

You ought to. *ToolBook* is, as its public relations and advertising loudly triumph, a way for almost anyone to create useful *Windows* applications without writing a single line of programming code. No kidding! Personnel Managers who want to lay out a short introductory piece on their companies, MIS Directors looking to rough out prototypes of in-house applications for approval by executive staff, students looking to impress instructors, in fact, nearly anybody can use *ToolBook* to create full fledged *Windows* programs quickly and easily. Honestly, anybody with a modicum of computer literacy and a couple of hours can now call themselves, if not a programmer, a designer. Is there a catch? Nope, there are several.

I

Profile

Y

M

Originally, we planned to review "*Windows Development Tools*" from two other companies in this article; by name, *Actor*, from The Whitewater Group, and *KnowledgePro/Win* from Knowledge Garden. We withdrew those two products from full consideration due to their respective natures versus *ToolBook* (see sidebar). *ToolBook*, a product that when fully installed occupies over six megabytes of hard disk (including a raft of sample applications; you can reduce space usage to something just over 1 MB), is the only one that provides true code-less development. The power comes at considerable expense, however. Not only does *ToolBook* **truly** require an 80286 or higher equipped system to run (we've talked before about other *Windows* applications that made such claims but didn't really need anything more than an 8088), but even on a midline '386 with 6 MB of available extended memory, the single word that best describes *ToolBook* is SLOW. Even if you happen to be one of the fortunate few who can afford to run *ToolBook* and its applications entirely from a RAM disk, plan on waiting around a lot, particularly if your applications get large. Since either the integration of many objects on individual "pages" or the inclusion of numerous pages in your "books" will quickly lead to great size, this will happen more often than not for users making serious business use of the product.

The idea behind *ToolBook* is beautiful in its simplicity. *Windows* is very modular, consisting of a continuous series of individual **objects**, such as boxes containing text, graphic images, dialog boxes and scroll bars to encourage easy interaction, etc. *ToolBook* enables you to tie these elements together quickly and easily to create the electronic equivalent of books. You can make books that read linearly (page one leads to page two (and/or three, four, or higher, but never back), choose to allow for returns through previously read pages, or enable readers to skip all over in what can almost appear to be as unencumbered as flipping through pages of a bound volume. The environment in which you work can be customized to a surprisingly high level, and control over the colors, attributes, and placement of your objects is nearly limitless; you can even assign passwords and restrict access rights. If you need to create an object such as a graphic image, *ToolBook* also provides a respectable set of tools with which to work.

Gee, this sounds neat, right? Well, it is, but there are problems besides the not-at-all-incidental speed issue. (Asymetrix promises they're working feverishly to address that, and the first of the following.) Art that you import to *ToolBook* from other applications can come in **only** via *Windows*'s Clipboard, so if you have a library of vector based images in a proprietary format, you probably can forget them, and even if you can convert them to an appropriate format, there's a size limit of 64 kilobytes that imported color images cannot exceed. Images you create in *ToolBook* are completely captive there. Learning to use *ToolBook* is frustrating, because the documentation is disjointed and leaves many holes for you to fill in (the graphic image size limitation, for example, is undocumented, and the error message generated is misleading).

Finally, there are a couple of things that we perceive as attitude problems. The installation program automatically and without first asking approval creates a separate program group for *Toolbook* and its applications (meanwhile installing only a few of the included samples therein), and the documentation makes a point that we find incongruous with the promoted purpose of the product. In the volume devoted to explaining *ToolBook*'s English-like programming language, targeted at those who choose to write code manually, you are told, flatly, to "look elsewhere" for an explanation of writing *Windows* dynamic link libraries. To their credit, the people at Asymetrix have put helpful information on the subject on-line on their electronic bulletin board, making themselves one of the elsewheres to look. Perhaps what this implies is that *ToolBook* is targeted at "real" programmers, after all.



**IT'S YOUR MONEY!**