August, 1999

Editor's View

Stumbling on the Leap Year

Leap Day 2000 is confusing people world-wide. Plus a new Visual Studio service pack has big wins for VFP developers.

By Tamar E. Granor, Editor

Two readers contacted me in the last few months asking about the year 2000 and its status as a leap year. The first was convinced that 2000 is not a leap year because he remembered learning that years divisible by 100 are not leap years.

The second was far more interesting. It came from a reader in India and included several newspaper clippings in a language he identified as Gujarati. According to the reader, the articles indicated that February, 2000 will have 30 days because of a special rule every 400 years! Since I don't read Gujarati, I can't confirm his translation, but I do see the numerals 2000 and 30 in the headlines for these stories.

It's not surprising that people are confused. The basic rule everyone knows is that any year divisible by four is a leap year. Most people also know that century years are excluded, but fewer are aware that years divisible by 400 *are* leap years. So, in fact, 2000 is a leap year (though February will only have 29 days, not 30), but 1900 was not.

Why such a confusing rule? Because nature isn't as precise as people want to be. It takes a little more than 365 days for the Earth to revolve around the sun. The "little more" is about a quarter of a day, but not exactly. The "no leap years in century years except those divisible by 400" rule is an adjustment to keep the calendar in synch with the Earth and Sun. Even this rule isn't perfect, though - every now and then, you hear about the time wizards adding a "leap second" to make a further adjustment.

For most of the world, 2000 is the first time this rule comes into play. The calendar we use (the Gregorian calendar) was adopted beginning in 1582. Great Britain (and, by extension, the US) switched in 1752. The last major country to sign on was Russia in 1918.

To make things more interesting, when the change occurred in each country, a number of days were simply dropped to bring the equinoxes and solstices back in synch with religious calendars. (You may have seen articles about George Washington's real birthday – February 11 - as opposed to the one we were taught in school – February 22. He was born before the Gregorian calendar was adopted in the British Empire.)

What does all this mean for you? First, the calendar change in 1752 (or whichever year applies to you) means that date math in VFP or any other application is reliable only back to the switchover date. Adding and subtracting days, months and years that overlap the change will lead to errors.

In addition, depending how your applications handle the year 2000, the leap year can mean trouble. If your solution to two-digit entry of years is to simply accept the entry, then add 100 years to all dates before a specified rollover, your code will fail on February 29, 2000. That's because there was no February 29, 1900 and FoxPro checks input dates for validity before even the Valid routine is invoked.

In fact, most of the code in Christof Lange's Y2K solution that appeared in the March '98 issue was addressed to the leap day problem. Code that inputs dates as character strings and then manipulates and validates the characters doesn't have this problem (though it can face other Y2K issues).

Bottom line: even if your Y2K work is done, you better go back and check that your code can handle that once in a quadricentennial event: Leap Day, 2000.

Service Pack Fixes Bugs and Adds Features

By the time you read this, Service Pack 3 (SP3) for Visual Studio will be available. Unlike earlier service packs, this one has a direct impact on Visual FoxPro developers. First, it includes dozens of bug fixes. In fact, more than two dozen are related to the infamous "C0000005" error alone. ("C0000005" is often the error code that appears when VFP crashes).

In addition, the service pack includes some new features. The one likely to affect the most developers is that the COMPILE command can now be used at run-time, so that you can create programs on the fly (the StrToFile() function added in VFP 6 helps with this), compile them and run them. There's also a new Session class that lets you create private data sessions without having to use a form or toolbar.

From Microsoft's point of view, the most important enhancements in this service pack relate to custom COM servers. In particular, a new run-time library solves a blocking problem in servers that meant that only one person could run a method at a time. With SP3, the problem is solved.

There are other enhancements as well, including some to the various Xbase tools (Application Builder and Coverage Profiler, among others) that come with VFP.

Since I'm writing this before the Service Pack is available, I can't tell you exactly where to find it. But you'll be able to download it from Microsoft's web site. A good place to start looking is msdn.microsoft.com\vfoxpro.