

Top 10 Reasons to Use TPF/GI

An indispensable part of your application-testing toolbox.

Eliminates the need to switch views using cryptic commands.

1. Windows, Windows, Windows

Multiple windows show programmers all the information they need. Windows dock to save space and automatically refresh when programmers step.

Employs color to let users intuit what is happening and catch bugs quickly.

2. Color, Color, Color

Changed data is highlighted in a **contrasting color**. Programmers can visually verify their programs by watching fields change as they step.

Isolates programs and data to make early testing reliable.

3. Data and Program Isolation

Isolates programmers in virtual environments so that one test session can never corrupt another. No other tool matches TPF/GI in early testing reliability.

Shows programmers what they need to know, when they need to know it.

4. Intelligent Feedback

ECB info pops forward when users step. TPF application errors display without requiring users to press OK. Core areas gray out when invalid.

Combines a modern interface with legacy commands to reduce training.

5. Reduced Training Time

GUI reduces training time for novice programmers. Continued access to CMSTPF commands keeps experienced programmers happy.

Supports state-of-the-art debugging for both legacy and future needs.

6. State-of-the-Art Source View Debugging

Variables, including pointers, arrays, structs, and objects can be edited in C and C++. Expressions can be edited in Assembler and SABRE programs.

Reduces utilization of expensive hardware resources.

7. Breaks the Hardware Bottleneck

Programmers get low-cost, virtual TPF environments that allow them to use multiple systems and connect to remote resources.

Opens the door to ever-more-powerful debugging tools for CMSTPF users.

8. Doorway to Future Debugging Power

TPF/GI is an easy step up for current CMSTPF programmers, but contains enhancements and improvements that CMSTPF alone will never have.

Makes programmers more productive with a customizable GUI environment.

9. GUI Usability

Supports GUI features that newer programmers desire. Customizable colors and toolbars let all programmers hone their productivity to a razor-sharp edge.

Allows customer enhancements with an open, extensible environment.

10. Extensibility

An application programming interface (API) allows TPF/GI to be extended. Customers hook in their own library systems and enhancements.