Essential Resources For GUI Design

By Jim Douglas

The most challenging aspect of GUI programming isn't the mechanics of managing windows and controls. As we've seen in the accompanying article, many of the mechanical details can be automated. The real difficulty lies in learning how to think in an entirely new way.

In a GUI program, you don't have the same direct control over program flow as you do with a legacy character-mode program. Instead, you create a window with various graphical controls on it and, in essence, wait for the user to do something. As long as a control is visible and enabled, the user can activate it by, for example, pushing a button, selecting a menu item, checking a check box, or clicking a radio button. Your program needs to be prepared to respond in a logical way to whatever the user chooses to do.

GUIBuilder™ does a good job of managing the technical housekeeping details of an event-driven GUI program, but it can't help with program design. Because a GUI program operates very differently from a traditional character-mode program, it's a good idea to invest some time in reading and thinking about GUI design issues. The following references are a useful starting point:

- The Windows Interface Guidelines for Software Design
  (www.microsoft.com/win32dev/uiguide/)
- Macintosh Human Interface Guidelines
  (developer.apple.com/techpubs/mac/HIGuidelines/HIGuidelines-2.html)
- Interface Hall of Shame, Interface Hall of Fame, and useful links
  (www.iarchitect.com)
- Articles in previous issues of *The BASIS Advantage*:
  Right Resources Can Open Whole New Windows
  (www.basis.com/advantage/mag-v1n4/resources.html)
  Off The Shelf: Resources For The Programmer's Library
  (www.basis.com/advantage/mag-v1n4/offtheshelf.html)