

August, 1995

Editor's View

From the Mailbag

Readers' letters make us think about what we're doing.

By Tamar E. Granor, Editor

"Letters, we get letters," David Letterman's TV band sometimes sings when Dave responds to viewer mail. Well, I get a few letters, but boy, do I get a lot of viewer e-mail.

Lots of recent messages have asked "When will Visual FoxPro be released?" By the time you read this, I'm pretty sure the answer will be, "It's shipping now."

Why So Soon?

A common thread in my recent mail is reaction to our coverage of Visual FoxPro. These messages are divided into two quite distinct groups - those who love it and those who hate it. There's not much to say to the first group except "thanks." For the most part, those who are unhappy raise two issues—"the product isn't available now" and "what about older versions?"

Last things first. As I've said here before, we'll continue to cover FoxPro 2.x as long as we believe a significant number of our readers are using it. In the last few issues, there have been many articles which apply only to FoxPro 2.x or to both 2.x and Visual FoxPro. We're working hard to see that each article is correctly labeled. We won't be covering anything older than 2.x, though—this is *FoxPro* Advisor, not *FoxBase* Advisor.

As to timing, the nature of magazine publishing and the nature of Beta testing seriously conflict with each other. We work three (sort of four, now) months in advance. You're reading the August issue, which hit the street around July 1. Our writers had an April 1 deadline for this issue. I'm writing this column in late April.

With a new product release, we're chasing a moving target. When we decided to devote March's issue to Visual FoxPro, we expected the product to ship in March or perhaps April. By the time it became clear that wouldn't happen, it was too late to change our plans.

In addition, there's no way we'd be able to provide enough information in just one issue to get people started. Even this issue (our fifth with Visual FoxPro coverage) still contains some introductory information—in this case, Karen Higgins' and Mac Rubel's articles on converting older applications.

If we'd waited for the product to ship before covering it (even if we'd have perfect foreknowledge and timed it so our first VFP issue coincided with release), you'd be months into working with the product before we could finish covering the basics.

While some of the articles weren't immediately useful to those without beta copies of the product, their value hasn't been lost over time. Instead, you now have a storehouse of useful information on Visual FoxPro.

Miscellany

I also get letters and e-mails on a wide variety of other topics. One reasonably frequent question is why we don't do more coverage of FoxPro/Mac (or sometimes, FoxPro/Unix). The answer to this is that we do. Most of our articles apply just as well to those platforms. If an article is platform-specific, we say so. If we don't say so, we believe it applies equally to all four FoxPro platforms.

As for not having articles aimed specifically at FoxPro/Mac or FoxPro/Unix, I welcome your proposals. If you're familiar with one of those platforms and have something to share with other FoxPro users, get a copy of our Writers' Guidelines and send me your ideas.

An interesting letter I received recently came from Hal Pawluk, one of the pioneers of Xbase. (Hal's the one responsible for the original dBASE II "Bilge Pump" ad and spent some time at Fox Software, as well.) Along with some tips and things, Hal offered this comment, "Don't use 'Hungarian notation' for variable names. Ever." and proceeded to explain why he dislikes that notation. Hungarian notation is the standard promulgated by The FoxPro Codebook and others which includes a first letter for scope ("g" for global, "l" for local, and so forth), a second letter for type ("c" for character, etc.) and then the actual name. While I don't personally agree with Hal's conclusions, he made me think about something I usually take for granted. Not a bad day's work.

Not long after Hal's letter reached me, I received an e-mail from another reader very upset about the naming convention adopted for the Visual FoxPro documentation. He says, Microsoft along with others "have settled on naming standards, without ever consulting the general FoxPro community." And further, "the standard calls for very strict naming conventions adopted long ago by one particular company." This reader considers it unfair for Microsoft to involve outsiders in the process. He feels those involved in designed the naming convention are receiving an unfair advantage.

Frankly, I think he missed the point. Take a look at the documentation for FoxPro 2.x. There is no naming convention in place, no standard. The examples are generally pretty good, but there's no guidance for developers regarding naming.

The most important thing about any naming convention is having it. The details don't matter. You need to have a set of rules and stick by it. The FoxPro 2.x docs don't.

For Visual FoxPro, Microsoft realized this was an important issue and proposed a set of naming conventions. They then accepted input from early Beta testers to improve the standard. (This is what Beta testing is all about, after all, and is exactly "consulting the general FoxPro community.") The end result is used throughout the documentation and is documented itself in the on-line help.

In addition, the standard is a modified version of Hungarian notation. While Flash Creative Management was the first to really push this notation in the FoxPro community, in fact, it's widely used throughout the programming world.

My take is that having a documented naming convention is a plus for all of us. No one will force you to use it, but if you do, your code will be easier for the rest of us to read. While we won't force our writers to use the suggested names (in fact, Kim Draeger's article in this issue includes a note about the rules in place at MicroEndeavors, where she works), we bet most of you will appreciate those who do.