Can VisualAge TPF Save The Day?

by Jeff Robinson

In the late 1980's, the world had the Microsoft's Disk Operating System (or DOS) which powered the majority of desktop computers everywhere. Forced to use a command-line, character-based, text-only interface, most users of DOS hated it and longed for something better. So, Microsoft gave them Windows and revolutionized the personal computer industry. (Apple and OS/2 lovers: silence, please).

Similarly, in the early 1990's, the world had the Internet — files and pages were manipulated via unsophisticated and unfriendly programs like FTP (file transfer protocol), Telnet, and Gopher. Web pages themselves began as unadorned, static, text-based documents, which were only for the sharing of scientific and academic papers. When the eventual demand came for more sophisticated and intelligent, hyper-linked web pages with graphics and animation, the Internet browser was born. Now, the "World Wide Web" is as popular as the television and the Internet is a household word. (Aren't you sick of those dot-com commercials?)

Likewise, in the late 1990's, IBM introduced to the TPF development community a new product that has the similar potential of changing things for the TPF industry: VisualAge TPF (or VA/TPF). Up until then, developers only had their VM-based or TSO-based environments to development and test their applications in. Now, with the advent of VA/TPF, they would have the choice of a more flexible, more modern, and more powerful PC-based, graphical user interface (GUI) development environment. In fact, it was even purported by industry insiders that VA/TPF could "revolutionize" TPF development forever.

What is VisualAge TPF?

You've probably heard or read all the hype by now concerning VA/TPF. By now you may have heard all about "Containers", "Families" and "remote" commands. So, for those unfamiliar with the basic product, here are some major items highlighting it:

- Features a Windows Explorer style GUI interface.
- Provides a powerful editor called LPEX (Live Parsing Extensible).
- Allows you to do workstation-based and mainframe-based tasks such as compiling, assembling, and loading your programs.
- Provides interactive source-level debuggers to test your TPF programs with.
- Includes a Performance Analyzer which provides graphical displays of how your code behaves during execution

However, a total VA/TPF development environment doesn't just consist of the stand-alone workstation product. In fact, there are several other external dependencies that must be in place before any installation can consider itself ready to use this tool. Here are some of those:

- Workstations must have a client-side Network File System (NFS) installed and properly configured.
- Proper configuration of OS/390 must be done so that remote (mainframe-based) tasks can be executed. This includes ensuring that the OpenEdition MVS (OMVS) component, a host of TCPIP listener programs called daemons, and the Hierarchical File System (HFS) are operational and ready for use.
- Other local compilers, linkers, etc. must be installed before those kind of local (workstation-based) actions can be done.
- A Source Code Management system should be installed so that developers can conveniently retrieve and update their programs. Otherwise, VM/CMS developers will have to manually transfer their existing programs to OS/390 and then transfer it back when they're done testing it.

Seizing The Day

Personally, I'm excited about the dawn of this new age in the TPF world and I think that anyone interested in the future of TPF should be also. Without a doubt, VisualAge TPF does introduce an important new chapter in the annals of TPF history. But will it revolutionize TPF in the same manner that Windows did for personal computes or that Browsers did for the Internet? The answer to that question is yet to be realized — but I hope so.

Right now, though, you may be pondering a more important question: Should you "Seize the Day" and bring VisualAge TPF into your own installation? Yes (what else did you expect me to say — I'm biased). But here are some important, realistic points to keep in mind as you consider whether to bring this potentially revolutionary product into your TPF shop.

1. VA/TPF Is Not A "Silver Bullet".
Like the silver bullets used to kill "werewolves" in the movies, VA/TPF will not "kill" all of your shop's development problems. In fact, never believe it when someone asserts that the purchase of a single tool into your company will cause all of your development problems to go away. Yes, some of your mainframe problems will be solved — but you'll discover newer problems germane to the PC-workstation environment.

2. Are You Prepared To Adequately Support The Product?
Moving developers from VM-based mainframe tools such as XEDIT and CMS to a slick editor like LPEX and a GUI interface like VA/TPF will be like teaching old dogs new tricks. Some may love the new environment, others will hate it, but they'll all be facing a learning curve that'll require that someone be available to help. Be prepared to dedicate a small staff of 1-3 persons just for that purpose.

3. Don't Harbor Unrealistic Expectations Concerning The Success of VA/TPF In Your Shop
If you're inclined to think that forcing your developers into a new environment will revolutionize your time-to-market ratio, think again. Most industry experts agree that it's not new tools that make for better programmers, but rather a better, more efficient way of doing things. So, if you think your developers are slow getting their code to production because of the "outdated" VM/CMS or TSO/ISPF environments, don't think that forcing them to use the new tools in VA/TPF will dramatically change anything.

4. Be Prepared To Promote The Product — Unceasingly.
A lot of old-timers will see the coming of VA/TPF like their grandparents saw the coming of the television: "It's only a fad that'll pass away in a few years." That means you'll have a lot of work to do in order to counter that influence in your shop. Make a commitment to sell the product in novel, creative ways so that developers will be excited about using it. Down the road when VA/TPF becomes the mainstay of TPF development, you can take credit for pushing its use at your installation.

5. Don't Stifle Your Success Due To Lack of Adequate Education and Training
Although VA/TPF may save TPF from the dark ages of development, it won't be successful if you don't plan on educating your developers in its use. This product is not like a bicycle — you can't just give it to developers and expect that they'll figure out how to use it. Even in this enlightened age of the Internet, many TPFers still aren't familiar with how all the new-fangled PC-based applications work. So, invest in education, sponsor training and encourage learning so your developers will know how to use this new, powerful weapon in the TPF arsenal.

For more information about VisualAge TPF, visit the IBM web-site at:

http://www.s390.ibm.com/products/tpf/visage.htm

For tools that can be used to customize the VisualAge TPF workstation environment, visit my own web-site at:

http://www.robisoft.com