

[Rev up your memo searching capabilities with QMEMO]

Dealing with memo fields has always been something of a pain. Unlike the other fields in a table, where a simple Browse or Edit will display the data you want, memo fields require several steps before they're suitable for viewing - BROWsing the table, double clicking on the field in question, and then, usually, resizing the window. Anyone who's had to open up multiple windows in a project, screen or report table will vouch for me on this point.

Furthermore, finding the data in question is also a nuisance. Searching of the contents of a memo field is not Rushmore optimizable - even when the information you're searching for is in the beginning of the field. And you invariably end up using the "\$" operator anyway. Finally, because memo fields can contain gazillions of characters of data (it's true - see the System Capacities topic in FoxHelp), searches lend themselves to multiple conditions, which are often difficult to figure out and even harder to ensure that they're working correctly.

Bottom line: searching for something in a memo field is complex and S-L-O-W. Nonetheless, memo fields are critical for most developers, and if Microsoft took them away, I can just imagine the howls.

Many developers use a FoxPro database, complete with memo fields, to store messages of interest that they download from CompuServe's FoxForum. Tom Lewinson, a developer in New York City, is one of those developers. Several years ago, Tom wrote a simple front end for his FoxForum database that enables him to search for specific items inside the memo fields. As many of these projects do, his simple front end evolved, and the result is our Cool Tool of the Month: QMEMO.

Using QMEMO

QMEMO is a utility that provides a number of memo-searching capabilities all gathered under a single, easy to use interface. Let's take a walk through it.

After starting up QMEMO from the Command Window, you are prompted to select a table via a dialog similar to GETFILE's interface. If you select a table with more than one memo field, you are prompted to choose one of them. Nice touch: You can back out of this file and choose another one while you're in the "Select a Memo Field" dialog.

Step 1. Once you've got a handle on a memo field, the main QMEMO screen displays. Your first move will be to enter a search condition, and choose how the search expression is interpreted. A QMEMO search expression consists of one or more text strings separated by delimiters. The delimiters control how the Boolean evaluation of multiple strings is interpreted. This may seem a little confusing - a few examples will help.

Searching for a single string is accomplished with an expression like so:

```
FoxPro/
```

where the slash indicates the end of the expression. Preceding a string with an exclamation mark will search for a memo that does not contain the string:

```
!dBASE/
```

Memos can be searched for multiple expressions in a number of ways. Multiple strings are separated by a slash; how the strings are combined depends on the setting of a "Type of Search" radio button. If the radio button is set to "Type 1", the strings are interpreted as being joined with "AND" operators. If the radio button is set to "Type 2", the strings are interpreted as being joined with "OR" operators. For example, the expression

```
FoxPro/FoxBase/
```

would result in memos containing both strings if Type 1 was selected, but would result in memos containing one or the other or both strings if Type 2 was selected.

When included as a second delimiter, the asterisk acts as a "group separator" much like parens do in regular FoxPro expressions. The expression

```
FoxPro/FoxBase/*Visual FoxPro/
```

will search for memos that contains either FoxPro and FoxBase, or Visual FoxPro. If the Type 2 radio button was selected, the expression would result in memos that contain Visual FoxPro and either FoxPro or FoxBase.

In order to help you develop expressions, the FoxPro equivalent expression displays below the string you've entered.

Step 2. Once you've selected a search condition, you can further narrow the search by specifying a maximum length of match. In other words, you can look for matches that occur within X number of bytes of each other. This capability is similar to some sophisticated on-line retrieval services where you can look for all instances of when "Bill Clinton" and "MacDonald's" within 100 words of each other.

Step 3. You can specify whether or not the search should be case sensitive.

Step 4. You can set a filter condition on other fields in the table. (See below.)

Step 5. You can save both search conditions and filters and recall them for later use.

Step 6. After a search condition has been entered and validated, you can either do a record-at-a-time search, or search the file non-stop, logging the "hits" to a file.

Step 7. If you've selected the record-at-a-time search and you find a match, an extensive dialog displays. You have the options to continue searching in the same memo, more records, or to switch to auto-logging. You can also manually move the record pointer, start a new query, or select a different memo field to search.

If you've selected the auto-logging search, the record numbers of the records that match are stored in a file. You can then page from record to record in the hit file or view all records.

Additional Capabilities

There are extensive dialogs behind several of the steps listed above. For instance, when creating a filter on other fields to winnow down the set of records to be searched, you are led through a series of about six dialogs to build up to 15 conditions. There is actually a complete filter building program that is called from QMEMO.

Another example comes when selecting the Save pushbutton in order to save a search condition or a filter. You are prompted whether to save the search condition or the filter, and to give it a name. Then, when you opt to recall something (either a search condition or a filter), you are led through a table of expressions and can select one or multiple expressions. The selection process includes the ability to tag all or none, toggle the currently selected expression, and delete an expression. Finally, a comprehensive Help system is available at the press of F1. Not only does he explain how to use QMEMO but also gives some insight on its internal workings and how to optimize searching.

By the way...

It may seem that I should be done with my description of QMEMO, but I've not mentioned one slight feature that might be of interest to, say, one or two of you. QMEMO is FAST! I ran informal timing tests on several tables with memo files ranging from 400K to 4 MB, and found QMEMO to find the same text anywhere from 2 to 5 times as fast as commands issued interactively. Obviously, search times depend on the amount of RAM, caching, and disk speed; these tests were run on a ThinkPad 350C with an ordinary hard disk and 12 MB of RAM - your mileage may vary

Possible enhancements

Tom has spent a great deal of time working on the functionality and the engine underneath. As a result, it runs under DOS 2.5 but he hasn't produced a Windows or Mac version yet. And while the interface is completely Power Tool compatible, I've found that it isn't always intuitive as what to do next; I suspect that with some feedback, he may provide revise the placement of the objects to be more user friendly. Furthermore, the filter tool writes directly to the desktop and assumes 25 line mode, so if you've got a number of system windows open in 50 line mode, you can temporarily lose your place. Nonetheless, after just a few minutes, you can be up and running with it; and be completely comfortable in under an hour. If you do any large amount of searching through memo fields, you'll want to investigate this tool.

Where to find it

[options 1 - if it's a FoxTalk exclusive]

QMEMO is a FoxTalk exclusive. QMEMO.APP as well as the source code can be found on this month's Companion Disk.

[option 2 - if it's a regular Cool Tool - I don't know how much he's charging, or if he's even figured that out.]

QMEMO is shareware; QMEMO.APP can be found on this month's Companion Disk, as well as on CompuServe's FoxForum. You can also get it directly from Tom at Thinker's Apprentice, 212.222.5050 or on CompuServe, 70461,151 for \$____. When you register your copy, you'll get the source code as well.