

## Extending the FoxPro Function Set: The Ultimate FoxPro Libraries

After cracking open the Language Reference and reading page upon page of descriptions of commands and functions, you begin to wonder if Microsoft left anything for the future. The hundreds of commands and functions take over a thousand pages of documentation, to the point where reference manuals are now measured in pounds rather than pages.

Nonetheless, as you work with FoxPro, you'll find yourself creating short routines to determine calculate values or perform operations that aren't part of the standard function set. The problem is a matter of time. Sure, the excitement of creation is one of the better benefits of being a programmer - even the most jaded will admit to a secret vicarious thrill when a little routine works the first time out. However, a great deal more than a great idea and five minutes are required to create solid, reusable functions. Testing, debugging, bulletproofing, documenting, and then repeating the process over and over as new twists on the original problem are discovered - this is what takes time. The solution many look for is a pre-designed collection of routines to continue where FoxPro's native command and function set left off.

Recognizing this, a number of people have put together formal libraries of functions and programs. Probably the most well-known is Mac Rubel's FoxPro Power Tools - a collection of 400+ programs that started out life as a book and is now a commercial product. Minnesotan Ira Wald's ProTools - 210 routines strong - is also widely used and has a strong following among FoxForum regulars. A more manageable library is Michel Creppy's 80 function Ultimate FoxPro Library.

Creppy has organized his routines into over a dozen categories: array handling, data formatting, database operations, dates and times, display utilities, handling of the environment, errors, and event, I/O utilities, DOS functions, password and security, strings and text handling, and validations. Let's take a look at a few of them.

I've found the array handling functions to be the most useful. A\_SLICE copies one or more columns from one to another, A\_JOIN combines two array into one - the second "next" to the first, and A\_TAG combines two arrays into one - the second "below" the first. Other functions provide the ability to increase or decrease the width of an array (A\_ADDCOL and A\_DELCOL), determine the number of times a value occurs in a specific row or column (A\_OCCURS), eliminate duplicates in a specific column of an array (A\_CLEAN) and sort an array for unique values in a particular column (A\_UNIQUE).

Another category of routines that each of us has worked on at one time or another is data formatting. There are functions to convert AM/PM times to 24 hour format (TO\_24HRS), expand a time string to full length or AM/PM time strings (IS\_TIME and TO\_AM\_PM), format negative numbers into floating brackets format (BRACKETS) and people's names into a telephone directory-like format (S\_F\_NAME), and strip a telephone number of non-numeric characters - for use with communications software (CLEANTEL).

Date and time calculations (as opposed to formatting) give you access to all those data elements that users want but FoxPro doesn't provide. FIRSTDOM and LASTDOM return the date of the first and last day of a specific month. DATE\_GAP figures the difference between 2 dates (in days, weeks, months, quarters, or years), while TIME\_GAP returns the difference in seconds between two times.

If you've not developed your own set of routines for password entry and maintenance, you might find GET\_PSSW (password entry), ONE\_WAY (one way password encryption), and VALID\_PW (password validation for conformance to tight security criteria). Even if you've already developed your own password validation, you might find some interesting ideas about ensuring users don't try to cheat the intent of passwords.

If you're into the whimsical, you might want to take a look at GLIDE. This function provides the ability to glide a window across the screen, display a message that "wraps" around the screen, and either construct a message out of "falling" letters and words, or dissolve a message by dropping those "falling" letters and words off the screen.

The string and text processing functions include a check to see if a character string is composed entirely of numbers (NUMERIC) or specific characters (FILLED), divide a string into a list of individual character parameters (EXPLODE), extract the primary filename or filename extension from a full DOS filename or split a DOS pathname into file and path portions (SIMPLNAM, XTENSION, and FILENAME), and reduce consecutive spaces, tabs and CR/LFs to one space (REDUCE).

The documentation for the functions is contained in a 500K text file as well as lengthy headers in each program (in the source code). While Creppy doesn't adhere to the naming and coding conventions that have become popular since the 2.0 beta, he has a distinct and interesting style that you would do well to examine and learn from.

Creppy and his crew at Le Man Software in London, England, first wrote THE\_ULTIMATE and THE\_ULT ROUTINES LIBRARY assembler library for Data General minis. That library contains 300 routines, and the goal for their FoxPro version is to do the same. The Ultimate FoxPro Library is not freeware - the .APP is available on CompuServe's FoxForum or on the Companion Disk. Registering will provide you with the source code for all routines, the right to use the routines in your applications, lifetime support for the routines, and six months of updates (approximately 25 - 50 new functions).