01:35.515	393598	France	F-GNRY	AAT3	\$F
8	1160	2007-12-11 09:01:35.646	2007-12-11 09:01:35.646	3980C3	France	F-HAGD	SIRA	\$F
9	1161	2007-12-11 09:01:35.795	2007-12-11 09:01:35.795	39A338	France	F-HIZY	R44	\$F
10	1472	2007-12-11 09:02:34.949	2007-12-11 09:02:34.949	38887C	France	F-JYWG	ULAC	\$F
11	1473	2007-12-11 09:02:35.150	2007-12-11 09:02:35.150	38089C	France	F-JWRL	ULAC	\$F
12	1854	2007-12-11 09:03:47.684	2007-12-11 09:03:47.684	393CD2	France	F-GPGS	FA50	\$F
13	1855	2007-12-11 09:03:47.828	2007-12-11 09:03:47.828	3928CE	France	F-GKGO	FA50	\$F
14	1881	2007-12-11 09:03:52.789	2007-12-11 09:03:52.789	3949AA	France	F-GSNK	F900	\$F
15	2130	2007-12-11 09:04:42.739	2010-07-15 10:39:58.010	385E3B	France	F-WJXX	A343	\$F

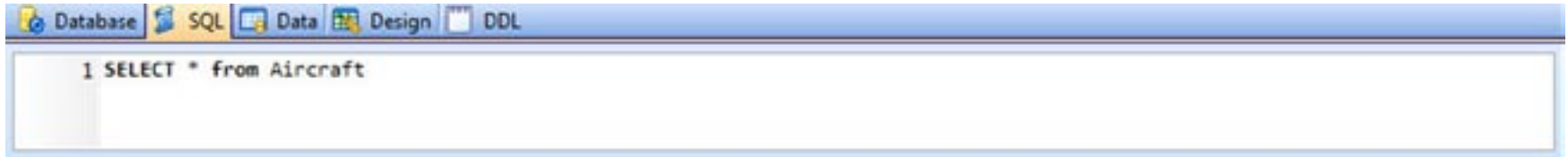
The queries only have to be entered as plain text in the Query panel. They can of course be copied and pasted from any source. The queries are initiated by either clicking the *F5* key or selecting *SQL*, *Execute SQL* from the main menu bar. Where there is more than one query they can be initiated one at a time using *Shift-F5*.

Restoring Changed Data

It is wise to make a copy of any database before attempting to make any change to its contents. It is possible however to undo any changes that have been made using the queries described above by the use of a further simple query.

```
UPDATE Aircraft SET UserTag=NULL WHERE UserTag LIKE "$%"
```

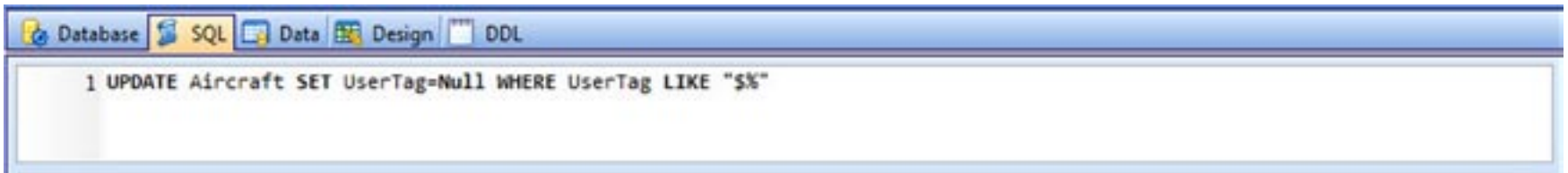
This query will remove all the UserTag entries having a \$x entry. If other type of UserTag entries have been made then these will not be affected.



The screenshot shows a window with a blue header bar containing icons for Database, SQL, Data, Design, and DDL. The main area is a white text box containing the SQL query: `1 SELECT * from Aircraft`.

The list of aircraft displayed can be restored to the complete contents of the table at anytime with this query.

```
SELECT * from Aircraft WHERE Registration LIKE "ES-%";
```



The screenshot shows a window with a blue header bar containing icons for Database, SQL, Data, Design, and DDL. The main area is a white text box containing the SQL query: `1 UPDATE Aircraft SET UserTag=NULL WHERE UserTag LIKE "$%"`.

A Note from COAA on the use of Plane Symbols

Although users can create and use different designs of plane symbols to their hearts content the original intention was that the symbols would be self selecting on the basis of the category code contained in the flight number Mode-S message. That code has four sets of eight entries with the first set 'A' being the most commonly used. A list of the ADS-B Emitter Category Set 'A' is as follows:

- 0 = No ADS-B Emitter Category Information
- 1 = Light (< 15 500 lbs.)
- 2 = Small (15 500 to 75 000 lbs.)
- 3 = Large (75 000 to 300 000 lbs.)
- 4 = High Vortex Large (aircraft such as B-757)
- 5 = Heavy (> 300 000 lbs.)
- 6 = High Performance (> 5 g acceleration and > 400kts)
- 7 = Rotorcraft

The overwhelming majority of aircraft encode 'A0' which mean that there is no category information available. The symbol selection is therefore currently based on a special coding (\$x) in the User tag field for each aircraft. If users wish to use further updates and enhancements of the PlanePlotter program then they are encouraged to conform to the defined types 1-7 as far as possible and only to use the code space \$8-\$F for the other specific types of symbol.

It may be that in the future more aircraft start to encode the category field correctly and COAA may decide to use that field to control the symbol. Even if they then allow the User tag to take precedence problems may arise for the user as an aircraft with no User tag say with a code 'A1' will invoke the planesymbol1.txt design. If a user has decided to use say 'A1' for an SR61 then some puzzling plots will be displayed if a Cessna 152 should then signal 'A1'!

Many thanks to Bev Ewen-Smith at COAA for his assistance and continuing work on PlanePlotter

More information on using plane symbols can be found at the
Yahoo Planeplotter group at <http://groups.yahoo.com/group/planeplotter>
or in the **PlanePlotters User Guide** now available at <http://www.planeplotters.com/ppug.html>